Generating Informative Samples for Risk-Averse Fine-Tuning of Downstream Tasks

Heasung Kim[*], **Taekyun Lee**, **Hyeji Kim**, **and Gustavo de Veciana**
Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712
{heasung.kim, taekyun, hyeji, deveciana}@utexas.edu

Abstract

Risk-averse modeling is critical in safety-sensitive and high-stakes applications. Conditional Value-at-Risk (CVaR) quantifies such risk by measuring the expected loss in the tail of the loss distribution, and minimizing it provides a principled framework for training robust models. However, direct CVaR minimization remains challenging due to the difficulty of accurately estimating rare, high-loss events—particularly at extreme quantiles. In this work, we propose a novel training framework that synthesizes informative samples for CVaR optimization using score-based generative models. Specifically, we guide a diffusion-based generative model to sample from a reweighted distribution that emphasizes inputs likely to incur high loss under a pretrained reference model. These samples are then incorporated via a loss-weighted importance sampling scheme to reduce noise in stochastic optimization. We establish convergence guarantees and show that the synthesized, high-loss-emphasized dataset substantially contributes to the noise reduction. Empirically, we validate the effectiveness of our approach across multiple settings, including a real-world wireless channel compression task, where our method achieves significant improvements over standard risk minimization strategies.

1 Introduction

Risk-averse learning has become increasingly relevant in high-stakes applications where robustness to rare but costly failures is critical. In those domains, models must not only achieve strong average-case performance but also avoid catastrophic errors on atypical inputs. A widely adopted risk measure for capturing such sensitivity is the *Conditional Value-at-Risk* (CVaR), which focuses on the expected loss in the worst-performing $(1-\beta)$ fraction of the input space (Rockafellar and Uryasev, 2000), making it well-suited for applications requiring robustness guarantees, e.g., large language models, system scheduling, control, medical, wireless communications, and more (Chaudhary et al., 2025; Tan et al., 2017; Ahmadi et al., 2021; Chan et al., 2014; Yang et al., 2022).

Despite its appeal, minimizing CVaR remains challenging in practice. As the quantile level β approaches one, loss contributions become dominated by *rare, high-risk inputs that are unlikely to be observed through standard sampling* from the data distribution. Without adequate coverage of these tail events, naive Monte Carlo (MC) methods yield high-variance estimates of CVaR and inefficient optimization, ultimately limiting the reliability of risk-averse training.

[*]Corresponding Author.
Source code: https://github.com/Heasung-Kim/generating-informative-samples-for-risk-averse-fine-tuning-of-downstream-tasks

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

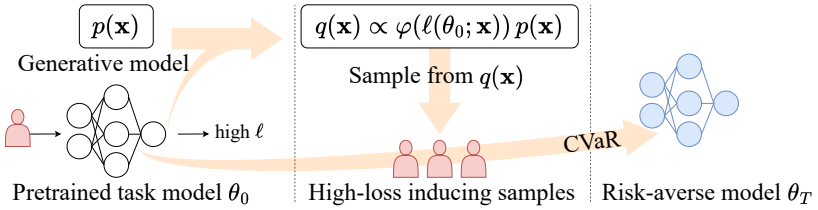

Figure 1: System overview. A score-based generative model is guided using the loss values from a pretrained model to sample high-loss inputs for CVaR optimization.

Recent advances in generative modeling offer new opportunities to address these limitations. In particular, score-based generative models enable expressive and controllable sampling from complex distributions, and have shown promise in tasks ranging from data augmentation to density estimation. Concurrently, pretrained task models are becoming widely available and serve as informative priors for task performance. These developments motivate a fundamental question: *Can we actively generate training inputs that are more informative for CVaR optimization?*

In this work, we propose a novel framework that integrates pretrained (reference) models and generative models to synthesize *informative samples* for risk-averse training. Our key observation is that inputs which induce high loss under a pretrained model are highly beneficial for risk-averse model training. This motivates a data generation strategy that explicitly targets failure modes of the initial model and uses them to guide risk-aware training more effectively.

To realize this idea, we develop a method that uses pretrained loss values to guide a score-based generative model toward a reweighted sampling distribution that emphasizes high-risk inputs. Our approach leverages recent advances in training-free guidance for diffusion models, allowing the generative process to be steered without retraining (Chung et al., 2022; Yu et al., 2023; Kim et al., 2025c;a). The resulting samples are explicitly biased toward regions where the model is likely to fail and are used to perform CVaR minimization via importance-weighted optimization. The main contributions of this work are summarized as follows.

Framework Design. We propose a novel risk-averse learning framework based on loss-guided generative importance sampling. As illustrated in Figure 1, our approach proceeds in two stages: (i) a pretrained model is used to guide a score-based generative model to sample from a reweighted distribution that emphasizes high-loss inputs; (ii) these samples are then used for CVaR minimization via importance-weighted training, resulting in improved robustness and reduced training noise.

Theoretical Analysis. We provide convergence analysis for our framework and show that, under mild assumptions, generating samples from high-loss regions provably reduces the noise of the CVaR optimization process.

Empirical Validation. We empirically validate our method in both synthetic and real-world settings. In a controlled regression task with highly imbalanced modes, the proposed method successfully synthesizes rare, high-loss samples that are critical for minimizing tail risk. In a real-world application—wireless channel state information (CSI) compression—our method consistently improves CVaR performance in the high β regime, compared to existing robust and risk-minimization baselines.

To the best of our knowledge, this is the first work that leverages generative models to perform risk-averse learning by targeting high-loss regions via loss-guided importance sampling.

2 Related Work

Risk-Averse Learning and Conditional Value-at-Risk. Risk-averse learning seeks to prioritize robustness over average-case performance, especially in high-stakes settings where rare but severe failures are unacceptable. A widely used risk measure in this context is the *Conditional Value-at-Risk* (CVaR) (Rockafellar and Uryasev, 2000), which quantifies the expected loss in the $(1-\beta)$-worst portion of the input distribution. Due to its ability to explicitly penalize high-loss instances, CVaR has been adopted in a broad range of applications, including finance, credit, operational risk management, robust control in wireless communications, large language models, and more (Alexander et al., 2006;

Andersson et al., 2001; Filippi et al., 2020; Yang et al., 2022; Chaudhary et al., 2025; Chow and Ghavamzadeh, 2014).

However, despite its appeal, CVaR training is notoriously challenging due to the high variance in empirical estimates, particularly when targeting extreme quantiles (Troop et al., 2021).

Importance Sampling for Risk Measures and Optimization. Importance sampling can be utilized for improving the variance of risk estimation, particularly when rare events dominate the objective. Prior works have explored its use in CVaR estimation and optimization (Bardou et al., 2009; Deo and Murthy, 2021; He et al., 2024a), including sampling-based gradient estimators based on likelihood ratios (Tamar et al., 2015). However, these approaches typically operate on a fixed dataset and focus on reweighting existing samples to reduce estimation variance.

Our method introduces a fundamentally new perspective: we utilize a generative model to directly *synthesize* importance-weighted samples. Under the availability of a generative model, we guide the sample generation process toward high-loss regions using pretrained model losses. This enables us to reduce the noise of CVaR optimization while expanding the effective support of the training distribution.

Generative Models for Data Augmentation and Downstream Task Learning. Recent progress in generative modeling, particularly in diffusion and score-based generative models, has enabled high-fidelity sample generation and accurate distribution approximation (Chen et al., 2024; Wang et al., 2024b). These models have been successfully applied across diverse domains for data augmentation, including load forecasting (Xu and Zhu, 2024), medical imaging (He et al., 2024b), and audio synthesis (Bahmei et al., 2022). Recently, generative models are increasingly used to augment training datasets with specific purposes (Zheng et al., 2023), e.g., enhancing semantic diversity (Shivashankar and Miller, 2023; Trabucco et al., 2023), generating label-specific instances (Shao et al., 2019), and bridging distributional gaps between training and test data (Wang et al., 2024a).

Scope and Distinctiveness of the Proposed Approach. While prior work has primarily leveraged generative models to enhance generalization by enriching the diversity of training data, our approach adopts a different objective: synthesizing samples that are explicitly *informative for risk-sensitive training*. Rather than uniformly augmenting the training distribution, we concentrate generation toward high-loss regions—those most relevant for CVaR minimization. This targeted generation paradigm aligns directly with risk-averse learning objectives, offering a principled and efficient path toward robust model training.

3 Preliminaries and Problem Formulation

Recent advances in generative modeling, particularly score-based generative models, have substantially improved the quality and controllability of synthetic data generation. A key strength of the score-based generative models lies in their ability to support *guided sampling*, where samples can be drawn from a distribution that is shifted or reweighted relative to a base distribution. In this work, we leverage this capability to generate *rare, high-loss-inducing samples* that are underrepresented in standard datasets but critical for risk-sensitive objectives. As we will show, synthesizing such samples provides both theoretical and practical advantages in CVaR minimization.

3.1 Score-based Generative Models and Training-Free Guided Sampling

Let $\mathbf{X}^p \in \mathbb{R}^{d_1}$ be a random variable with density $p(\mathbf{x})$, where $\mathbf{x}$ denotes a realization. A generative model seeks to approximate $p(\mathbf{x})$ or its associated score function $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ to enable efficient sampling from the underlying distribution. A key innovation of the recent score-based generative models is to model the score function not directly on $p(\mathbf{x})$, but on noise-perturbed data distributions $p_t(\mathbf{x})$ indexed by a continuous-time parameter $t \in [0, T]$. This enables the data generation process to be formulated as a stochastic differential equation (SDE), which has been shown to enhance both training stability and sample quality (Song et al., 2021). The perturbed distributions are modeled via the Itô SDE:

$$\mathrm{d}\mathbf{X}^p_{(t)} = \mathsf{f}(\mathbf{X}^p_{(t)}, t)\,\mathrm{d}t + \sigma(t)\,\mathrm{d}\mathbf{W}_{(t)}, \quad \mathbf{X}^p_{(0)} \sim p(\mathbf{x}), \tag{1}$$

where $\mathsf{f} : \mathbb{R}^{d_1} \times [0,T] \to \mathbb{R}^{d_1}$ is the drift term, $\sigma(t) : [0,T] \to \mathbb{R}$ is the diffusion coefficient, and $\mathbf{W}_{(t)}$ is a standard d_1-dimensional Brownian motion. The marginal distribution of $\mathbf{X}^p_{(t)}$ is denoted $p_t(\mathbf{x})$, and the initial distribution $p_0(\mathbf{x})$ corresponds to the data distribution we aim to model.

Given access to the time-indexed score $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$, samples from $p = p_0$ can be obtained by solving the reverse-time SDE (Anderson, 1982):

$$\mathrm{d}\mathbf{X}^p_{(t)} = \left(\mathsf{f}(\mathbf{X}^p_{(t)}, t) - \sigma(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{X}^p_{(t)})\right) \mathrm{d}t + \sigma(t)\, \mathrm{d}\tilde{\mathbf{W}}_{(t)},$$

where $\tilde{\mathbf{W}}_{(t)}$ denotes reverse-time Brownian motion. Sampling is typically initialized from a simple prior such as a standard Gaussian $p_T(\mathbf{x})$ for large T, and trajectories are integrated backward to recover $\mathbf{X}^p_{(0)} \sim p_0$. For notational simplicity, scalar-vector multiplication denotes elementwise scaling.

Beyond sampling from the base distribution $p(\mathbf{x})$, recent advancements in the score-based generative models allow sample generation from modified target distributions of the form $q(\mathbf{x}) \propto w(\mathbf{x})p(\mathbf{x})$, where $w(\mathbf{x})$ is a task-specific importance weight. While classical approaches such as the cross-entropy method (CEM) and fine-tuning of generative models require retraining to realize such reweighted distributions, training-free guidance techniques for the score-based generative models enable approximate sampling from q without modifying the base generative model. These methods exploit the fact that, under the same SDE dynamics in (1), if a process begins from $q(\mathbf{x}) = q_0(\mathbf{x}) \propto w(\mathbf{x})p(\mathbf{x})$ instead of p as $\mathrm{d}\mathbf{X}^q_{(t)} = \mathsf{f}(\mathbf{X}^q_{(t)}, t)\, \mathrm{d}t + \sigma(t)\, \mathrm{d}\mathbf{W}_{(t)}$ with $\mathbf{X}^q_{(0)} \sim q = q_0$, then its marginal q_t such that $\mathbf{X}^q_{(t)} \sim q_t$ satisfies $q_t(\mathbf{x}) \propto p_t(\mathbf{x})\, \mathbb{E}_{\mathbf{X}^p_{(0)} \sim p(\cdot \mid \mathbf{X}^p_{(t)} = \mathbf{x})} \left[w(\mathbf{X}^p_{(0)}) \right]$, where $p(\cdot \mid \mathbf{X}^p_{(t)} = \mathbf{x})$ denotes the conditional distribution of the initial state given state $\mathbf{x}$. Taking the log and the gradient yields the identity:

$$\nabla_{\mathbf{x}} \log q_t(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) + g(\mathbf{x}, t), \quad \text{where} \quad g(\mathbf{x}, t) \coloneqq \nabla_{\mathbf{x}} \log \mathbb{E}_{\mathbf{X}^p_{(0)} \sim p(\cdot \mid \mathbf{X}^p_{(t)} = \mathbf{x})} \left[w(\mathbf{X}^p_{(0)}) \right].$$

This additional term g, often referred to as the *guidance*, can be approximated using known quantities such as the score function of p_t and the weight function w (Chung et al., 2022; Kim et al., 2025c; Yu et al., 2023), enabling sampling from approximated q via the reverse-time SDE without any further training.

Motivated by these developments, we propose a new learning framework that leverages guided sample generation to construct *informative training data* specifically tailored for *risk-averse learning*. Rather than drawing training samples uniformly from $p(\mathbf{x})$, we aim to generate samples that contribute to noise reduction in risk-sensitive objectives, thereby improving model robustness to rare but high-loss events.

3.2 Risk-Averse Learning via Conditional Value-at-Risk

Our ultimate goal is training of *risk-averse* task models, in which the objective is not merely to optimize expected model performance, but to mitigate the impact of potentially rare but high-loss outcomes. We consider *Conditional Value-at-Risk (CVaR)*, one of the most widely adopted risk measures, which builds upon the concept of *Value-at-Risk (VaR)*.

