Appendix A: Examples of Heavy-Tailed Distributions

We start this section by introducing and characterizing the log-normal distribution, which, as will be explained in the following, has the "lightest" tail among all heavy-tailed distributions of our definition.

EXAMPLE EC.1. *Log-normal distribution.* The log-normal distribution $X \sim \text{Lognormal}(\mu, \sigma^2)$ is defined by $ln(X) \sim \mathcal{N}(\mu, \sigma^2)$, and has probability density function

$$p(x;\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right). \tag{EC.1}$$

The distribution has properties $\mathbb{E}X = \exp(\mu + \sigma^2/2)$ and $\text{Var}(X) = [\exp(\sigma^2) - 1]\exp(2\mu + \sigma^2)$.

We now verify that the log-normal distribution satisfies Definition 1 of heavy-tailed distribution. From definition of the log-normal distribution we have cdf

$$F(x;\mu,\sigma) = \Phi\left(\frac{\ln x - \mu}{\sigma}\right),$$

where $\Phi(x)$ is the cdf of standard normal distribution. Therefore, the tail probability is given as

$$\mathbb{P}(X > x) = 1 - F(x;\mu,\sigma) = Q\left(\frac{\ln x - \mu}{\sigma}\right),$$

where the Q-function is the tail probability of the standard normal distribution, and can be upper and lower bounded as (see Borjesson and Sundberg (1979), for example)

$$\left(\frac{x}{1+x^2}\right)\phi(x) < Q(x) < \frac{\phi(x)}{x}, \quad x > 0,$$

where $\phi(x)$ is the pdf of standard normal distribution. Therefore, for the log-normal distribution $\text{Lognormal}(\mu, \sigma^2)$, we have

$$\mathbb{P}(X > x) > \frac{\sigma(\ln x - \mu)}{\sigma^2 + (\ln x - \mu)^2} \cdot \phi\left(\frac{\ln x - \mu}{\sigma}\right) = \frac{\sigma(\ln x - \mu)}{\sqrt{2\pi}\,(\sigma^2 + (\ln x - \mu)^2)} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right). \tag{EC.2}$$

To simplify the analysis, consider the case $\mu = 0,\ \sigma = 1$, we have (EC.2) equal to

$$\begin{aligned}
&\frac{\ln x}{\sqrt{2\pi}\,(1 + (\ln x)^2)} \exp\left(-\frac{(\ln x)^2}{2}\right) = \frac{\ln x}{\sqrt{2\pi}\,(1 + (\ln x)^2)} \left(\exp\left(-\frac{\ln x}{2}\right)\right)^{\ln x} \\
&= \frac{\ln x}{\sqrt{2\pi}\,(1 + (\ln x)^2)} x^{-(\ln x)/2} > \frac{1}{2\sqrt{2\pi}\ln x} x^{-(\ln x)/2}
\end{aligned}$$

for big x. Therefore, we have

$$\ln \mathbb{P}(X > x) > \ln\left(x^{-(\ln x)/2}\right) - \ln(2\sqrt{2\pi}\ln x) = -\frac{1}{2}(\ln x)^2 - \ln(2\sqrt{2\pi}\ln x). \tag{EC.3}$$

To verify the definition by showing

$$\limsup_{x\to\infty} \frac{\ln \mathbb{P}(\|X\| > x)}{x^\beta} \le -C$$

does not hold for any $\beta > 0$ and $C > 0$, observe that

$$(\ln x)^2 = o(x^\beta); \quad \ln(\ln x) = O(x^\beta),$$

for arbitrary $\beta > 0$, and take this into (EC.3). As a result,

$$\limsup_{x\to\infty} \frac{\ln \mathbb{P}(\|X\| > x)}{x^\beta} = 0,$$

contradicting the definition of light-tailed, which means the log-normal distribution is heavy-tailed.

To give a better characterization of the tail orders, we note that, roughly speaking, log-normal distribution lies at the boundary between the lighted-tailed and heavy-tailed distributions, and is of limiting order of the polynomial-order distributions, as the polynomial order α (for $\mathbb{P}(X > x) = O(x^{-\alpha})$) goes to ∞. Provided in the following examples are the most common families of distribution with polynomial rate of decay. The formulation and properties of these distributions are used many times in our work.