Let $\theta \in \mathbb{R}^{d_2}$ denote the parameters of the task model, and let $\ell(\theta; \mathbf{x})$ be the loss incurred on input $\mathbf{x}$. For a given quantile (confidence) level $\beta \in (0,1)$, the VaR is defined as the smallest threshold α such that the loss does not exceed α with probability at least β:

$$\mathrm{VaR}_\beta(\theta) = \min\{\alpha \in \mathbb{R} : \mathbb{P}[\ell(\theta; \mathbf{X}^p) \le \alpha] \ge \beta\} \tag{2}$$

where $\mathbb{P}[\ell(\theta; \mathbf{X}^p) \le \alpha] = \int_{\ell(\theta;\mathbf{x}) \le \alpha} p(\mathbf{x})\, \mathrm{d}\mathbf{x}$, and the distribution is assumed to be continuous with respect to α. While VaR captures a quantile of the loss distribution, it does not reflect the magnitude of losses beyond the threshold. The Conditional Value-at-Risk addresses this by computing the expected loss in the tail beyond $\mathrm{VaR}_\beta(\theta)$:

$$\mathrm{CVaR}_\beta(\theta) = \mathbb{E}_{\mathbf{X}^p \sim p}[\ell(\theta; \mathbf{X}^p) \mid \ell(\theta; \mathbf{X}^p) \ge \mathrm{VaR}_\beta(\theta)]. \tag{3}$$

Our objective is to find model parameters θ^* that minimize the CVaR at a given quantile level β:

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^{d_2}} \mathrm{CVaR}_\beta(\theta). \tag{4}$$

Algorithm 1 Risk-Averse Model Training via Loss-Guided Importance Sample Generation

Input: Initial model θ_0, generative model $\nabla \log p_t(\mathbf{x})$, confidence level β, function φ, dataset $\mathcal{B}$

Output: Risk-averse model θ_K

1: Generate importance samples $\{\mathbf{x}_i\}_{i=1}^{B_q} \sim q(\mathbf{x}) \propto \varphi(\ell(\theta_0; \mathbf{x}))\, p(\mathbf{x})$
2: Compute $Z = \mathbb{E}_{\mathbf{X}^p \sim p}[\varphi(\ell(\theta_0; \mathbf{X}^p))]$
3: Initialize $\alpha_0 \leftarrow Z$.
4: **for** $k = 1$ **to** $K + 1$ **do**
5: Sample data pair $\{\mathbf{x}, \ell(\theta_0; \mathbf{x})\}$ from q
6: MC estimation of $\alpha_{k-1} + \mathbb{E}_{\mathbf{X}^q \sim q}\left[\frac{Z(\ell(\theta_{k-1}; \mathbf{X}^q) - \alpha_{k-1})^+}{\varphi(\ell(\theta_0; \mathbf{X}^q))(1-\beta)}\right]$
7: $(\theta_k, \alpha_k)^\top \leftarrow \texttt{SubGradientDescent}(\partial F_\beta, \theta_{k-1}, \alpha_{k-1})$

This objective is known to be equivalent to the following unconstrained optimization problem (Rockafellar and Uryasev, 2000) as

$$\theta^*, \alpha^* = \arg \min_{\theta \in \mathbb{R}^{d_2}, \alpha \in \mathbb{R}} F_\beta(\theta, \alpha) \;\text{ where }\; F_\beta(\theta, \alpha) = \alpha + \frac{1}{1-\beta} \mathbb{E}_{\mathbf{X}^p \sim p}[(\ell(\theta; \mathbf{X}^p) - \alpha)^+]. \quad (5)$$

Here, $(x)^+ = \max(x, 0)$ denotes the positive part function. The solution α^* corresponds to VaR_β, and θ^* minimizes CVaR_β.

4 Risk-Averse Model Training via Loss-Guided Importance Samples

While the CVaR objective in (5) provides a principled framework for addressing tail-risk scenarios in downstream tasks, its empirical estimation is particularly challenging, especially at high confidence levels ($\beta \to 1$). In this regime, the corresponding VaR threshold α becomes large, and the expectation term $\mathbb{E}_{\mathbf{X}^p \sim p}[(\ell(\theta; \mathbf{X}^p) - \alpha)^+]$ becomes increasingly difficult to estimate due to the *rarity of high-loss instances*, for which $(\ell(\theta; \mathbf{X}^p) - \alpha)^+$ is nonzero. For most samples from $p(\mathbf{x})$, this term evaluates to zero, leading to high variance and poor gradient signals during training. Consequently, naive sampling from the base distribution $p(\mathbf{x})$, even with a generative model, becomes inefficient and often requires an infeasibly large number of samples to stably estimate the CVaR objective for training.

Key Idea. To address these, we propose leveraging the generative model to sample from an *importance-weighted* distribution $q(\mathbf{x})$ tailored to highlight high-loss regions. Based on the availability of a pretrained model θ_0 and a score-based generative model capable of sampling from $p(\mathbf{x})$, our approach consists of two components: (i) Sample inputs from a weighted distribution $q(\mathbf{x}) \propto \varphi(\ell(\theta_0; \mathbf{x}))\, p(\mathbf{x})$, where $\varphi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a nondecreasing function that prioritizes high-loss examples. (ii) Use the importance samples to perform CVaR minimization via importance-weighted MC estimation.

Intuitively, this sample generation strategy concentrates on examples that are informative for CVaR optimization, those in the tail of the loss distribution. These rare, high-loss instances expose the model to critical failure modes and enable more effective risk-averse training. We refer to our approach as *Risk-Averse Model training via loss-guided Importance Samples* (RAMIS).

4.1 Algorithm

Algorithm 1 takes as input: an initial pretrained model θ_0, a score-based generative model capable of sampling from $p(\mathbf{x})$, a target quantile level $\beta \in (0, 1)$, and a non-decreasing weighting function φ used to construct the importance sampling distribution. We assume access to a dataset $\mathcal{B} = \{\mathbf{x}_i\}_{i=1}^{B}$, where each $\mathbf{x}_i$ is drawn i.i.d. from the base distribution $p(\mathbf{x})$. This dataset may be externally provided or synthesized via the generative model.

Line 1: We generate samples from the importance-weighted distribution, $q_0(\mathbf{x}) = q(\mathbf{x}) \propto \varphi(\ell(\theta_0; \mathbf{x}))\, p(\mathbf{x})$, by guiding the generative model using loss values computed under θ_0. In score-based generative models, this corresponds to solving the following reverse-time SDE:

$$\mathrm{d}\mathbf{X}^q_{(t)} = \left(\mathsf{f}(\mathbf{X}^q_{(t)}, t) - \sigma(t)^2 \nabla_{\mathbf{x}} \log q_t(\mathbf{X}^q_{(t)})\right) \mathrm{d}t + \sigma(t)\, \mathrm{d}\tilde{\mathbf{W}}_{(t)}. \quad (6)$$

The specific implementation of this guided importance sampling process may vary based on the chosen generative model guidance method and is detailed in Appendix B.

Line 2: The expectation of the importance weight function $\varphi(\ell(\theta_0;\mathbf{x}))$ over the base distribution $p(\mathbf{x})$ is computed as $\mathbb{E}_{\mathbf{X}^p\sim p}[\varphi(\ell(\theta_0;\mathbf{X}^p))] = Z$. This normalization factor Z can be approximated as $Z \approx \frac{1}{|\mathcal{B}|}\sum_{\mathbf{x}\in\mathcal{B}}\varphi(\ell(\theta_0;\mathbf{x}))$ and is utilized in subsequent optimization iterations.

Lines 4–7: At each training step, we draw a sample from $q(\mathbf{x})$ along with its corresponding importance weight $\varphi(\ell(\theta_0;\mathbf{x}))$. The CVaR objective $\alpha_{k-1} + \mathbb{E}_{\mathbf{X}^q\sim q}\left[\frac{Z(\ell(\theta_{k-1};\mathbf{X}^q)-\alpha_{k-1})^+}{\varphi(\ell(\theta_0;\mathbf{X}^q))(1-\beta)}\right]$ is estimated via MC, which corresponds to a variational form in (5). Note that the likelihood ratio $p(\mathbf{x})/q(\mathbf{x})$ simplifies to $Z/\varphi(\ell(\theta_0;\mathbf{x}))$. We perform subgradient-based optimization using a subroutine `SubGradientDescent` (see Appendix A for details), updating both θ and α in the direction that minimizes the estimated CVaR objective.

Importance Sampling Mechanism. The proposed approach differs fundamentally from conventional importance sampling techniques, which re-evaluate model performance at each iteration and dynamically adjust sampling probabilities over a fixed dataset. Such methods introduce additional per-iteration computational overhead (El Hanchi and Stephens, 2020; Needell et al., 2014; Zhao and Zhang, 2015).

In contrast, our framework adopts a fixed importance sampling distribution constructed prior to training. We *guide the generative model to directly produce samples* from the target importance-weighted distribution. This eliminates the need for iterative reweighting or per-batch loss evaluations. Importantly, during training (Lines 4–7 in Algorithm 1), our method introduces *no additional computational overhead* beyond a lightweight scalar reweighting of the loss term $(\ell(\theta;\mathbf{x}) - \alpha)^+$.

4.2 Theoretical Analysis

In this subsection, we present a convergence analysis of the proposed RAMIS framework and justify how loss-guided importance sampling improves risk-averse training. Specifically, we show that sampling from a reweighted distribution, which is biased toward high-loss regions under a reference model, reduces the noise of stochastic gradient descent.

Assumption 1 (Convexity, smoothness, and bounded loss)**.** For all $\mathbf{x} \in \mathbb{R}^{d_1}$, $\ell(\theta;\mathbf{x})$ are convex, continuously differentiable, $0 \le \ell(\theta;\mathbf{x}) < M < \infty$, and $\ell(\theta;\mathbf{x})$ and the norm of $\nabla\ell(\theta;\mathbf{x})$ are L_1-smooth and L_2-Lipschitz, respectively. For all $k \in [0, K]$, $\|\theta_k\| \le \kappa$.

Assumption 1 implies the standard convexity and smoothness of the loss function. We provide a formal definition of convexity and smoothness in Appendix A. Also, the parameterized model norm is bounded. Building on the CVaR minimization analysis of Meng and Gower (2023), which relies on the stochastic model-based framework of Davis and Drusvyatskiy (2019), we have the following convergence property.

Theorem 1 (Convergence Rate)**.** *Suppose that Assumption 1 holds and over iterations $k = 1, \dots, K+1$, Algorithm 1 uses realizations $\mathbf{x}_k$ that are i.i.d. with q. Let $\{\phi_k\}$ be the iterates generated by Algorithm 1 such that $\phi_k = (\theta_k, \alpha_k)^\top$, ϕ^* is a minimizer of F_β, and set $\lambda_k = \frac{\lambda}{\sqrt{K+1}}$. For a given quantile β, we have*

$$\mathbb{E}\left[F_\beta\left(\frac{1}{K+1}\sum_{t=1}^{K+1}\phi_k\right) - F_\beta(\phi^*)\right] \le \frac{\|(\theta_0, Z)^\top - \phi^*\|^2}{2\lambda\sqrt{K+1}} + \frac{\lambda\hat{v}(q)}{\sqrt{K+1}}, \tag{7}$$

where $\hat{v}(q) = \mathbb{E}_{\mathbf{X}^p\sim p}\left[\frac{w^*(\mathbf{X}^p)^2}{(1-\beta)^2}\frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)} + 1\right]$ *and* $w^*(\mathbf{x}) = \left((\sqrt{2L_1\ell(\theta_0;\mathbf{x})} + 2L_2\kappa))^2 + 1\right)^{1/2}$.

Remark 1 (Loss-dependent Optimization Noise)**.** Theorem 1 establishes an $\mathcal{O}(1/\sqrt{K})$ convergence rate with a stochastic noise term $\hat{v}(q)$ that depends on the initial loss $\ell(\theta_0;\mathbf{x})$. This dependence on the loss value is well-aligned with the standard results in stochastic optimization (Zhao and Zhang, 2015; Davis and Drusvyatskiy, 2019), where the stochastic noise is governed by the *gradient of the loss*. To understand this relationship more precisely, consider the case where the loss function satisfies a Polyak–Łojasiewicz (PL) condition (Karimi et al., 2016). That is, for some $\mu > 0$ and all θ, $2\mu\left(\ell(\theta_0;\mathbf{x}) - \ell^*\right) \le \|\nabla_\theta\ell(\theta_0;\mathbf{x})\|^2$ where $\ell^* = \min_\theta \ell(\theta;\mathbf{x})$. Note that the PL condition holds for several classes of neural networks

(Liu et al., 2019; Zhou and Liang, 2017; Charles and Papailiopoulos, 2018; Hardt and Ma, 2016). Under L_1-smoothness, $\|\nabla_\theta \ell(\theta_0; \mathbf{x})\|^2 \leq 2L_1 \ell(\theta_0; \mathbf{x})$ and we have

$$2\mu \left(\ell(\theta_0; \mathbf{x}) - \ell^*\right) \leq \|\nabla_\theta \ell(\theta_0; \mathbf{x})\|^2 \leq 2L_1 \ell(\theta_0; \mathbf{x}). \tag{8}$$

This chain of inequalities implies that the *high-loss samples contribute proportionately to the norm of the gradient*.

Remark 2 (Noise Reduction via Importance Sampling)**.** The term $\hat{v}(q)$ in Theorem 1 is minimized when the sampling distribution $q(\mathbf{x})$ is chosen as $q(\mathbf{x}) \propto w^*(\mathbf{x})p(\mathbf{x})$, which depends on the quantities, potentially impractical to compute in real-world scenarios. To circumvent this, we propose using a non-decreasing surrogate weighting function $\varphi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ that approximates the behavior of desired importance weights. Specifically, sampling from the distribution $q(\mathbf{x}) \propto \varphi(\ell(\theta_0; \mathbf{x}))\, p(\mathbf{x})$ reduces the term $\hat{v}(q)$ relative to naive sampling from $p(\mathbf{x})$ under the following condition:

$$\hat{v}(p) \geq \hat{v}(q) \iff \mathbb{E}[w^*(\mathbf{X}^p)^2] \geq \mathbb{E}\left[\frac{w^*(\mathbf{X}^p)^2}{\varphi(\ell(\theta_0; \mathbf{X}^p))}\right] \cdot \mathbb{E}[\varphi(\ell(\theta_0; \mathbf{X}^p))]. \tag{9}$$

We observe that simple choices of φ—such as the identity function—yield strong performance empirically (Sec. 5). In summary, the analysis establishes that loss-guided importance sampling based on the pretrained model can reduce the error of CVaR optimization. Rather than merely increasing the sample size, we leverage score-based generative models to synthesize samples from the proposed reweighted distribution, enabling efficient risk-averse training.