EXAMPLE EC.2. *Location-scale t distribution.* The location-scale t distribution $\mathcal{T}(\nu, \mu, \tau^2)$ has probability density function

$$\begin{aligned} &p(x; \nu, \mu, \tau^2) \\ &= \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})\sqrt{\pi\nu\tau^2}} \left(1 + \frac{1}{\nu}\frac{(x-\mu)^2}{\tau^2}\right)^{-\frac{\nu+1}{2}}, \end{aligned} \tag{EC.4}$$

and is of polynomial order $P(X > x) = O(x^{-\nu})$, $|x| \to \infty$. The distribution has properties $\mathbb{E}X = \mu$ for $\nu > 1$ and $\text{Var}(X) = \tau^2\nu/(\nu-2)$ for $\nu > 2$.

EXAMPLE EC.3. *Pareto distribution.* The Pareto distribution $\mathcal{P}(\alpha, x_m)$ has probability density function

$$p(x; \alpha, x_m) = \frac{\alpha x_m^\alpha}{x^{\alpha+1}}, \quad x \ge x_m, \tag{EC.5}$$

and $p(x; \alpha, x_m) = 0$ for $x < x_m$. Note that the Pareto distribution does not have full support. The distribution is of polynomial order $P(X > x) = O(x^{-\alpha})$, $x \to \infty$, and has properties $\mathbb{E}X = \alpha x_m/(\alpha-1)$ for $\alpha > 1$ and $\text{Var}(X) = x_m^2\alpha/[(\alpha-1)^2(\alpha-2)]$ for $\alpha > 2$.

EXAMPLE EC.4. *Zero Symmetric Pareto distribution.* The Zero Symmetric Pareto distribution $\mathcal{ZSP}(\alpha)$ is a generalization of Pareto distribution that has full support on $\mathbb{R}$ and is symmetric by zero. The probability density function of $\mathcal{ZSP}(\alpha)$ is

$$p(x; \alpha, x_m) = \frac{\alpha}{2(1+|x|^{\alpha+1})}, \quad x \in \mathbb{R}, \tag{EC.6}$$

The distribution is of polynomial order $P(X > x) = O(x^{-\alpha})$, $x \to \infty$.

Appendix B: Proofs

B.1. Proof of Lemma 1

Proof: The proof is straightforward. Note that $\mathcal{S}_p = \{\mathbf{x}^{(j)} \in \mathcal{S} | \|\mathbf{x}^{(j)}\| \geq \mathrm{VaR}_{data}(p)\}$. By definition of $\mathcal{S}_p$ and the definition of VaR, we have

$$\mathbb{P}_{\text{data}}(\mathcal{S}_p = \emptyset) = \mathbb{P}_{\text{data}}\{\|\mathbf{x}^{(j)}\| < \mathrm{VaR}_{\text{data}}(p),\ \forall j = 1, 2, \ldots, M\} = (1-p)^M.$$

When $M \geq O(p^{-2})$, $(1-p)^M \leq (1-p)^{(1/p)\cdot(1/p)}$, therefore

$$\lim_{p\to 0} \mathbb{P}_{\text{data}}(\mathcal{S}_p \neq \emptyset) = 1 - \lim_{p\to 0} \mathbb{P}_{\text{data}}(\mathcal{S}_p = \emptyset) \geq 1,$$

which completes the proof.

B.2. Proof of Theorem 1

Proof: For two arbitrary points $\mathbf{x}_0^{(j)}$ and $\mathbf{x}_0^{(k)}$ from the dataset $\mathcal{S} = \{\mathbf{x}_0^{(n)}\}_{n=1}^N$. We first have the same upper bound as given in the proof of Proposition 1 of Song and Ermon (2020):

$$\mathbb{E}_{q(\mathbf{x}|\mathbf{x}_0^{(j)})}[r(\mathbf{x}|\mathbf{x}_0^{(k)})] \leq \frac{1}{2}\exp\left(-\frac{\|\mathbf{x}_0^{(k)} - \mathbf{x}_0^{(j)}\|^2}{8\sigma_T^2}\right).$$