5 Experiments

Evaluation Summary. We evaluate the effectiveness of our proposed framework across both synthetic and real-world tasks. Specifically, we aim to answer the following questions: *(i) Can we generate high-loss-inducing samples using score-based generative models by pretrained models? (ii) Do these samples improve downstream robustness relative to existing robust optimization methods? (iii) Does the approach generalize to high-stakes, real-world applications?*

To this end, we conduct two sets of experiments: Sec. 5.1: We evaluate our method on a controlled regression task over a Gaussian mixture distribution to assess robustness under data heterogeneity and sample scarcity. Sec. 5.2: We apply our method to a real-world wireless channel state information (CSI) compression task, demonstrating its potential practical utility.

Baselines and Fairness. We compare against strong risk-sensitive and robust optimization baselines: Stochastic Subgradient Method (SSGM) for CVaR minimization (Meng and Gower, 2023), DORO (Zhai et al., 2021), χ^2-DRO (Namkoong and Duchi, 2016), and standard ERM (i.e., CVaR at $\beta = 0$). All methods start from the same pretrained checkpoint and are trained on the same number of samples from the same generative model; RAMIS uses the identical budget but replaces standard samples with loss-guided (importance) samples. Running SSGM without importance sampling isolates the contribution of our loss-guided sampling scheme, while comparisons to DORO and χ^2-DRO test whether state-of-the-art robust objectives can mitigate tail risk absent our mechanism. ERM serves as a conventional average-risk baseline.

5.1 Risk-Averse Regression over Density-Heterogeneous Gaussians

Task Overview. We consider a synthetic regression task where inputs $\mathbf{x} \in \mathbb{R}^2$ are drawn from a Gaussian mixture distribution with three components centered at $(-0.6, 0.6)$, $(0.6, 0.6)$, and $(0.0, -0.6)$, each with standard deviation 0.06 but with unbalanced mixing weights of 0.001, 0.01, and 0.989, respectively (See leftmost subplot in Figure 2). The objective is to predict the second coordinate $\mathbf{x}_{[1]}$ from the first $\mathbf{x}_{[0]}$ using a quadratic regression model trained via risk minimization methods with Mean Squared Error (MSE) loss. This setup can pose a significant challenge for robust learning due to *the extreme imbalance*: standard training on limited number of samples from $p(\mathbf{x})$ yields poor performance on rare but critical components (e.g., the 0.01 or 0.001-weight modes), which dominate the tail risk.

To evaluate our method, we adopt a two-phase training strategy. First, we obtain a pretrained (reference) model θ_0 on 2×10^2 samples from $p(\mathbf{x})$. Second, we use the loss values $\ell(\theta_0; \mathbf{x})$ itself to construct an importance-weighted distribution $q(\mathbf{x}) \propto \ell(\theta_0; \mathbf{x})\, p(\mathbf{x})$ and draw the same number of

Table 1: CVaR (mean $\pm$ std) across Quantile Levels β (lower is better).

β	RAMIS (ours)	SSGM	DORO	χ^2-DRO	ERM
0.99	**0.0723 $\pm$ 0.0428**	0.3516 $\pm$ 0.4008	0.4543 $\pm$ 0.4861	0.4819 $\pm$ 0.5279	0.3550 $\pm$ 0.3983
0.95	**0.0235 $\pm$ 0.0089**	0.0552 $\pm$ 0.0456	0.0618 $\pm$ 0.0293	0.0908 $\pm$ 0.0613	0.0684 $\pm$ 0.0516
0.90	**0.0152 $\pm$ 0.0044**	0.0279 $\pm$ 0.0170	0.0381 $\pm$ 0.0137	0.0526 $\pm$ 0.0325	0.0405 $\pm$ 0.0243
0.80	**0.0098 $\pm$ 0.0022**	0.0177 $\pm$ 0.0111	0.0213 $\pm$ 0.0081	0.0291 $\pm$ 0.0133	0.0248 $\pm$ 0.0116
0.50	**0.0050 $\pm$ 0.0008**	0.0078 $\pm$ 0.0037	0.0099 $\pm$ 0.0032	0.0098 $\pm$ 0.0024	0.0130 $\pm$ 0.0054

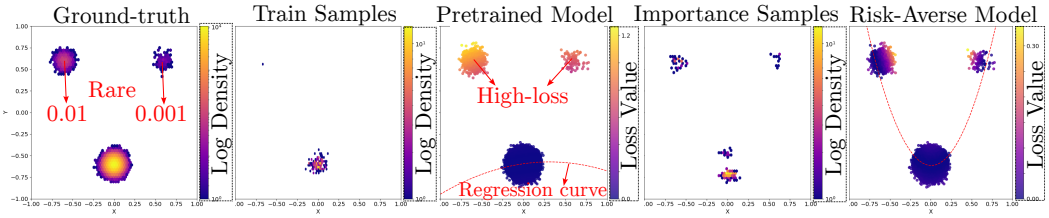

Figure 2: **Visualization of the process.** From left to right: (1) true data distribution $p(\mathbf{x})$, (2) samples drawn from $p(\mathbf{x})$, (3) pretrained model and its loss map, (4) samples drawn from $q(\mathbf{x}) \propto \ell(\theta_0; \mathbf{x})p(\mathbf{x})$, and (5) the final risk-averse model trained on these samples. The loss-guided sampling expands support to rare regions and enables robust optimization.

new samples from this distribution using the corresponding generative model. These samples are then used to train a risk-averse model. More detailed setup and results are provided in Appendix C.

Results. Figure 2 visually illustrates our method. The leftmost panel shows the ground-truth distribution $p(\mathbf{x})$, which includes two rare Gaussian components with low probabilities (0.01 and 0.001). As depicted in the second panel, a limited number of samples drawn from $p(\mathbf{x})$ rarely cover these low-density regions, resulting in limited exposure during training. Consequently, the pretrained model trained on these samples shows high loss to the rare (tail) regions, as reflected in the loss map shown in the third panel–in the colormap, these tail regions appear in yellowish hues, indicating higher loss.

We exploit this loss landscape by constructing an importance-weighted distribution based on $\ell(\theta_0; \mathbf{x})$ and guiding the generative model to sample accordingly. The fourth panel shows samples generated from this reweighted distribution. Despite using the same sample budget (2×10^2), these samples provide *substantially better coverage of the support set, especially in the tails*. The final panel shows that training on these importance samples leads to a risk-averse model that performs reliably across both high-density and tail regions of the input space.

Table 1 reports the CVaR performance across varying quantile levels β for our method and baseline approaches. Across all risk levels, our framework (RAMIS) consistently achieves the lowest CVaR, demonstrating superior robustness in tail-risk regimes. Among the baselines, SSGM performs second-best, while other robust optimization methods such as DORO and χ^2-DRO, as well as ERM, exhibit significantly higher risk. These results highlight that access to high-loss-inducing importance samples generated via pretrained model guidance provides a distinct advantage that cannot be matched by applying robust optimization techniques over uniformly sampled data.

5.2 Risk-Averse Compression of Wireless Channel State Information

Task Overview. In wireless communication systems, *Channel State Information* (CSI) captures key physical-layer characteristics such as signal directionality, multipath components, and propagation strength between transmitters and receivers (Lin, 2022). Accurate CSI feedback from the transmitter to the receiver is crucial for tasks like beamforming, scheduling, and adaptive modulation. However, modern CSI matrices are typically high-dimensional, necessitating efficient compression to support bandwidth-constrained channel feedback (Guo et al., 2022). To ensure reliable communication in practical deployments, especially under worst-case scenarios, *risk-averse* compression is essential.

In this experiment, we assess the performance of the proposed method in the context of risk-averse CSI compression. We assume access to a pretrained score-based generative model trained on a CSI dataset generated by the Quadriga simulator (Jaeckel et al., 2021), where a single CSI instance is a 256×32 complex matrix, and a baseline CSI compressor trained using ERM. The CSI compressor is implemented as a vector-quantized autoencoder (Van Den Oord et al., 2017), comprising an encoder, a quantization bottleneck, and a decoder. Following Algorithm 1, we guide the pretrained generative

Table 2: CSI Compression CVaR (mean $\pm$ std), in units of 10^{-3} (lower is better).

β	RAMIS (ours)	SSGM	DORO	χ^2-DRO	ERM
0.99	**2.10 $\pm$ 0.0028**	2.21 $\pm$ 0.0341	2.28 $\pm$ 0.0032	2.50 $\pm$ 0.2330	2.22 $\pm$ 0.0016
0.95	**1.39 $\pm$ 0.0085**	1.44 $\pm$ 0.0075	1.46 $\pm$ 0.0205	1.52 $\pm$ 0.0014	1.44 $\pm$ 0.0019
0.90	**1.07 $\pm$ 0.0047**	1.12 $\pm$ 0.0113	1.10 $\pm$ 0.0083	1.14 $\pm$ 0.0022	1.10 $\pm$ 0.0018
0.80	**0.76 $\pm$ 0.0067**	**0.76 $\pm$ 0.0026**	**0.76 $\pm$ 0.0052**	0.78 $\pm$ 0.0252	**0.76 $\pm$ 0.0013**
0.50	0.39 $\pm$ 0.0018	**0.38 $\pm$ 0.0006**	0.39 $\pm$ 0.0249	**0.38 $\pm$ 0.0260**	**0.38 $\pm$ 0.0003**

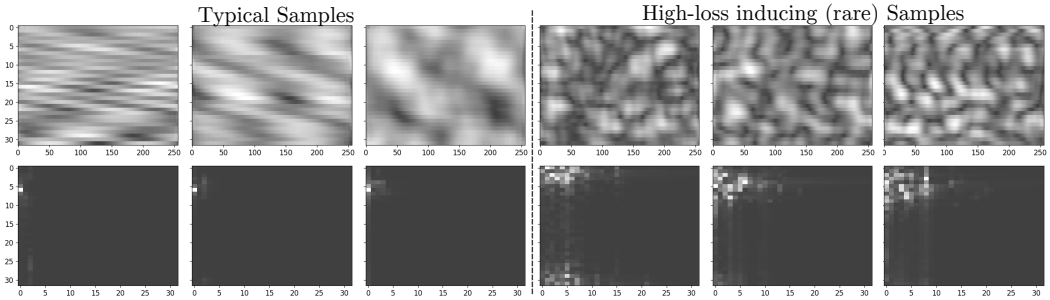

Figure 3: **Visualization of generated samples.** Left three columns: Typical samples with median loss values sampled from the base distribution. Right three columns: High-loss samples generated by pretrained model loss-guided sampling, which exhibit a 6.2×10^{-3} reconstruction loss—***rare and unseen*** across 8×10^4 samples from the base distribution. Top row: Spatial-frequency (Y/X) representation; Bottom row: Angular-delay (Y/X) representation.

model using the *MSE loss values of the initial compressor* to generate informative, high-loss samples. These samples are then used to fine-tune the compressor using a CVaR-based objective. Detailed specifications of the dataset, model architecture, and training parameters are provided in Appendix D.

Results. Table 2 reports the CVaR performance in terms of reconstruction distortion (MSE) across various quantile levels β. Lower distortion indicates better robustness. RAMIS consistently achieves the lowest CVaR in the high-risk regime ($\beta \in \{0.9, 0.95, 0.99\}$), outperforming all baselines, including SSGM, DORO, χ^2-DRO, and ERM. As β decreases toward 0.5, where CVaR approaches to the expected loss, the performance gap narrows and RAMIS converges with SSGM and ERM. The performance gain at the high β region supports that the conventional methods, which rely on samples drawn from the original data distribution, are insufficient for minimizing tail risk.

Figure 3 further illustrates the nature of the samples generated via pretrained-loss-guided importance sampling. The top row shows representations in the spatial-frequency domain, while the bottom row visualizes the corresponding angular-delay profiles, computed via 2D inverse FFT (IFFT) with truncation to the low-delay region for interpretability. The left three columns present typical generated samples from the base generative model, chosen as the three median distortion examples by MSE. Across this set, the worst observed reconstruction error was 1.10×10^{-3}.

By contrast, the right three columns show samples obtained via RAMIS, using the generative model guided by the pretrained model loss. These samples exhibit significantly higher reconstruction distortion, with the average MSE increasing by 4.99×10^{-4} (from 9.95×10^{-5}) and a maximum distortion exceeding 6.23×10^{-3}. Notably, the corresponding angular-delay representations reveal more complex scattering patterns, indicating that the proposed framework successfully targets rare, high-loss scenarios that are otherwise underrepresented in the base distribution.

5.3 Additional Analysis

To further assess the efficiency of the proposed framework, we analyze (i) the computational cost of generating importance-weighted samples and (ii) the effect of the weighting function φ, which controls the emphasis placed on high-loss regions. Detailed results and ablation studies are provided in Appendix E.

6 Discussion, Limitations of Work, and Future Directions

This paper introduces RAMIS, a novel risk-averse learning framework that integrates score-based generative modeling with pretrained model feedback to synthesize high-loss, informative samples for downstream optimization. In contrast to existing generative approaches that aim to increase sample diversity or generalization, our framework targets the *utility* of samples specifically for minimizing tail-risk objectives such as CVaR. By leveraging pretrained loss signals as importance guidance, we enable generative models to contribute directly to risk-sensitive training.

Our study primarily focuses on the theoretical motivation behind loss-guided generative sampling and demonstrates its effectiveness through controlled synthetic experiments and a domain-specific application in wireless communication. While these results validate the core principles of RAMIS, broader applications remain to be explored. As diffusion-based generative models continue to evolve and become increasingly accessible across domains where risk-sensitive optimization is critical, we expect RAMIS to generalize naturally to these settings.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant Nos. 2148224 and 2443857, the ARO Award W911NF2310062, and the ONR Award N000142412542, and is supported in part by funds from OUSD R&E, NIST, and industry partners as specified in the Resilient & Intelligent NextG Systems (RINGS) program and the WNCG/6G@UT.