Let $\mathbf{x}_0^{(j)}$ be replaced by $\bar{\mathbf{x}}$, and $\mathbf{x}_0^{(k)}$ be replaced by an arbitrary $\tilde{\mathbf{x}} \in \mathcal{S}'_p$, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}})\right] \leq \frac{1}{2}\exp\left(-\frac{\|\tilde{\mathbf{x}} - \bar{\mathbf{x}}\|^2}{8\sigma_T^2}\right). \tag{EC.7}$$

Next, starting from the definition of heavy-tailed data, we first show that Definition 1 of heavy-tailed is equivalent to

$$\limsup_{x\to\infty} \frac{\ln \mathbb{P}(\|X\| > x)}{x^\beta} = 0, \quad \forall \beta > 0. \tag{EC.8}$$

This can be checked by noting that $\mathbb{P}(\|X\| > x) \in [0, 1]$, therefore $[\ln \mathbb{P}(\|X\| > x)]/x^\beta < 0,\ \forall x > 0$, and thus have supreme limit at most 0. If $\exists\ \beta_0 > 0$ such that it is not zero, then the definition of light-tailed distribution holds. Therefore, (EC.8) gives the equivalent definition of heavy-tailed distributions.[1]

Using definition (EC.8), for a heavy-tailed random variable $X \sim q_{\text{data}}(\mathbf{x}_0)$, there is a sequence of $\{x_n\}$ such that $x_n \to \infty$, and

$$\lim_{n\to\infty} \frac{\ln \mathbb{P}(\|X\| > x_n)}{x_n^\beta} = 0.$$

Therefore, $\forall \varepsilon > 0$, $\exists\ N$, $\forall n > N$,

$$\ln \mathbb{P}(\|X\| > x_n) > -\varepsilon x_n^\beta. \tag{EC.9}$$

[1] We note that (EC.8) can be compared to the formulation of heavy-tailed distributions given in Lemma 1.2 of Nair et al. (2013),

$$\liminf_{x\to\infty} -\frac{\ln \mathbb{P}(\|X\| > x)}{x} = 0.$$

Therefore, the difference in the two versions of definitions of heavy-tailed distributions lies in only the choice of β.

We now look at the definition of VaR, given as

$$\mathrm{VaR}_X(p) = \inf_{x\in\mathbb{R}}\{\mathbb{P}(|X|\geq x)\leq p\} = \inf_{x\in\mathbb{R}}\{\ln\mathbb{P}(|X|\geq x)\leq \ln p\}.$$

Note that from (EC.9) we have $\mathbb{P}(\|X\| > x_n) > \exp\{-\varepsilon x_n^\beta\},\ \forall n$. Together with the definition of VaR, this gives

$$x_n < \mathrm{VaR}_{\text{data}}(\exp(-\varepsilon x_n^\beta)),\ \forall n. \tag{EC.10}$$

To explain, (EC.10) arises from that the risk level of x_n is bigger than $p_n = \exp(-\varepsilon x_n^\beta)$, which means $\mathrm{VaR}_{\text{data}}(\exp(-\varepsilon x_n^\beta))$ shoud be more to the extreme, i.e., on the right side of x_n. Also, the relationship between x_n and p_n is established as $x_n = (-\ln p_n/\varepsilon)^{1/\beta}$, and $p_n \to 0$ along with $x_n \to \infty$.

Now consider $\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x};\mathbf{x}^{p_n})]$. Note that when p_n is sufficiently small, implying $\forall \tilde{\mathbf{x}}^{p_n}\in\mathcal{S}'_{p_n}$, $\|\tilde{\mathbf{x}}^{p_n}\|$ sufficiently large, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}}^{p_n})] \leq \frac{1}{2}\exp\left(-\frac{\|\tilde{\mathbf{x}}^{p_n}-\bar{\mathbf{x}}\|^2}{8\sigma_T^2}\right) \leq \frac{1}{2}\exp\left(-\frac{\|\tilde{\mathbf{x}}^{p_n}\|^2}{2\sigma_T^2}\right).$$