References

Mohamadreza Ahmadi, Xiaobin Xiong, and Aaron D Ames. Risk-averse control via CVaR barrier functions: Application to bipedal robot locomotion. *IEEE Control Systems Letters*, 6:878–883, 2021.

Siddharth Alexander, Thomas F Coleman, and Yuying Li. Minimizing CVaR and VaR for a portfolio of derivatives. *Journal of Banking & Finance*, 30(2):583–605, 2006.

Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12 (3):313–326, 1982.

Fredrik Andersson, Helmut Mausser, Dan Rosen, and Stanislav Uryasev. Credit risk optimization with conditional value-at-risk criterion. *Mathematical programming*, 89:273–291, 2001.

Sravan Kumar Ankireddy, Heasung Kim, and Hyeji Kim. Residual diffusion models for variable-rate joint source channel coding of MIMO CSI. *arXiv preprint arXiv:2505.21681*, 2025.

Behnaz Bahmei, Elina Birmingham, and Siamak Arzanpour. CNN-RNN and data augmentation using deep convolutional generative adversarial network for environmental sound classification. *IEEE Signal Processing Letters*, 29:682–686, 2022.

Olivier Bardou, Noufel Frikha, and Gilles Pages. Computing var and CVaR using stochastic approximation and adaptive unconstrained importance sampling. 2009.

Qiuyu Cai, Chao Dong, and Kai Niu. Attention model for massive MIMO CSI compression feedback and recovery. In *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–5. IEEE, 2019.

Timothy CY Chan, Houra Mahmoudzadeh, and Thomas G Purdie. A robust-CVaR optimization approach with application to breast cancer therapy. *European Journal of Operational Research*, 238(3):876–885, 2014.

Zachary Charles and Dimitris Papailiopoulos. Stability and generalization of learning algorithms that converge to global optima. In *International conference on machine learning*, pages 745–754. PMLR, 2018.

Sapana Chaudhary, Ujwal Dinesha, Dileep Kalathil, and Srinivas Shakkottai. Risk-averse fine-tuning of large language models. *Advances in Neural Information Processing Systems*, 37:107003–107038, 2025.

Yunhao Chen, Zihui Yan, and Yunjie Zhu. A comprehensive survey for generative data augmentation. *Neurocomputing*, page 128167, 2024.

Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021.

Yinlam Chow and Mohammad Ghavamzadeh. Algorithms for CVaR optimization in MDPs. *Advances in neural information processing systems*, 27, 2014.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2022.

Erik Dahlman, Stefan Parkvall, and Johan Skold. *4G: LTE/LTE-advanced for mobile broadband*. Academic press, 2013.

Abdelaati Daouia, Irène Gijbels, and Gilles Stupfler. Extremiles: A new perspective on asymmetric least squares. *Journal of the American Statistical Association*, 114(527):1366–1381, 2019.

Abdelaati Daouia, Irene Gijbels, and Gilles Stupfler. Extremile regression. *Journal of the American Statistical Association*, 117(539):1579–1586, 2022.

Damek Davis and Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

Anand Deo and Karthyek Murthy. Efficient black-box importance sampling for var and CVaR estimation. In *2021 Winter Simulation Conference (WSC)*, pages 1–12. IEEE, 2021.

Ayoub El Hanchi and David Stephens. Adaptive importance sampling for finite-sum optimization and sampling with decreasing step-sizes. *Advances in Neural Information Processing Systems*, 33:15702–15713, 2020.

Carlo Filippi, Gianfranco Guastaroba, and Maria Grazia Speranza. Conditional value-at-risk beyond finance: a survey. *International Transactions in Operational Research*, 27(3):1277–1319, 2020.

Jiajia Guo, Chao-Kai Wen, Shi Jin, and Geoffrey Ye Li. Overview of deep learning-based CSI feedback in massive MIMO systems. *IEEE Transactions on Communications*, 70(12):8017–8045, 2022.

Moritz Hardt and Tengyu Ma. Identity matters in deep learning. *arXiv preprint arXiv:1611.04231*, 2016.

Shengyi He, Guangxin Jiang, Henry Lam, and Michael C Fu. Adaptive importance sampling for efficient stochastic root finding and quantile estimation. *Operations Research*, 72(6):2612–2630, 2024a.

Tianyu He, Peiyi Han, Shaoming Duan, Zirui Wang, Wentai Wu, Chuanyi Liu, and Jianrun Han. Generative data augmentation with differential privacy for non-iid problem in decentralized clinical machine learning. *Future Generation Computer Systems*, 160:171–184, 2024b.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Stephan Jaeckel, Leszek Raschkowski, Kai Borner, Lars Thiele, and Frank Burkhardt andErnst Berlein. Quadriga - quasi deterministic radio channel generator, user manual and documentation. *Fraun-hofer Heinrich Hertz Institute*, 2.6.1, 2021.

Stephan Jaeckel et al. Quadriga: A 3-d multi-cell channel model with time evolution for enabling virtual field trials. *IEEE transactions on antennas and propagation*, 62(6):3242–3256, 2014.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 795–811. Springer, 2016.

Heasung Kim, Taekyun Lee, Hyeji Kim, and Gustavo de Veciana. Importance sampling via score-based generative models. *arXiv preprint arXiv:2502.04646*, 2025a.

Heasung Kim, Taekyun Lee, Hyeji Kim, Gustavo De Veciana, Mohamed Amine Arfaoui, Asil Koc, Phil Pietraski, Guodong Zhang, and John Kaewell. Generative diffusion model-based compression of MIMO CSI. In *ICC 2025 - IEEE International Conference on Communications*, pages 6323–6328, 2025b.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *The Thirteenth International Conference on Learning Representations*, 2025c.

Taekyun Lee, Juseong Park, Hyeji Kim, and Jeffrey G Andrews. Generating high dimensional user-specific wireless channels using diffusion models. *IEEE Transactions on Wireless Communications*, 2025.

Xingqin Lin. An overview of 5G advanced evolution in 3GPP release 18. *IEEE Communications Standards Magazine*, 6(3):77–83, 2022.

Mingrui Liu, Zhuoning Yuan, Yiming Ying, and Tianbao Yang. Stochastic AUC maximization with deep neural networks. *arXiv preprint arXiv:1908.10831*, 2019.

Yusha Liu and Osvaldo Simeone. Hyperrnn: Deep learning-aided downlink CSI acquisition via partial channel reciprocity for fdd massive mimo. In *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 31–35. IEEE, 2021.

Zhilin Lu, Jintao Wang, and Jian Song. Multi-resolution CSI feedback with deep learning in massive MIMO system. In *IEEE International Conference on Communications*, pages 1–6. IEEE, 2020.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022.

Si Yi Meng and Robert M Gower. A model-based method for minimizing CVaR and beyond. In *International Conference on Machine Learning*, pages 24436–24456. PMLR, 2023.

Hongseok Namkoong and John C Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. *Advances in neural information processing systems*, 29, 2016.

Deanna Needell, Rachel Ward, and Nati Srebro. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. *Advances in neural information processing systems*, 27, 2014.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

R Tyrrell Rockafellar and Stanislav Uryasev. Optimization of Conditional Value-at-Risk. *Journal of risk*, 2: 21–42, 2000.

Siyu Shao, Pu Wang, and Ruqiang Yan. Generative adversarial networks for data augmentation in machine fault diagnosis. *Computers in Industry*, 106:85–93, 2019.

C Shivashankar and Shane Miller. Semantic data augmentation with generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 863–873, 2023.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Aviv Tamar, Yonatan Glassner, and Shie Mannor. Optimizing the CVaR via sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2015.

Zhongfu Tan, Guan Wang, Liwei Ju, Qingkun Tan, and Wenhai Yang. Application of CVaR risk aversion approach in the dynamical scheduling optimization model for virtual power plant connected with wind-photovoltaic-energy storage system with uncertainties and demand response. *Energy*, 124:198–213, 2017.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*, 2023.

Dylan Troop, Frédéric Godin, and Jia Yuan Yu. Bias-corrected peaks-over-threshold estimation of the CVaR. In *Uncertainty in Artificial Intelligence*, pages 1809–1818. PMLR, 2021.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

Tianqi Wang, Chao-Kai Wen, Shi Jin, and Geoffrey Ye Li. Deep learning-based CSI feedback approach for time-varying massive MIMO channels. *IEEE Wireless Communications Letters*, 8(2):416–419, 2018.

Yinong Oliver Wang, Younjoon Chung, Chen Henry Wu, and Fernando De la Torre. Domain gap embeddings for generative dataset augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28684–28694, 2024a.

Zaitian Wang, Pengfei Wang, Kunpeng Liu, Pengyang Wang, Yanjie Fu, Chang-Tien Lu, Charu C Aggarwal, Jian Pei, and Yuanchun Zhou. A comprehensive survey on data augmentation. *arXiv preprint arXiv:2405.09591*, 2024b.

Chao-Kai Wen, Wan-Ting Shih, and Shi Jin. Deep learning for massive MIMO CSI feedback. *IEEE Wireless Communications Letters*, 7(5):748–751, 2018.

Linna Xu and Yongli Zhu. Generative modeling and data augmentation for power system production simulation. In *NeurIPS 2024 Workshop on Data-driven and Differentiable Simulations, Surrogates, and Solvers*, 2024.

Wei Yang, Chi Lin, Haipeng Dai, Pengfei Wang, Jiankang Ren, Lei Wang, Guowei Wu, and Qiang Zhang. Robust wireless rechargeable sensor networks. *IEEE/ACM Transactions on Networking*, 31(3):949–964, 2022.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23174–23184, 2023.

Ali Zaidi, Fredrik Athley, Jonas Medbo, Ulf Gustavsson, Giuseppe Durisi, and Xiaoming Chen. *5G Physical Layer: principles, models and technology components*. Academic Press, 2018.

Runtian Zhai, Chen Dan, Zico Kolter, and Pradeep Ravikumar. Doro: Distributional and outlier robust optimization. In *International Conference on Machine Learning*, pages 12345–12355. PMLR, 2021.

Peilin Zhao and Tong Zhang. Stochastic optimization with importance sampling for regularized loss minimization. In *international conference on machine learning*, pages 1–9. PMLR, 2015.

Chenyu Zheng, Guoqiang Wu, and Chongxuan Li. Toward understanding generative data augmentation. *Advances in neural information processing systems*, 36:54046–54060, 2023.

Yi Zhou and Yingbin Liang. Characterization of gradient dominance and regularity conditions for neural networks. *arXiv preprint arXiv:1710.06910*, 2017.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: It provides the full set of assumptions and a complete and correct proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes, the implementation details are provided along with source code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Yes, we provide source code. At submission time, we release anonymized versions.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies all the implementation details in the appendices.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Yes. We provide the standard deviation of the performance under multiple random seeds.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes. This work provides sufficient information regarding the computational resources for the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes. This work complies with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no specific societal impact of the work.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper does not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes. All external assets are properly credited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Yes. The implementation code is clearly documented and released with the paper.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The main method development in this research does not involve LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Contents

Table 3: Notation and Description

Notation	Description	Note
$\mathbf{X}$	Data sample	Random variable
$\mathbf{X}^p_{(t)}$	State of the diffusion at time t started from p	$\mathbf{X}^p_{(0)} \sim p$
$\mathbf{x}$	Realization of $\mathbf{X}$	$\mathbf{x} \in \mathbb{R}^{d_1}$
$p(\mathbf{x})$	Base data distribution	
$q(\mathbf{x})$	Importance sampling distribution	
θ	Parameters of task model	$\theta \in \mathbb{R}^{d_2}$
θ_0	Pretrained reference model	
$\ell(\theta; \mathbf{x})$	Loss on input $\mathbf{x}$ for model θ	
β	CVaR confidence level	
α	Value-at-Risk (VaR) threshold	
$F_\beta(\theta, \alpha)$	Surrogate CVaR objective	
$\varphi(\cdot)$	Weighting function	Non-decreasing, $\varphi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$
Z	Normalization constant	$Z = \mathbb{E}_p[\varphi(\ell(\theta_0; \mathbf{X}^p))]$
K	Total number of training iterations	
λ_k	Learning rate at iteration k	
ϕ_k	Joint variable (θ_k, α_k) at iteration k	
ϕ^*	Optimal joint variable	
$\hat{v}(q)$	Noise term in convergence bound	
κ	Bound on parameter norm $\|\theta_k\|$	$\kappa < \infty$
$\mathcal{B}$	Dataset of i.i.d. samples from $p(\mathbf{x})$	

A Technical Results

Definition 1. A differentiable function f is convex if $f(\theta') \geq f(\theta) + \langle \nabla f(\theta), \theta' - \theta \rangle, \forall \theta', \theta$.

Definition 2. For a given $L > 0$, a differentiable function f is L-smooth if $\|\nabla f(\theta') - \nabla f(\theta)\| \leq L\|\theta' - \theta\|, \forall \theta', \theta$.

A.1 Proof of Theorem 1

The proof builds upon the stochastic subgradient method analysis for CVaR minimization developed in Meng and Gower (2023), which itself extends the model-based optimization framework of stochastic convex optimization presented in Davis and Drusvyatskiy (2019).

We consider the unconstrained formulation of the CVaR minimization problem given by

$$(\theta^*, \alpha^*) = \arg \min_{\theta \in \mathbb{R}^{d_2}, \alpha \in \mathbb{R}} F_\beta(\theta, \alpha),$$

where the objective function is represented as

$$F_\beta(\theta, \alpha) = \alpha + \frac{1}{1-\beta} \mathbb{E}_{\mathbf{X}^q \sim q} \left[\frac{p(\mathbf{X}^q)}{q(\mathbf{X}^q)} (\ell(\theta; \mathbf{X}^q) - \alpha)^+ \right],$$

with $(z)^+ = \max(z, 0)$ denoting the positive-part operator.

Consider a realization $\mathbf{x}$ sampled from distribution q. Then, the subgradients of $F_\beta(\theta, \alpha; \mathbf{x})$, i.e., the objective value from a realization $\mathbf{x}$, with respect to θ and α are given by

$$\partial_\theta F_\beta(\theta, \alpha; \mathbf{x}) = \frac{1}{1-\beta} \frac{p(\mathbf{x}) \mathbf{1}_{\ell(\theta;\mathbf{x}) > \alpha} \nabla_\theta \ell(\theta; \mathbf{x})}{q(\mathbf{x})}, \tag{10}$$

$$\partial_\alpha F_\beta(\theta, \alpha; \mathbf{x}) = 1 - \frac{1}{1-\beta} \frac{p(\mathbf{x}) \mathbf{1}_{\ell(\theta;\mathbf{x}) > \alpha}}{q(\mathbf{x})}. \tag{11}$$

Accordingly, the stochastic subgradient updates at iteration k are expressed as:

$$\theta_{k+1} = \theta_k - \lambda_k \cdot \frac{1}{1-\beta} \cdot \frac{p(\mathbf{x}_k)}{q(\mathbf{x}_k)} \cdot \mathbf{1}_{\ell(\theta_k;\mathbf{x}_k) > \alpha_k} \cdot \nabla_\theta \ell(\theta_k; \mathbf{x}_k), \tag{12}$$

$$\alpha_{k+1} = \alpha_k - \lambda_k \cdot \left(1 - \frac{1}{1-\beta} \cdot \frac{p(\mathbf{x}_k)}{q(\mathbf{x}_k)} \cdot \mathbf{1}_{\ell(\theta_k;\mathbf{x}_k) > \alpha_k} \right), \tag{13}$$

where λ_k denotes the step size at iteration k. The updates in (12)-(13) correspond to the procedure $\texttt{SubGradientDescent}(\partial F_\beta, \theta_k, \alpha_k)$ in Algorithm 1.

To analyze convergence, we introduce a linearization model, i.e., the stochastic one-sided model, centered at the current iterate as

$$f_{\phi_k}(\phi, \mathbf{x}) = \alpha_k + \frac{1}{1-\beta} \cdot \frac{p(\mathbf{x})}{q(\mathbf{x})} \cdot (\ell(\theta_k; \mathbf{x}) - \alpha_k)^+ + \langle g_k, \phi - \phi_k \rangle \tag{14}$$

where $g_k \in \partial F_\beta(\phi_k; \mathbf{x})$, $\phi_k = \begin{pmatrix} \theta_k \\ \alpha_k \end{pmatrix}$, and $\phi = \begin{pmatrix} \theta \\ \alpha \end{pmatrix}$. The update step is then equivalently expressed as

$$(\theta_{k+1}, \alpha_{k+1}) = \arg\min_{\theta,\alpha} f_{\phi_k}(\phi, \mathbf{x}) + \frac{1}{2\lambda_k} \|\phi - \phi_k\|^2.$$

Under this formulation, the convergence behavior of the algorithm can be analyzed via the theoretical framework of model-based stochastic subgradient methods. In particular, under the following assumptions **(B1)–(B4)** involving sample accessibility, one-sided accuracy, weak convexity, and Lipschitz continuity, the method achieves a convergence rate of $\mathcal{O}(1/\sqrt{K})$ after K iterations (Meng and Gower, 2023; Davis and Drusvyatskiy, 2019).

Our analysis uses this framework in the CVaR minimization setting with the fixed importance sampling distribution. Specifically, we consider a sampling distribution $q(\mathbf{x})$ that is constructed a priori based on an initial loss evaluation and a task-dependent importance weight function. This extension allows the optimization to benefit from variance reduction while preserving the convergence guarantees of stochastic model-based methods. We next verify assumptions **(B1)–(B4)**.

(B1) Sampling. It is possible to generate i.i.d. realizations $\mathbf{x}_1, \mathbf{x}_2, \ldots \sim q$. This condition is satisfied by the underlying assumption of Theorem 1.

(B2) One-sided accuracy. There exists $\zeta \in \mathbb{R}$ and there is an open convex set U containing the domain and a measurable function $(\phi_k, \phi, \mathbf{x}) \mapsto f_{\phi_k}(\phi; \mathbf{x})$, defined on $U \times U \times \Omega$, satisfying

$$\mathbb{E}_{\mathbf{X}^q \sim q}[f_{\phi_k}(\phi_k; \mathbf{X}^q)] = F_\beta(\theta_k, \alpha_k) \qquad \forall \phi_k \in U, \tag{15}$$

and

$$\mathbb{E}_{\mathbf{X}^q \sim q}[f_{\phi_k}(\phi; \mathbf{X}^q) - F_\beta(\theta, \alpha)] \leq \frac{\zeta}{2} \|\phi - \phi_k\|^2, \tag{16}$$

where Ω is the sample space.

The equality (15) holds due to the definition of $f_{\phi_k}(\phi, \mathbf{x})$. Moreover, $\mathbb{E}_{\mathbf{X}^q \sim q}[f_{\phi_k}(\phi; \mathbf{X}^q) - F_\beta(\theta, \alpha)] = F_\beta(\theta_k, \alpha_k) - F_\beta(\theta, \alpha) + \mathbb{E}_{\mathbf{X}^q \sim q}[\langle g_k, \phi - \phi_k \rangle] \leq 0$ by the convexity of $F_\beta(\theta, \alpha)$ with respect to ϕ, indicating $\zeta = 0$.

(B3) Weak convexity. $f_{\phi_k}(\phi; \mathbf{x})$ is convex for all ϕ_k, a.e. $\mathbf{x} \in \Omega$. This holds by the linearization model definition in (14).

(B4) Lipschitz property. There exist $V \in \mathbb{R}$ and a measurable function $v : \Omega \to \mathbb{R}_+$ satisfying $\sqrt{\mathbb{E}_{\mathbf{X}}[v(\mathbf{X})^2]} \leq V$ such that

$$f_{\phi_k}(\phi_k; \mathbf{x}) - f_{\phi_k}(\phi; \mathbf{x}) \leq v(\mathbf{x}) \|\phi_k - \phi\|. \tag{17}$$

To show this, we examine the one-sided model gap as follows.

$$f_{\phi_k}(\phi_k, \mathbf{x}) - f_{\phi_k}(\phi, \mathbf{x}) = \langle g_k, \phi_k - \phi_k \rangle - \langle g_k, \phi - \phi_k \rangle \leq \|g_k\| \|\phi_k - \phi\| \tag{18}$$

where g_k is the subgradient of the estimated object. The norm of the subgradient is given as follows.

$$\|g_k\|^2 = \left\| \frac{1}{1-\beta} \frac{p(\mathbf{x}_k) u_k \nabla_\theta \ell(\theta_k; \mathbf{x}_k)}{q(\mathbf{x}_k)} \right\|^2 + \left\| 1 - \frac{1}{1-\beta} \frac{p(\mathbf{x}_k) u_k}{q(\mathbf{x}_k)} \right\|^2 \tag{19}$$

$$\leq \frac{1}{(1-\beta)^2} \left\| \frac{p(\mathbf{x}_k) \nabla_\theta \ell(\theta_k; \mathbf{x}_k)}{q(\mathbf{x}_k)} \right\|^2 + 1 + \frac{p(\mathbf{x}_k)^2}{q(\mathbf{x}_k)^2 (1-\beta)^2} \tag{20}$$

where $u_k = \mathbf{1}_{\ell(\theta_k;\mathbf{x}_k)>\alpha_k}$ and the inequality holds by the subadditivity of the norm. We denote the square root of the upper bound as v as

$$v(\mathbf{x}) \coloneqq \sqrt{\frac{1}{(1-\beta)^2}\left\|\frac{p(\mathbf{x})\nabla_\theta \ell(\theta_k;\mathbf{x})}{q(\mathbf{x})}\right\|^2 + 1 + \frac{p(\mathbf{x})^2}{q(\mathbf{x})^2(1-\beta)^2}}. \tag{21}$$

This function $v(\mathbf{x})$ satisfies the pointwise Lipschitz condition (17). Furthermore, we use upper bounds on $v(\mathbf{x})$ to show the connection between the gradient of the loss and its value.

Consider the expected value of the stochastic noise as follows.

$$\sqrt{\mathbb{E}_{\mathbf{X}^q\sim q}[v(\mathbf{X}^q)^2]} = \sqrt{\mathbb{E}_{\mathbf{X}^q\sim q}\left[\frac{1}{(1-\beta)^2}\left\|\frac{p(\mathbf{X}^q)\nabla_\theta \ell(\theta_k;\mathbf{X}^q)}{q(\mathbf{X}^q)}\right\|^2 + 1 + \frac{p(\mathbf{X}^q)^2}{q(\mathbf{X}^q)^2(1-\beta)^2}\right]}. \tag{22}$$

We then have

$$\mathbb{E}_{\mathbf{X}^q\sim q}\left[\frac{1}{(1-\beta)^2}\left\|\frac{p(\mathbf{X}^q)\nabla_\theta \ell(\theta_k;\mathbf{X}^q)}{q(\mathbf{X}^q)}\right\|^2 + 1 + \frac{p(\mathbf{X}^q)^2}{q(\mathbf{X}^q)^2(1-\beta)^2}\right] \tag{23}$$

$$\leq \mathbb{E}_{\mathbf{X}^q\sim q}\left[\frac{1}{(1-\beta)^2}\left\|\frac{p(\mathbf{X}^q)(\|\nabla_\theta \ell(\theta_0;\mathbf{X}^q)\| + 2L_2\kappa)}{q(\mathbf{X}^q)}\right\|^2 + 1 + \frac{p(\mathbf{X}^q)^2}{q(\mathbf{X}^q)^2(1-\beta)^2}\right] \tag{24}$$

$$= \mathbb{E}_{\mathbf{X}^p\sim p}\left[\frac{1}{(1-\beta)^2}\frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)}\left(\|\nabla_\theta \ell(\theta_0;\mathbf{X}^p)\| + 2L_2\kappa\right)^2 + 1 + \frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)(1-\beta)^2}\right] \tag{25}$$

where the first inequality holds by the assumption that the gradient of the loss function satisfies

$$|\|\nabla \ell(\theta;\mathbf{x})\| - \|\nabla \ell(\theta';\mathbf{x})\|| \leq L_2\|\theta-\theta'\| \quad \forall \theta,\theta' \in \mathbb{R}^{d_2}, \mathbf{x}\in\mathbb{R}^{d_1}, \tag{26}$$

and the norm of the model parameter θ has a bounded value κ with $L_2\|\theta_k - \theta_0\| \leq 2L_2\kappa$. A corresponding bound also holds with L_1 in place of L_2 by the reverse triangle inequality, $|\|\nabla \ell(\theta_k;\mathbf{x})\| - \|\nabla \ell(\theta_0;\mathbf{x})\|| \leq \|\nabla \ell(\theta_k;\mathbf{x}) - \nabla \ell(\theta_0;\mathbf{x})\|$.

Moreover, L_1–smoothness and convexity yield

$$\|\nabla_\theta \ell(\theta_0;\mathbf{x})\|^2 \leq 2L_1(\ell(\theta_0;\mathbf{x}) - \ell^*) \leq 2L_1\ell(\theta_0;\mathbf{x}). \tag{27}$$

Combining this, we define $\hat{v}(q)$ such that

$$\hat{v}(q) \coloneqq \mathbb{E}_{\mathbf{X}^p\sim p}\left[\frac{1}{(1-\beta)^2}\frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)}\left(\sqrt{2L_1\ell(\theta_0;\mathbf{X}^p)} + 2L_2\kappa\right)^2 + 1 + \frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)(1-\beta)^2}\right], \tag{28}$$

which satisfies $\sqrt{\mathbb{E}_{\mathbf{X}}[v(\mathbf{X})^2]} \leq \sqrt{\hat{v}(q)}$ and we set $V = \sqrt{\hat{v}(q)}$.

To simplify the term, we introduce $w^*(\mathbf{x}) = \sqrt{\left(\sqrt{2L_1\ell(\theta_0;\mathbf{x})} + 2L_2\kappa\right)^2 + 1}$ which gives us $\hat{v}(q) = \mathbb{E}_{\mathbf{X}^p\sim p}\left[\frac{w^*(\mathbf{X}^p)^2}{(1-\beta)^2}\frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)} + 1\right]$.

The imposed conditions **(B1–B4)** allow us to directly apply the standard model-based stochastic gradient convergence analysis, as established in Theorem 4.4 of Davis and Drusvyatskiy (2019) and Theorem 5.2 of Meng and Gower (2023). This yields the following convergence bound

$$\mathbb{E}\left[F_\beta\left(\frac{1}{K+1}\sum_{t=1}^{K+1}\phi_k\right) - F_\beta(\phi^*)\right] \leq \frac{\|(\theta_0,Z)^\top - \phi^*\|^2}{2\lambda\sqrt{K+1}} + \frac{\lambda\hat{v}(q)}{\sqrt{K+1}}. \tag{29}$$

A.2 Proof of Remark 2

Recall the definition of the stochastic noise $\hat{v}(q) = \mathbb{E}_{\mathbf{X}^p\sim p}\left[\frac{w^*(\mathbf{X}^p)^2}{(1-\beta)^2}\frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)} + 1\right]$. Since the scalar constant $1/(1-\beta)^2$ and the additive term $+1$ are independent of the choice of the sampling distribution q. Consider the importance sampling distribution q such that $q(\mathbf{x}) \propto \varphi(\ell(\theta_0;\mathbf{x}))p(\mathbf{x})$.