By $\tilde{\mathbf{x}}^{p_n}\in\mathcal{S}'_{p_n}$ and the definition of $\mathcal{S}'_{p_n}$, we have $\|\tilde{\mathbf{x}}^{p_n}\| \geq \mathrm{VaR}_{\text{data}}(p_n)$. Together with (EC.10), we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}}^{p_n})] \leq \exp\left(-\frac{\|\tilde{\mathbf{x}}^{p_n}\|^2}{2\sigma_T^2}\right) \leq \exp\left(-\frac{x_n^2}{2\sigma_T^2}\right)$$

Plug in $x_n = (-\ln p_n/\varepsilon)^{1/\beta}$, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}}^{p_n})] \leq \exp\left(-\frac{(-\ln p_n)^{2/\beta}}{2\sigma_T^2\varepsilon^{1/\beta}}\right) \leq \exp\left\{-\left(\ln\frac{1}{p_n}\right)^{2/\beta}\right\},$$

where the second inequality comes from that ε is arbitrary. Also, taking the supremum over $\tilde{\mathbf{x}}\in\mathcal{S}'_{p_n}$ gives

$$\sup_{\tilde{\mathbf{x}}\in\mathcal{S}'_{p_n}} \mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] \leq \exp\left\{-\left(\ln\frac{1}{p_n}\right)^{2/\beta}\right\},$$

Therefore,

$$\varliminf_{p\to 0}\sup_{\tilde{x}\in\mathcal{S}_p} \mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})]\exp\left\{(\ln(1/p))^{2/\beta}\right\} \leq 1, \quad \forall\beta>0.$$

B.3. Proof of Proposition 1

Proof: We first prove the case where π_{data} is sub-Gaussian. Note that $\forall\tilde{\mathbf{x}}\in\mathcal{S}'_p$

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] = \int \frac{q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})}{\sum_{k=1}^M q(\mathbf{x}|\mathbf{x}^{(k)})} \geq \frac{1}{M}\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}}).$$

The inequality comes from that $q(\mathbf{x}|\bar{\mathbf{x}})$ is the pdf of Gaussian distribution, and we assume $\sigma_T > 1$ w.l.o.g. Now consider $\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})$. Similar to the proof of Proposition 1 of Song and Ermon (2020), we have

$$\begin{aligned}
\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}}) &= \frac{1}{(2\pi\sigma_T^2)^d}\int \exp\left(-\frac{1}{2\sigma_T^2}\left[\|\mathbf{x}-\tilde{\mathbf{x}}\|^2+\|\mathbf{x}-\bar{\mathbf{x}}\|^2\right]\right)d\mathbf{x} \\
&= \frac{1}{(2)^{d/2}(2\pi\sigma_T^2)^{d/2}}\frac{1}{(2\pi(\sigma_T/\sqrt{2})^2)^{d/2}}\int \exp\left(-\frac{1}{4(\sigma_T/\sqrt{2})^2}\left[\|\mathbf{x}-\tilde{\mathbf{x}}\|^2+\|\mathbf{x}-\bar{\mathbf{x}}\|^2\right]\right)d\mathbf{x} \\
&= \frac{1}{(4\pi\sigma_T^2)^{d/2}}\exp\left(-\frac{\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|^2}{4\sigma_T^2}\right),
\end{aligned} \tag{EC.11}$$

where the last equality uses the same technique in the proof of Proposition 1 of Song and Ermon (2020), and we simply plug $\sigma_T/\sqrt{2}$ into their equations and omit the details of deriving the result.

The definition of sub-Gaussian gives that $\exists\, C > 0$,

$$\limsup_{x\to\infty} \frac{\ln \mathbb{P}(\|X\| > x)}{x^2} \le -C.$$

Therefore, $\exists A$ such that $\forall x > A$,

$$\ln \mathbb{P}(\|X\| > x) \le -\frac{C}{2}x^2.$$

Also, we have

$$\text{VaR}_X(p) = \inf_{x\in\mathbb{R}} \{\ln \mathbb{P}(|X| \ge x) \le \ln p\}$$

Therefore, $\forall x > A$, let $p(x) = e^{-Cx^2/2}$, by definition of VaR, which takes the infimum, we have $x \ge \text{VaR}_X(p(x))$.