By definition, we have

$$\hat{v}(p) = \mathbb{E}_{\mathbf{X}^p\sim p}\Big[\frac{w^*(\mathbf{X}^p)^2}{(1-\beta)^2}+1\Big] = \frac{1}{(1-\beta)^2}\mathbb{E}_{\mathbf{X}^p\sim p}\big[w^*(\mathbf{X}^p)^2\big]+1$$

and

$$\hat{v}(q) = \frac{1}{(1-\beta)^2}\mathbb{E}_{\mathbf{X}^p\sim p}\Big[\frac{w^*(\mathbf{X}^p)^2}{\varphi(\ell(\theta_0;\mathbf{X}^p))}\Big]\mathbb{E}_{\mathbf{X}^p\sim p}\big[\varphi(\ell(\theta_0;\mathbf{X}^p))\big]+1,$$

which directly gives us the equivalent condition

$$\mathbb{E}[w^*(\mathbf{X}^p)^2] \geq \mathbb{E}\left[\frac{w^*(\mathbf{X}^p)^2}{\varphi(\ell(\theta_0;\mathbf{X}^p))}\right]\cdot\mathbb{E}[\varphi(\ell(\theta_0;\mathbf{X}^p))]. \tag{30}$$

Similarly, consider the per-iteration CVaR objective as

$$F_\beta(\theta_k,\alpha_k) = \alpha_k + \frac{1}{1-\beta}\mathbb{E}_{\mathbf{X}^q\sim q}\left[\frac{p(\mathbf{X}^q)}{q(\mathbf{X}^q)}(\ell(\theta_k;\mathbf{X}^q)-\alpha_k)^+\right]. \tag{31}$$

Then, the variance of the MC estimator under q is smaller than that under the base distribution p if and only if

$$\mathbb{E}_{\mathbf{X}^p\sim p}\left[((\ell(\theta_k;\mathbf{X}^p)-\alpha_k)^+)^2\right] \geq \mathbb{E}_{\mathbf{X}^q\sim q}\left[\frac{Z}{\varphi(\ell(\theta_0;\mathbf{X}^q))}((\ell(\theta_k;\mathbf{X}^q)-\alpha_k)^+)^2\right]. \tag{32}$$

Intuitively, this condition is satisfied when high-loss examples under the reference model θ_0 tend to remain high-loss during fine-tuning, so that $((\ell(\theta_k;\mathbf{x})-\alpha_k)^+)^2$ and $\varphi(\ell(\theta_0;\mathbf{x}))$ are positively correlated. Such an assumption can be realistic in settings where high-loss inputs often persist across training iterations and require multiple optimization steps to mitigate their contribution to risk.

B Implementation of the Pretrained Model Loss-guided Sampling

This section outlines the practical implementation of loss-guided sampling using a pretrained score-based generative model, characterized by its score function $\nabla_{\mathbf{x}}\log p_t(\mathbf{x})$.

Following standard practice (Ho et al., 2020; Lugmayr et al., 2022; Nichol and Dhariwal, 2021; Choi et al., 2021), we adopt $\mathsf{f}(\mathbf{x},t) = -\frac{1}{2}\beta(t)\mathbf{x}, \quad \sigma(t) = \sqrt{\beta(t)}$, where $\beta(t)$ is a non-negative scalar-valued function satisfying $0 \leq \beta(t) \leq 1$. [2]

B.1 SDE Discretization

Given the approximated score function of the importance sampling density, $\nabla_{\mathbf{x}}\log q_t(\mathbf{x}) \approx \nabla_{\mathbf{x}}\log p_t(\mathbf{x}) + \tilde{g}(\mathbf{x},t)$ where $\tilde{g}(\mathbf{x},t)$ is given in (41), we follow the reverse-time SDE formulation to simulate the generative process. Recall that the continuous-time reverse SDE governed by the time-dependent score function $\nabla_{\mathbf{x}}\log q_t(\mathbf{x})$ is given as

$$\mathrm{d}\mathbf{X}^q_{(t)} = \left(-\frac{1}{2}\beta(t)\,\mathbf{X}^q_{(t)} - \beta(t)\nabla_{\mathbf{x}}\log q_t(\mathbf{X}^q_{(t)})\right)\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\tilde{\mathbf{W}}_{(t)}.$$

This formulation is equivalent to the reverse-time dynamics derived in Song et al. (2021) under the variance-preserving setting.

To implement the sampling process in discrete time, we adopt the DDPM-style discretization. Let $t \in \{0,\ldots,T-1\}$. We define discrete-time variance schedulers as

$$\alpha_t \coloneqq 1-\beta_t, \qquad \bar{\alpha}_t \coloneqq \prod_{s=0}^{t}\alpha_s = \prod_{s=0}^{t}(1-\beta_s). \tag{33}$$

[2] We follow conventional score-based generative model notation while avoiding clashes with CVaR parameters. CVaR: (α,β); generative model: variance-schedule parameters (α,β). Thus, (α,β) are used only for diffusion schedules, whereas (α,β) are used for denoting the VaR value and confidence level.

We use the cosine schedule (Nichol and Dhariwal, 2021) in its discrete form as

$$\nu_t = \cos^2\Big(\frac{t/T + \varepsilon_\beta}{1+\varepsilon_\beta} \cdot \frac{\pi}{2}\Big), \qquad \beta_t = 1 - \frac{\nu_{t+1}}{\nu_t}, \qquad \alpha_t = 1 - \beta_t, \tag{34}$$

with $\varepsilon_\beta = 0.008$ for numerical stability. Then the update formula is given as

$$\mathbf{x}_{(t-1)} = \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_{(t)} + \beta_t \nabla_{\mathbf{x}_{(t)}} \log q_t(\mathbf{x}_{(t)})\Big) + \sqrt{\tilde{\beta}_t}\,\mathbf{z}, \qquad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{35}$$

where $\tilde{\beta}_t = \beta_t \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}$.

For baseline methods that do not use importance samples, we utilize samples generated from the pretrained base score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ through the reverse process.

B.2 Guidance Approximation

The noise-perturbed score function of the importance sampling density q can be represented as a summation of the base score function and a guidance term g as follows.

$$\nabla_{\mathbf{x}} \log q_t(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) + g(\mathbf{x}, t), \tag{36}$$

where the guidance term $g(\mathbf{x}, t)$ is defined by

$$g(\mathbf{x}, t) \coloneqq \nabla_{\mathbf{x}} \log \mathbb{E}_{\mathbf{X}_0^p \sim p_{\mathbf{X}_0^p|\mathbf{X}_t^p}(\cdot|\mathbf{x})}\left[w(\mathbf{X}_0^p)\right], \tag{37}$$

with $w(\mathbf{X}_0^p)$ denoting the weight function as $q(\mathbf{x}) \propto w(\mathbf{x})p(\mathbf{x})$. In our setting $w(\mathbf{x}) = \varphi(\ell(\theta_0; \mathbf{x}))$; for brevity we write w throughout this section.

Since this conditional expectation in (37) is intractable in general, a first-order Taylor expansion of $w(\mathbf{X}_0^p)$ around the conditional mean can be considered. Let

$$\bar{\mathbf{x}}_0'|_{\mathbf{x},t} \coloneqq \mathbb{E}[\mathbf{X}_0^p \mid \mathbf{X}_t^p = \mathbf{x}]$$

denote the conditional mean of $\mathbf{X}_0^p$ given $\mathbf{X}_t^p = \mathbf{x}$. Linearizing the loss function around this point yields

$$w(\mathbf{X}_0^p) \approx w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t}) + \nabla w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t})^\top (\mathbf{X}_0^p - \bar{\mathbf{x}}_0'|_{\mathbf{x},t}).$$

Taking the expectation over $p_{\mathbf{X}_0^p|\mathbf{X}_t^p}$ eliminates the second term due to the zero-mean residual, giving the following approximation

$$g(\mathbf{x}, t) \approx \nabla_{\mathbf{x}} \log w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t}). \tag{38}$$

The conditional mean $\bar{\mathbf{x}}_0'|_{\mathbf{x},t}$ is represented via Tweedie's formula (Chung et al., 2022; Yu et al., 2023), which connects the posterior mean to the score function of the marginal at time t,

$$\bar{\mathbf{x}}_0'|_{\mathbf{x},t} = \frac{1}{\sqrt{\bar{\alpha}(t)}}\left(\mathbf{x} + (1-\bar{\alpha}(t))\nabla_{\mathbf{x}} \log p_t(\mathbf{x})\right), \tag{39}$$

where $\bar{\alpha}(t) = \exp\left(-\int_0^t \beta(s)ds\right)$.

We further simplify (38) by using the finite difference approximation of the Hessian (Kim et al., 2025a). Specifically, for a small step size $\epsilon > 0$, the following directional approximation holds:

$$\nabla_{\mathbf{x}} \log p_t(\mathbf{x}) + \epsilon H_{p_t}(\mathbf{x}) \nabla_{\bar{\mathbf{x}}_0'|_{\mathbf{x},t}} \log w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t}) \approx \nabla_{\mathbf{x}} \log p_t\left(\mathbf{x} + \epsilon \nabla_{\bar{\mathbf{x}}_0'|_{\mathbf{x},t}} \log w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t})\right), \tag{40}$$

which gives us

$$\begin{aligned} g(\mathbf{x}, t) \approx \tilde{g}(\mathbf{x}, t) &\coloneqq \frac{1}{\sqrt{\bar{\alpha}(t)}} \nabla_{\bar{\mathbf{x}}_0'|_{\mathbf{x},t}} \log w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t}) \\ &+ \frac{1}{\epsilon(1-\bar{\alpha}(t))^{-1}\sqrt{\bar{\alpha}(t)}} \left(\nabla_{\mathbf{x}} \log p_t\left(\mathbf{x} + \epsilon \nabla_{\bar{\mathbf{x}}_0'|_{\mathbf{x},t}} \log w(\bar{\mathbf{x}}_0'|_{\mathbf{x},t})\right) - \nabla_{\mathbf{x}} \log p_t(\mathbf{x})\right). \end{aligned} \tag{41}$$

In our applications, the approximation in (41) can yield a guidance term for $w(\mathbf{x}) = \varphi(\ell(\theta_0; \mathbf{x}))$; sampling proceeds by replacing the base score with $\nabla_{\mathbf{x}} \log q_t(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) + \tilde{g}(\mathbf{x}, t)$ in the reverse-time updates in (35).

C Experiments on Gaussian Mixtures

This experiment is designed to evaluate the performance of the proposed RAMIS framework and baseline methods in a controlled setting where the data distribution contains low-density (i.e., rare) regions that are often underrepresented in standard training regimes. Specifically, we construct a synthetic mixture-of-Gaussians distribution in which certain components contribute small probability mass compared to high-density regions. These rare components are configured to induce high loss under models trained with ERM, thereby creating a challenging testbed for assessing risk-aware generative sampling.

To evaluate the sampling behavior of the proposed method, particularly the ability to capture rare, high-loss regions, we leverage the closed-form expression of the ground-truth score function for the mixture of Gaussian distributions. This allows us to perform both accurate generative modeling and precise evaluation of coverage in the tail of the data distribution.

Score Function for Mixture of Gaussians. We consider a synthetic mixture-of-Gaussians (MoG) prior at $t = 0$:

$$p_0(\mathbf{x}) = \sum_{i=1}^{N} \pi_i \, \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i),$$

where $\pi_i \geq 0$ and $\sum_{i=1}^{N} \pi_i = 1$ are the mixture weights. Under the forward diffusion process, each Gaussian component evolves with

$$\boldsymbol{\mu}_i^t = \sqrt{\bar{\alpha}_t}\, \boldsymbol{\mu}_i, \quad \boldsymbol{\Sigma}_i^t = \bar{\alpha}_t \, \boldsymbol{\Sigma}_i + (1 - \bar{\alpha}_t)\mathbf{I}, \quad p_i(\mathbf{x} \mid t) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_i^t, \boldsymbol{\Sigma}_i^t).$$

The score function of each individual component is given by

$$\nabla_{\mathbf{x}} \log p_i(\mathbf{x} \mid t) = -\left(\boldsymbol{\Sigma}_i^t\right)^{-1} (\mathbf{x} - \boldsymbol{\mu}_i^t).$$

Let $\rho_i(\mathbf{x}, t)$ denote the posterior responsibility of the i-th component, defined as

$$\rho_i(\mathbf{x}, t) = \frac{\pi_i \, p_i(\mathbf{x} \mid t)}{\sum_{j=1}^{N} \pi_j \, p_j(\mathbf{x} \mid t)}.$$

Then, the overall score function of the mixture distribution at time t is given by the weighted average

$$\nabla_{\mathbf{x}} \log p_t(\mathbf{x}) = \sum_{i=1}^{N} \rho_i(\mathbf{x}, t) \, \nabla_{\mathbf{x}} \log p_i(\mathbf{x} \mid t).$$

We set $T = 100$ and use the discretization method and the variance scheduling presented in Appendix B.1.

Task Model. For the regression task on the Gaussian Mixture distribution, we consider a simple nonlinear model that maps the input vector $\mathbf{x} \in \mathbb{R}^2$ to a scalar prediction. The model operates on the first coordinate of the input, applying a quadratic transformation. The target label is defined as the second coordinate of the input. The loss function is the mean squared error between the model prediction and the label as follows.

$$\ell(\theta; \mathbf{x}) = \left(\theta_{[1]} \mathbf{x}_{[1]}^2 + \theta_{[2]} \mathbf{x}_{[1]} + \theta_{[3]} - \mathbf{x}_{[2]}\right)^2.$$

Pretrained Model Training. The regression model is first trained using ERM on a dataset containing 200 samples drawn from the predefined generative model. The dataset is evenly split into training and validation sets, each consisting of 100 samples. Training is conducted for 1000 epochs using the Adam optimizer with a learning rate of 0.1 and a batch size of 100.

Fine-Tuning. Each method, including the proposed RAMIS framework and all baselines, performs 1000 epochs of fine-tuning to adapt to newly generated samples. For RAMIS, fine-tuning is conducted using samples drawn via importance sampling from the generative model, where the importance weight function is chosen as the identity function, i.e., $\varphi(\ell) = \ell$. Baseline methods are fine-tuned using the same number of newly generated samples, but drawn uniformly from the base generative model. All fine-tuning procedures use the same optimizer and training configuration as the initial ERM phase.

Table 4: CVaR (mean $\pm$ std) across quantile levels β when training with the Extremile objective.

Method \ β	0.99	0.95	0.90	0.80	0.50
RAMIS + Extremile	**0.187 $\pm$ 0.144**	**0.064 $\pm$ 0.021**	**0.044 $\pm$ 0.015**	**0.0288 $\pm$ 0.0062**	**0.0158 $\pm$ 0.0015**
Extremile	0.339 $\pm$ 0.207	0.090 $\pm$ 0.044	0.048 $\pm$ 0.021	0.0289 $\pm$ 0.0041	0.0179 $\pm$ 0.0025

Applications with Other Risk Measures. This subsection examines whether the proposed framework based on the score-based generative models and loss-guided sampling is also helpful when training with different risk objectives. As an example, we consider *Extremiles* (Daouia et al., 2022).