Now consider $\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}}^{p(x)})\right]$, where $\tilde{\mathbf{x}}^{p(x)} \in \partial\mathcal{S}'_{p(x)}$. From (EC.11), for small enough $p(x)$, we have $\|\mathbf{x}^{p(x)} - \bar{x}\| \le 2\|\mathbf{x}^{p(x)}\|$, which gives

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}}^{p(x)})\right] \ge \frac{1}{M(4\pi\sigma_T^2)^{d/2}} \exp\left(-\frac{\|\mathbf{x}^{p(x)} - \bar{\mathbf{x}}\|^2}{4\sigma_T^2}\right) \ge \frac{1}{M(4\pi\sigma_T^2)^{d/2}} \exp\left(-\frac{\|\mathbf{x}^{p(x)}\|^2}{2\sigma_T^2}\right).$$

By $\tilde{\mathbf{x}}^{p(x)} \in \partial\mathcal{S}'_{p(x)}$ and the definition of $\mathcal{S}'_{p(x)}$, we have $\|\tilde{\mathbf{x}}^{p(x)}\| \le \text{VaR}_{\text{data}}(p(x)) \le x$. Therefore, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}}^{p(x)})\right] \ge \frac{1}{M(4\pi\sigma_T^2)^{d/2}} \exp\left(-\frac{\|\mathbf{x}^{p(x)}\|^2}{2\sigma_T^2}\right) \ge \frac{1}{M(4\pi\sigma_T^2)^{d/2}} \exp\left(-\frac{x^2}{2\sigma_T^2}\right)$$

Plug in $x = \sqrt{-2\ln p/C}$, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}}^{p(x)})\right] \ge \frac{1}{M(4\pi\sigma_T^2)^{d/2}} \exp\left(\frac{\ln p}{C\sigma_T^2}\right).$$

Take $\sigma_T = 1/\sqrt{C}$, we then have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}}^{p(x)})\right] \ge \frac{1}{M(4\pi\sigma_T^2)^{d/2}} \cdot p,$$

giving the result of the first part.

Now consider the case that π_{data} is heavy-tailed. We first note that, with the condition that π_{data} has a probability density function, we can apply the L'Hopital's rule to (EC.8). Note that

$$\frac{\frac{d}{dx}\ln \mathbb{P}(\|X\| > x)}{\frac{d}{dx}x^\beta} = \frac{f(x)}{1-F(x)}\frac{1}{\beta}x^{1-\beta}.$$

Since β can be any positive real number, we can take $\beta = 1$. The R.H.S. becomes $f(x)/[1-F(x)]$. By condition of the proposition, $\lim_{x\to\infty} f(x)/[1-F(x)]$ exists, which implies $\lim_{x\to\infty} \ln \mathbb{P}(\|X\| > x)/x$ exists and is equal to the supremum limit, 0. In this case, the sequence $\{x_n\}$ in the proof of Theorem 1 can be

any sequence such that $\lim_{n\to\infty} x_n = \infty$, therefore $p_n = \exp(-\varepsilon x_n)$ can be any sequence such that $p_n \to 0$. Modified from the proof of Theorem 1, we have

$$\lim_{p\to 0} \sup_{\tilde{x}\in\partial\mathcal{S}'_p} \mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}})\right] \exp\left\{\left(\ln(1/p)\right)^2\right\} \leq 1.$$

Note that

$$\exp\left\{\left(\ln\frac{1}{p}\right)^2\right\} = \left(\frac{1}{p}\right)^{\ln(1/p)},$$

therefore,

$$\sup_{\tilde{x}\in\partial\mathcal{S}'_p} \mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}})\right] \leq p^{\ln(1/p)}$$

for small enough p.

Specifically, $\sup_{\tilde{x}\in\partial\mathcal{S}'_p} \mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}\left[r(\mathbf{x}|\tilde{\mathbf{x}})\right]$ decays faster than any polynomial of p.

B.4. Proof of Proposition 2

Proof: First note that x has a differentiable probability density function for Langevin process to be defined. For a sub-Gaussian random variable with probability density function $q(x)$, we have $q(x) \propto e^{-\gamma|x|^\beta}$, where $\beta \geq 2$. According to Section 2.3.1 of Roberts and Tweedie (1996), the diffusion is exponentially ergodic, that is, it achieves exponential convergence.