Let $\mathbf{L} = \ell(\theta; \mathbf{X})$ denote the per-sample loss and let $\tilde{F}$ be the CDF of $\mathbf{L}$. Following (Daouia et al., 2022, 2019), define $\tilde{K}_\tau(z) = z^{r(\tau)}$ for $z \in [0,1]$ and $0.5 \leq \tau < 1$, where $r(\tau) = \log(1/2)/\log(\tau)$, and set $J_\tau(z) = \frac{d}{dz}\tilde{K}_\tau(z)$. The probability-weighted-moment form of the τ extremile is given as

$$\mathbb{E}_{\mathbf{X}^p \sim p}\big[\, \mathbf{L}\, J_\tau(\tilde{F}(\mathbf{L}))\,\big].$$

In our implementation, $\tilde{F}$ is estimated on each minibatch using a weighted empirical CDF based on the importance weighted samples from the pretrained generative model. We evaluate $\beta \in \{0.99, 0.95, 0.90, 0.80, 0.50\}$ and set $\tau = \beta$ to align tail emphasis with the CVaR quantile.

Table 4 indicates that RAMIS+Extremile achieves lower tail risk than optimizing the Extremile objective alone on the evaluated setups without importance samples. This empirical evidence supports the usefulness of importance-weighted samples when training with a different risk objective. Moreover, designing the importance sampling distribution to reflect the chosen risk measure or target objective may further improve performance.

C.1 Baseline Method Implementations

We evaluate the proposed RAMIS framework against strong baselines that capture risk-sensitive training paradigms. All methods are implemented under a unified training pipeline, sharing the same network architecture, initialization, optimizer, and sample budget, in order to ensure a fair and controlled comparison.

Empirical Risk Minimization (ERM). ERM serves as the canonical risk-neutral baseline. It is trained directly using samples from the original data distribution $p(\mathbf{x})$ without reweighting or sample selection. The model is optimized to minimize the expected loss $\mathcal{L}_{\text{ERM}} = \mathbb{E}_{\mathbf{X}^p \sim p}\,[\ell(\theta; \mathbf{X}^p)]$. This objective corresponds to uniform sampling from p followed by standard stochastic gradient descent.

Stochastic SubGradient Method (SSGM). We also evaluate the stochastic model-based optimization method proposed in Meng and Gower (2023), which ours extends. This is recovered by disabling importance sampling, i.e., by setting the weighting function φ to a constant. In this setting, samples are generated from a pretrained score-based generative model approximating $p(\mathbf{x})$, and the model parameters are updated using stochastic subgradients of the loss.

Distributionally Robust CVaR (DORO). To implement the CVaR-DORO method (Zhai et al., 2021), given a minibatch of $\hat{B}$ samples and associated loss vector $\mathbf{l} \in \mathbb{R}^{\hat{B}}$, we sort losses in descending order and compute the following loss function

$$\mathcal{L}_{\text{CVaR-DORO}} = \frac{1}{(1-\beta)(\hat{B} - n_2)} \sum_{i=n_2}^{n_1} \mathbf{l}_{(i)},$$

where $\mathbf{l}_{(i)}$ denotes the i-th largest loss, and the selection range $[n_2, n_1]$ is determined by quantile truncation parameters $n_1 = \lfloor (1-\beta)\hat{B} \rfloor$ and $n_2 = 0$. Note that the loss vector is obtained via computing $\ell(\theta; \mathbf{x})$ where $\mathbf{x}$ is a realization of the minibatch. Although DORO was designed for an outlier-aware setting with $n_2 \geq 0$, here we set $n_2 = 0$ and use it simply as a within-minibatch sorter that selects the top $(1-\beta)$ fraction of highest-loss samples.

χ^2-Divergence Robust Optimization (χ^2-DRO). We implement a divergence-constrained robust optimization baseline based on χ^2-divergence risk envelopes (Namkoong and Duchi, 2016). The robust objective is given by

$$\mathcal{L}_{\chi^2} = \inf_{\eta \in [0, L_{\max}]} \left\{ C \cdot \sqrt{\mathbb{E}_{\mathbf{X}^p \sim p} \left[\left(\ell(\mathbf{X}^p) - \eta \right)^+ \right]^2} + \eta \right\},$$

where $C = \sqrt{1 + \left(\frac{1}{1-\beta} - 1 \right)^2}$ is a divergence-induced robustness factor and $L_{\max} = 10$ is a fixed upper bound for η. The inner minimization over η is solved numerically using Brent's method.

D Experiments on Wireless Communications Channel State Information

Background. Accurate downlink channel state information (CSI) feedback is essential for high throughput and effective interference coordination in 5G and beyond. The challenge intensifies in massive MIMO deployments that span hundreds to thousands of subcarriers (Dahlman et al., 2013; Zaidi et al., 2018), producing high-dimensional CSI that must be returned to the base station (BS) from the user equipment (UE). Conventional feedback schemes scale poorly in this regime due to the substantial signaling overhead required.

To address this bottleneck, recent work has focused on deep learning-based CSI compression (Wen et al., 2018; Guo et al., 2022). These methods use autoencoders: the UE compresses CSI into a compact representation, and the BS decodes it. Trained on environment-specific data, such models learn channel priors and typically surpass codebook-based and compressed-sensing approaches. Early efforts like CsiNet (Wen et al., 2018) introduced convolutional architectures that outperformed classical baselines, followed by extensions that exploit temporal (Wang et al., 2018; Liu and Simeone, 2021) and spatial (Lu et al., 2020; Cai et al., 2019) structures in CSI. This data-driven direction has attracted considerable interest in both research and standardization (e.g., 3GPP Release 18 (Lin, 2022)).

More recently, score-based generative models have emerged as powerful tools for modeling and synthesizing wireless channels (Lee et al., 2025). Their ability to generate realistic channel realizations has enabled applications in joint source–channel coding (Ankireddy et al., 2025) and neural CSI compression (Kim et al., 2025b), where generative priors capture complex channel distributions and facilitate robust reconstruction from highly compressed representations.

Objective. Building on these developments, we investigate the effectiveness of loss-guided channel generation for improving neural CSI compression. Specifically, we fine-tune a pretrained neural CSI compressor by augmenting its training set with high-loss channel realizations generated via a score-based channel generative model.

Dataset. We evaluate our framework on a wireless CSI compression task using synthetic datasets generated via the Quadriga channel simulator (QUAsi Deterministic RadIo channel GenerAtor) (Jaeckel et al., 2014; Jaeckel et al., 2021). Specifically, we construct eight distinct datasets, denoted $\mathcal{D}(1), \ldots, \mathcal{D}(8)$. These datasets are designed to capture diverse propagation environments representative of real-world deployment scenarios.

Table 5 details the configuration of each dataset. The datasets differ in center frequency, BS cell type (macro vs. micro), propagation conditions (line-of-sight (LOS) vs. non-line-of-sight (NLOS)), and the number of dominant signal clusters N_{dc} used in channel generation, as defined in (Jaeckel et al., 2021, Sec. 3). Each dataset corresponds to a distinct wireless environment with varying scattering complexity and geometry. All BSs are placed at coordinate (0, 0), with macro-cell BSs positioned at a height of 25 meters and micro-cell BSs at 10 meters. The UE is equipped with omnidirectional antennas and randomly located within a circular region of radius 30 meters centered at the BS. The BS antennas follow the 3GPP-3D antenna model.

To reduce training complexity, we operate on a compact angular–delay representation of the channel by transforming the CSI instances to the delay–angle domain via a 2D IFFT and cropping out the high-delay region, which is nearly zero. The result is a 32×32 complex-valued tensor per sample.

Table 5: Channel model configuration

	Center frequency	Channel model	Propagation	N_{dc}
$\mathcal{D}(1)$	0.8GHz	Urban Micro-Cell	LOS	10
$\mathcal{D}(2)$	2.4GHz	Urban Micro-Cell	LOS	10
$\mathcal{D}(3)$	0.8GHz	Urban Macro-Cell	LOS	5
$\mathcal{D}(4)$	2.4GHz	Urban Macro-Cell	LOS	5
$\mathcal{D}(5)$	0.8GHz	Urban Micro-Cell	NLOS	50
$\mathcal{D}(6)$	2.4GHz	Urban Micro-Cell	NLOS	50
$\mathcal{D}(7)$	0.8GHz	Urban Macro-Cell	NLOS	40
$\mathcal{D}(8)$	2.4GHz	Urban Macro-Cell	NLOS	40

Generative Model. The denoising network used in our diffusion model is based on a modified UNet architecture. The network is implemented using the `UNet2DModel` class provided by `diffusers` library (von Platen et al., 2022), with an input resolution of 32×32 and two input/output channels corresponding to the real and imaginary parts of the CSI tensor. Each block in the UNet contains two residual convolutional layers (`layers_per_block=2`). The encoder path comprises six downsampling stages with output channel sizes of $(128, 128, 256, 256, 512, 512)$, where the fifth block includes a spatial self-attention mechanism via the `AttnDownBlock2D` layer. The decoder mirrors this structure, employing six upsampling blocks with corresponding channel sizes in reverse order, and includes an attention layer (`AttnUpBlock2D`) in the second stage of the decoder.

The discretization of the SDE based on this score model follows the method in Appendix B.1, including the variance scheduling and $T = 100$.

Task Model. We adopt a vector-quantized autoencoder architecture for compressing and reconstructing CSI matrices. Specifically, we employ `VQModel` in `diffusers` (von Platen et al., 2022) and customize it to operate on two-channel inputs (e.g., representing real and imaginary components) and outputs reconstructions of the same dimensionality.

The encoder network comprises three 2D downsampling blocks, each with increasing channel capacity $(64, 128, 256)$ and two convolutional layers per block, enabling the model to effectively compress spatially correlated CSI features. Symmetrically, the decoder is composed of three upsampling blocks mirroring the encoder's configuration. The architecture utilizes the SiLU activation function and group normalization with 32 groups. The latent representation has 128 channels and is discretized by using four vector quantization learnable embeddings of dimension 128. During forward propagation, the output includes both the reconstructed CSI sample and an auxiliary vector quantization commitment loss.

Generative Model Training Configuration. The score-based generative model is trained for 10^3 epochs using a dataset consisting of 8×10^4 samples. We use the Adam optimizer with an initial learning rate of 10^{-4}. We adopt the one-cycle learning rate scheduler, which increases the learning rate linearly to a peak value during the first 25% of training steps and then anneals it following a cosine decay schedule. The scheduler is configured with total steps set to the product of the number of epochs and the number of batches per epoch.

Pretrained Model Training Configuration. The pretrained model is trained using ERM with the mean squared error loss over 10^2 epochs with a training set of size 10^5. We use the Adam optimizer with a learning rate of 10^{-4} and a batch size of 100.

Fine-Tuning. The fine-tuning learning rate is set to 10^{-5} and optimization is conducted using Adam. For RAMIS, the importance weight function φ is chosen as the squared loss function (i.e., $\varphi(\ell) = \ell^2$). For baseline methods, fine-tuning is performed using 10^5 newly generated samples from the generative model, without the importance reweighting.

The remaining details, including the baseline methods implementation and generative model discretization, follow the setups described in Appendix C.

Additional Results. As $\beta \to 0$, the conditional value-at-risk objective $\mathrm{CVaR}_\beta(\theta)$ converges to the standard empirical risk minimization (ERM) objective. We set $\beta = 0$ and evaluate the average-case performance of the proposed method in comparison with the baselines (Table 6).

Table 6: ERM performance when optimizing $\beta = 0$ (CVaR reduces to ERM). Metric units match Table 2; lower is better.

	RAMIS (ours)	SSGM	DORO	χ^2-DRO	ERM
ERM ($\beta = 0$)	0.2032	0.2032	0.2029	0.2029	**0.2028**

Table 7: Cross-quantile evaluation of models trained with β=0.99. Entries are mean $\pm$ std over seeds; lower is better.

β	RAMIS (ours)	SSGM	DORO	χ^2-DRO	ERM
0.99 (train target)	**2.10** $\pm$ 0.0028	2.21 $\pm$ 0.0341	2.28 $\pm$ 0.0032	2.50 $\pm$ 0.2330	2.22 $\pm$ 0.0016
0.95	**1.41** $\pm$ 0.0283	1.43 $\pm$ 0.0153	1.49 $\pm$ 0.0034	1.72 $\pm$ 0.2293	1.44 $\pm$ 0.0019
0.90	**1.09** $\pm$ 0.0121	1.10 $\pm$ 0.0059	1.14 $\pm$ 0.0029	1.38 $\pm$ 0.2267	1.10 $\pm$ 0.0018
0.80	0.77 $\pm$ 0.0028	0.78 $\pm$ 0.0012	0.80 $\pm$ 0.0014	1.02 $\pm$ 0.2249	**0.76** $\pm$ 0.0013
0.50	0.39 $\pm$ 0.0069	0.41 $\pm$ 0.0038	0.43 $\pm$ 0.0004	0.63 $\pm$ 0.2251	**0.38** $\pm$ 0.0003
0.00	0.21 $\pm$ 0.0048	0.23 $\pm$ 0.0045	0.24 $\pm$ 0.0005	0.44 $\pm$ 0.2229	**0.20** $\pm$ 0.0001

The results show that all baseline methods and the proposed RAMIS framework exhibit nearly identical performance under $\beta = 0$. This further indicates that RAMIS with importance samples neither improves nor degrades the average-case performance, confirming its consistency.

When $\beta \neq 0$, we study whether emphasizing tail risk induces trade-offs with average-case performance. We fix the training target to $\beta = 0.99$ (i.e., the worst 1% tail) and evaluate each trained model across a sweep of non-target quantiles $\beta \in \{0.99, 0.95, 0.90, 0.80, 0.50, 0.00\}$.

RAMIS delivers the strongest performance at the intended target $\beta = 0.99$ and across the high-risk tail, outperforming alternatives at $\beta \in \{0.95, 0.90\}$. Conversely, at quantiles under $\beta \leq 0.80$, including $\beta = 0.50$ and $\beta = 0$, ERM achieves the best results. Overall, the results show a tail–average trade-off: optimizing for large β improves robustness in rare, high-loss regimes, but can modestly reduce average-case performance.