For the log-normal distribution, for simplicity take $\mu = 0$ and $\sigma = 1$,

$$p(x) = \frac{1}{x\sqrt{2\pi}} \exp\left(-\frac{(\ln x)^2}{2}\right),$$

and

$$\log p(x) = -\frac{(\ln x)^2}{2} - \ln x - \frac{1}{2}\ln(2\pi).$$

Therefore,

$$|\nabla \log p(x)| = \left|\frac{1}{x}(1+\ln x)\right| \to 0, \quad x \to \infty.$$

According to Theorem 2.4 of Roberts and Tweedie (1996), the Langevin diffusion is not exponentially ergodic when $|\nabla \log p(x)| \to 0$. Therefore, when $q(x)$ is log-normal, Langevin diffusion does not achieve exponential convergence.

For distributions of polynomial order, i.e., $q(x) \propto x^{-\alpha} (\alpha > 0)$, we have $|\nabla \log q(x)| = |\alpha/x| \to 0$ as $x \to \infty$. Using Theorem 2.4 of Roberts and Tweedie (1996) again gives that the Langevin diffusion does not have exponential convergence.

B.5. Proof of Proposition 3

Proof: Note that we have

$$\nabla_x \log p(x) = -\frac{(\nu+1)x}{\nu+x^2}. \tag{EC.12}$$

For statement 1, we update with

$$X_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(X_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(X_t,t)\right) + \sigma_t\varepsilon,$$

with $\varepsilon_\theta(x,t) = -\sigma_t \nabla \log p(x;\nu,0,1)$, this is equal to

$$X_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(X_t - \frac{\sigma_t\beta_t}{\sqrt{1-\bar{\alpha}_t}}\frac{(\nu+1)X_t}{\nu+X_t^2}\right) + \sigma_t\varepsilon, \tag{EC.13}$$

For statement 2, we update with

$$X_{t-1} = X_t + s_t \nabla_X \log q(X_{t-1}) + \sqrt{2s_t}\varepsilon,$$

which is equal to

$$X_{t-1} = X_t - \frac{s_t(\nu+1)X_t}{\nu+X_t^2} + \sqrt{2s_t}\varepsilon. \tag{EC.14}$$

We note that equations (EC.13) and (EC.14) have similar forms, and are different only in terms of constants. Therefore, for simplicity we consider (EC.14) for demonstration. Note that $\sqrt{2s_t}\varepsilon$ is a Gaussian random variable and provides only the Gaussian tail. Therefore, we only consider the order of the first term, i.e.,

$$Y = X_t - \frac{s_t(\nu+1)X_t}{\nu+X_t^2}.$$

Note that the relationship between Y and X_t are deterministic. Without loss of generality we consider only the one-sided tail probability, $\lim_{y\to\infty}\mathbb{P}(Y>y)$. Note that, for Y to be sufficiently large, we also need X_t to be sufficiently large. Therefore, we have

$$\mathbb{P}\left(Y>y\right) = \mathbb{P}\left(X_t - \frac{s_t(\nu+1)X_t}{\nu+X_t^2} > y\right) \le \mathbb{P}\left(X_t - \frac{s(\nu+1)}{X_t} > y+\delta(y)\right), \tag{EC.15}$$

where $\lim_{y\to\infty}\delta(y)=0$. The equation is established by the equivalence of events when y is sufficiently large. To explain, the event $Y>y$ gives $X_t > f(y)$, where $\lim_{y\to\infty} f(y) = \infty$. This further gives $|(s_t(\nu+1)X_t)/(\nu+X_t^2) - (s_t(\nu+1))/X_t| = O(f(y)^{-3})$. Therefore, $Y>y$ implies $X_t - (s_t(\nu+1))/X_t > y+\delta(y)$ with $\delta(y)\to 0$. The last term of (EC.15) solves

$$X_t > \frac{y+\delta(y)+\sqrt{(y+\delta(y))^2+4s_t(\nu+1)}}{2}. \tag{EC.16}$$

For sub-Gaussian X_t, we have

$$\limsup_{x\to\infty}\frac{\ln\mathbb{P}(X_t>x)}{x^2} \le -C,$$

therefore, replacing x of the above equation with the *R.H.S.* of (EC.16), we have

$$\limsup_{y\to\infty} \frac{4\ln \mathbb{P}(Y>y)}{(y+\delta(y)+\sqrt{(y+\delta(y))^2+4s_t(\nu+1)})^2} \leq -C.$$

Note that

$$\limsup_{y\to\infty} \frac{4\ln \mathbb{P}(Y>y)}{(y+\delta(y)+\sqrt{(y+\delta(y))^2+4s_t(\nu+1)})^2} = \limsup_{y\to\infty} \frac{\ln \mathbb{P}(Y>y)}{y^2}.$$

Therefore, we have

$$\limsup_{y\to\infty} \frac{\ln \mathbb{P}(Y>y)}{y^2} \leq -C,$$

giving that Y is a sub-Gaussian random variable by definition. Therefore, $X_{t-1} = Y + \sqrt{2s_t\varepsilon}$ is also sub-Gaussian. By induction, we have X_{T-K} for finite K is sub-Gaussian.

B.6. Proof of Lemma 2

Proof: According to the definitions, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] = \int \frac{q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})}{\sum_{n=1}^N q(\mathbf{x}|\mathbf{x}_0^{(n)})} d\mathbf{x} \leq \sqrt{\int |q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})|^2 \, d\mathbf{x}} \sqrt{\int \left(\frac{1}{\sum_{n=1}^N q(\mathbf{x}|\mathbf{x}_0^{(n)})}\right)^2 d\mathbf{x}},$$

where the inequality follows from the Cauchy-Schwarz Inequality for integrals.

Proof of the upper bound:

Note that the second term of *R.H.S.* can be summarized into a constant C_1 independent of the choice of j and k chosen from the dataset. Now consider the first term, note that

$$
\begin{aligned}
&\int \left|q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})\right|^2 d\mathbf{x} \\
=&\prod_{d=1}^{D}\int_{-\infty}^{\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx \\
\overset{(1)}{\leq}&\prod_{d=1}^{D}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left\{\int_{-\infty}^{(\bar{x}_d+\tilde{x}_d)/2}\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}\left[1+\frac{\left(\frac{1}{2}(\bar{x}_d-\tilde{x}_d)\right)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right. \\
&\left.+\int_{(\bar{x}_d+\tilde{x}_d)/2}^{\infty}\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}\left[1+\frac{\left(\frac{1}{2}(\bar{x}_d-\tilde{x}_d)\right)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right\} \\
=&\prod_{d=1}^{D}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)} \\
&\left\{\int_{-\infty}^{(\bar{x}_d+\tilde{x}_d)/2}\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx+\int_{(\bar{x}_d+\tilde{x}_d)/2}^{\infty}\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right\} \\
=&\prod_{d=1}^{D}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}\left\{\int_{-\infty}^{(\bar{x}_d+\tilde{x}_d)/2}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right. \\
&\left.+\int_{(\bar{x}_d+\tilde{x}_d)/2}^{+\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right\} \\
=&\prod_{d=1}^{D}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}\left\{\int_{-\infty}^{+\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right. \\
&\left.+\int_{-\infty}^{+\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^4\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx\right\} \\
\overset{(2)}{=}&C_2\prod_{d=1}^{D}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}.
\end{aligned}
$$

To explain, the reason for (1) is that (*w.l.o.g.* assume at some dimension d, $\bar{x}_d < \tilde{x}_d$) for $x \in ((\bar{x}_d + \tilde{x}_d)/2, +\infty)$, $q_d(x|\bar{\mathbf{x}}) \leq q_d((\bar{x}_d+\tilde{x}_d)/2|\bar{\mathbf{x}})$, this is because the pdf of $q_d(x|\bar{\mathbf{x}})$ keeps decreasing after crossing $\bar{x}_d$, and decreases on $((\bar{x}_d + \tilde{x}_d)/2, +\infty)$. To derive (2), let $\nu = (m-1)/2$, where m is the new parameter, and the integral becomes some constant (independent of $\bar{\mathbf{x}}$ and $\tilde{\mathbf{x}}$) times the integral of the pdf of a t distribution on $(-\infty, +\infty)$. Summarizing all inequalities together, we have

$$
\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] \leq C_1\sqrt{C_2}\prod_{d=1}^{D}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)},
$$

where taking $C = C_1\sqrt{C_2}$ finishes the proof of the upper bound in the proposition.