E Further Experimental Results

E.1 Cost of Importance Sampling

Compared to the base generative model, the computational cost of importance sampling primarily arises from the need to evaluate a composed gradient of the importance weight function $w(\mathbf{x})$, which in our case is $\varphi(\ell(\theta_0, \mathbf{x}))$, alongside the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$. Specifically, the guided score used in our method is approximated as $g(\mathbf{x}, t) \approx \nabla_{\mathbf{x}} \log w(\bar{\mathbf{x}}'_0|_{\mathbf{x},t})$, as explored in (Chung et al., 2022). This gradient computation must be performed at each reverse diffusion step, making it more costly than standard sampling from the base model. This cost can be mitigated by adopting a lightweight approximation strategy based on the directional finite-difference method proposed in (Kim et al., 2025a). As a result, the sampling time increases by a modest factor, approximately $2.3\times$, relative to the base generative model. This includes the overhead of computing two evaluations of the score function per step, as well as a single backward pass to differentiate the importance weighting function.

This added cost may be negligible for scenarios where the goal is to capture rare, high-loss samples. In such regimes, conventional generative models often fail to sample from the critical low-density regions, which dominate tail-risk measures.

To quantify the cost-benefit trade-off, we report empirical results in Figure 4. Each plot shows the CVaR performance (vertical axis) as a function of the number of samples used (horizontal axis) for different values of $\beta \in \{0.99, 0.95, 0.90, 0.80, 0.50\}$.

As shown in Figure 4, when $\beta = 0.99$, the proposed method achieves superior CVaR performance even with only $1/8$ the number of samples compared to baseline methods. Similar trends are observed for $\beta = 0.95$, underscoring the importance of sampling from rare, high-risk regions. As β decreases (e.g., $\beta = 0.80$ or $\beta = 0.50$), the importance of extreme loss values diminishes, and a larger sample budget becomes necessary to achieve parity or superiority over robust baselines such as χ^2-DRO.

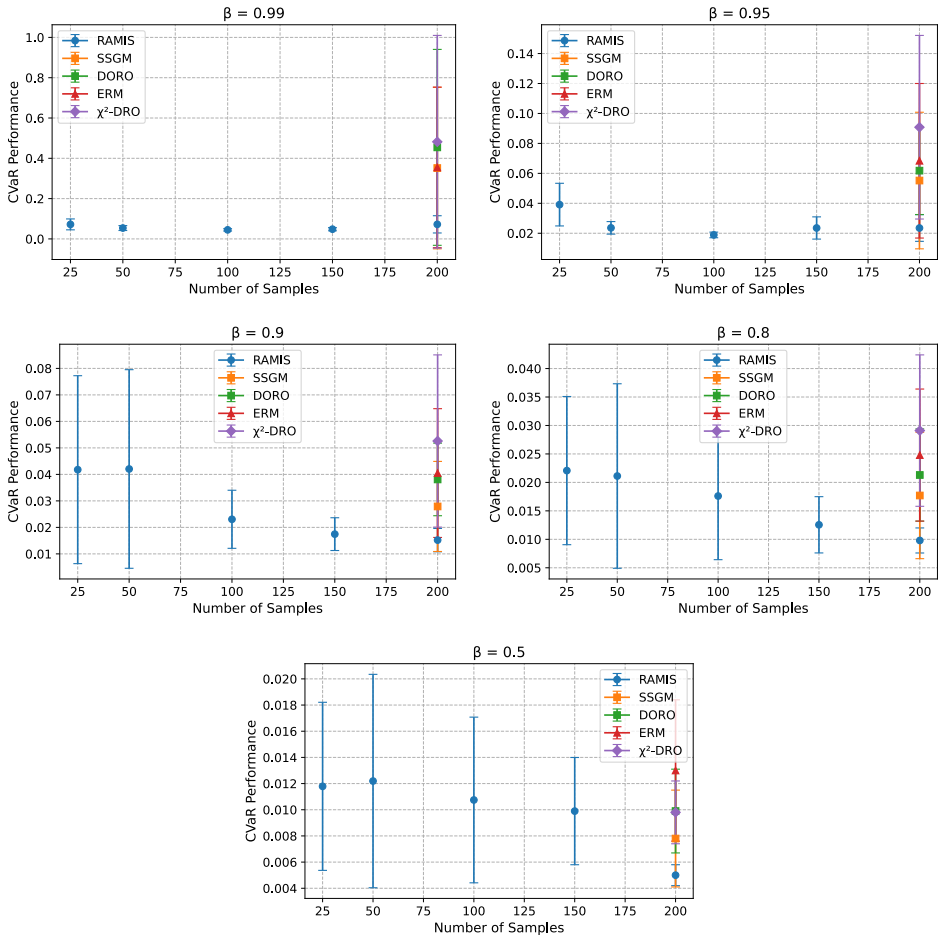

Figure 4: CVaR performance versus number of samples for varying β levels. The proposed method significantly outperforms baselines under high β (e.g., 0.99, 0.95), even with substantially fewer samples (e.g., $1/8$ of the budget).

E.2 Impact of Importance Level Emphasis

We investigate how the choice of the importance weighting function φ affects performance.

Square-root emphasis. Consider the optimization-noise term

$$\hat{v}(q) = \mathbb{E}_{\mathbf{X}^p \sim p}\left[\frac{w^*(\mathbf{X}^p)^2}{(1-\beta)^2}\frac{p(\mathbf{X}^p)}{q(\mathbf{X}^p)} + 1\right].$$

Then the desired importance weight is $q(\mathbf{x}) \propto w^*(\mathbf{x})p(\mathbf{x})$; substituting this choice yields a Jensen-type inequality as

$$\hat{v}(p) = \frac{1}{(1-\beta)^2}\mathbb{E}_{\mathbf{X}^p \sim p}\left[w^*(\mathbf{X}^p)^2\right] + 1 \geq \frac{1}{(1-\beta)^2}\mathbb{E}_{\mathbf{X}^p \sim p}\left[w^*(\mathbf{X}^p)\right]\mathbb{E}_{\mathbf{X}^p \sim p}\left[w^*(\mathbf{X}^p)\right] + 1. \tag{42}$$

Based on this, we consider

$$\varphi(\ell) = \sqrt{\ell} + c,$$

where $c \geq 0$ is a scalar hyperparameter that adjusts the relative emphasis on high-loss regions. Intuitively, smaller values of c amplify the contrast between low- and high-loss samples, focusing the generative model more aggressively on rare, high-risk regions. In contrast, larger values of c

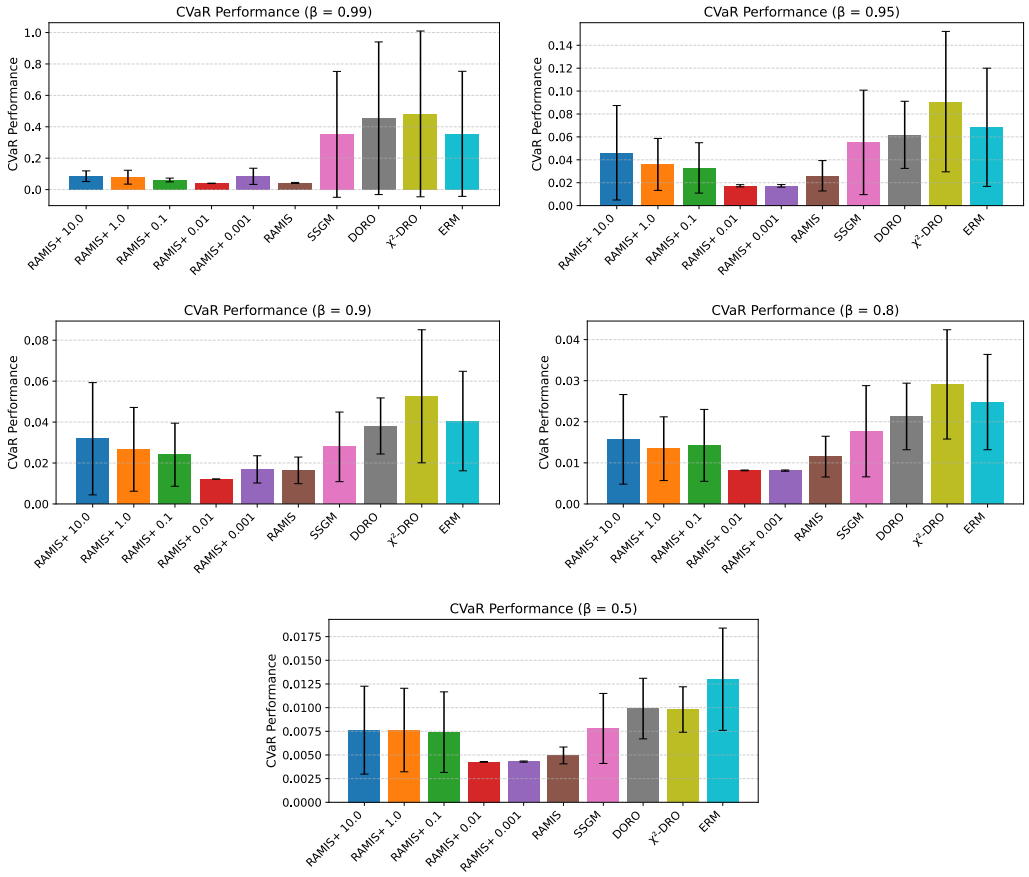

Figure 5: CVaR performance across different values of $\varphi(\ell) = \sqrt{\ell} + c$ with varying β levels.

can flatten the weight distribution, thereby reducing the selectivity of the sampling process. To empirically examine this effect, we evaluate $\varphi(\ell) = \ell + c$ for $c \in \{0.0, 0.001, 0.01, 0.1, 1.0, 10.0\}$, denoted "RAMIS+$c$" in Figure 5.

Figure 5 summarizes the CVaR performance under various risk levels β. For $\beta \in \{0.99, 0.95\}$, all φ variants significantly outperform both risk minimization and robust optimization baselines. Notably, configurations with $c \in \{0.001, 0.01\}$ achieve the best results, improving CVaR by over 30% relative to the strongest baseline.

Linear emphasis. We also consider the following linear design to mimic the behavior of w^* as

$$\varphi(\ell) = \ell + c.$$

Similarly, as shown in Figure 6, the best performance occurs for relatively small $c \in \{0.0, 0.01, 0.001\}$, while larger constants consistently degrade results. Across all settings, the proposed method outperforms baselines that do not use importance samples. In contrast, setting $c = 1.0$ or higher diminishes the emphasis on high-loss samples and leads to degradation in performance, approaching that of SSGM without importance sampling.

These results indicate that the additive constant c in φ can serve as a simple hyperparameter controlling the strength of importance sampling. Smaller c sharpens focus on high-risk inputs (beneficial for tail-sensitive objectives), whereas overly large c flattens the weights.

Overall, in these experiments, the linear choice $\varphi(\ell) = \ell + c$ slightly outperforms the square-root family, and we adopt $\varphi(\ell) = \ell$ in the main experiments.

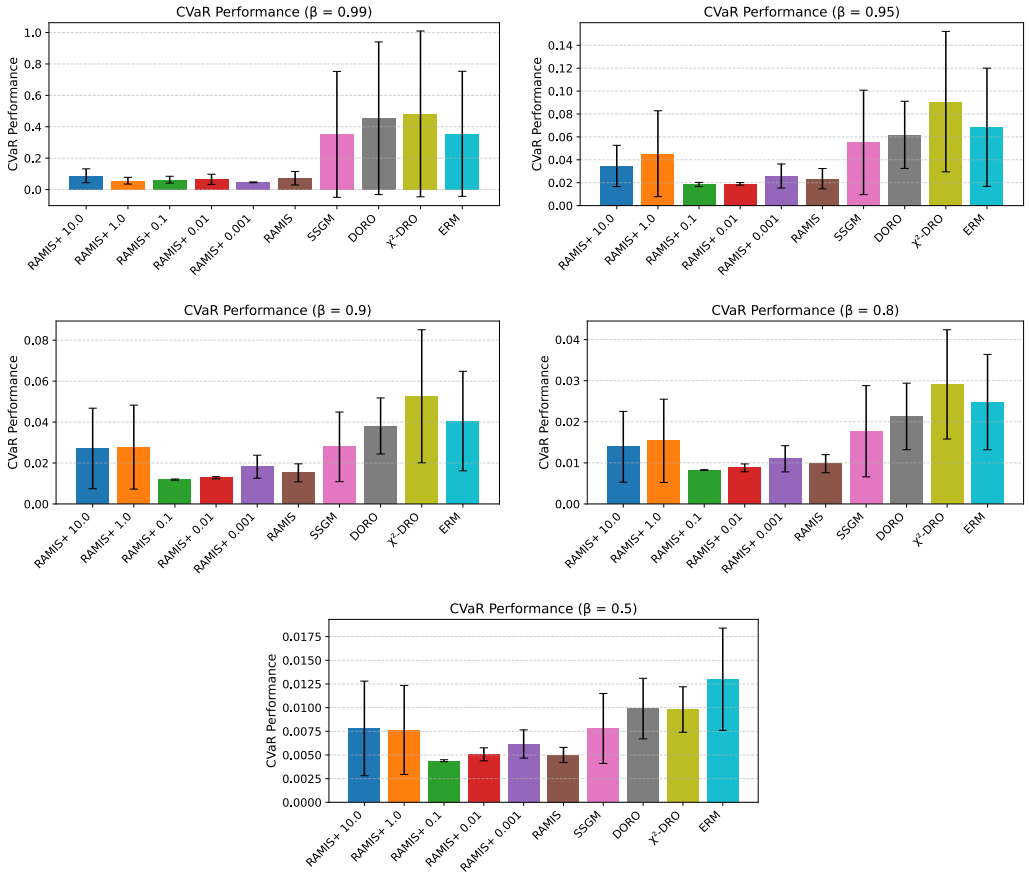

Figure 6: CVaR performance across different values of $\varphi(\ell) = \ell + c$ with varying β levels. Our method consistently outperforms baseline methods, especially under risk-sensitive settings ($\beta \geq 0.9$).

Limitations. A limitation of these experiments is that constructing the importance distribution via guided sampling can introduce sampling-approximation error in the score-based generative model. In addition, the analysis controls stochastic noise through an upper-bound surrogate. Exploring guidance procedures with reduced approximation error and alternative weight functions φ informed by tighter bounds may improve the quality of the importance sampling and its empirical performance.