Now we further show that this implies $\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] = \Omega\left(\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|^{-(\nu+1)}\right)$. Note that $\max_d\{(\bar{x}_d - \tilde{x}_d)^2\} \le \|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\| \le D\max_d\{(\bar{x}_d-\tilde{x}_d)^2\}$, and

$$\left[1+\frac{(\bar{x}_d-\tilde{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-\frac{\nu+1}{2}} < 1 \quad \text{for all } d.$$

Thus, we have

$$\begin{aligned}\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] &\le C\prod_{d=1}^{D}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)} \le C\left[1+\frac{\max_d\{(\bar{x}_d-\tilde{x}_d)^2\}}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)} \\ &\le C\left[1+\frac{\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|}{4D(\nu-2)\sigma_T^2}\right]^{-(\nu+1)} \le \frac{C}{(4D(\nu-2)\sigma_T^2)^{-(\nu+1)}}\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|^{-(\nu+1)},\end{aligned}$$

Proof of the lower bound:

For the lowerbound, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})] = \int \frac{q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})}{\sum_{n=1}^{N} q(\mathbf{x}|\mathbf{x}_0^{(n)})}d\mathbf{x} \ge \frac{1}{N}\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})d\mathbf{x}$$

Note that

$$\begin{aligned}&\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})d\mathbf{x} \\ =&\prod_{d=1}^{D}\int_{-\infty}^{\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}dx,\end{aligned}$$

and we investigate the lower bound of the integral for each single dimension d.

When $\bar{x}_d = \tilde{x}_d$, the integral above becomes

$$\int_{-\infty}^{\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx,$$

which has an explicit expression (note that the expression inside the integral is just a constant times the pdf of a t distribution):

$$\int_{-\infty}^{\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)}dx = \left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2\frac{\Gamma(\nu+1/2)}{\Gamma(\nu+1)}\sqrt{\pi(\nu-2)\sigma_T^2}.$$

For simplicity, we denote it as A.

When $|\bar{x}_d - \tilde{x}_d| \to \infty$ (w.l.o.g. assume $\bar{x}_d < \tilde{x}_d$), we have

$$\begin{aligned}
&\int_{-\infty}^{\infty} \left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx \\
\geq &\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left\{\int_{\bar{x}_d}^{(\bar{x}_d+\tilde{x}_d)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \left[1+\frac{((\bar{x}_d-\tilde{x}_d)/2)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)} dx\right. \\
&\left.+\int_{(\bar{x}_d+\tilde{x}_d)/2}^{\tilde{x}_d} \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \left[1+\frac{((\bar{x}_d-\tilde{x}_d)/2)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx\right\} \qquad \text{(EC.17)} \\
= &\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left[1+\frac{(\bar{x}_d-\tilde{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \\
&\left\{\int_{\bar{x}_d}^{(\bar{x}_d+\tilde{x}_d)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx + \int_{(\bar{x}_d+\tilde{x}_d)/2}^{\tilde{x}_d} \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx\right\}
\end{aligned}$$

Note that for $x \in [\bar{x}_d, (\bar{x}_d+\tilde{x}_d)/2]$, we have $\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \geq \left[1+\frac{(\bar{x}_d-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}$; and for $x \in [(\bar{x}_d+\tilde{x}_d)/2, \tilde{x}_d]$, we have $\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \geq \left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}$. Therefore, we further have

$$\begin{aligned}
&\left\{\int_{\bar{x}_d}^{(\bar{x}_d+\tilde{x}_d)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx + \int_{(\bar{x}_d+\tilde{x}_d)/2}^{\tilde{x}_d} \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx\right\} \\
\geq &\frac{|\tilde{x}_d-\bar{x}_d|}{2}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}.
\end{aligned}$$

Combining with (EC.17), we have

$$\begin{aligned}
&\int_{-\infty}^{\infty} \left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx \\
\geq &\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left[1+\frac{(\bar{x}_d-\tilde{x}_d)^2}{4(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \frac{|\tilde{x}_d-\bar{x}_d|}{2}\left[1+\frac{(\tilde{x}_d-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}.
\end{aligned}$$

As $|\bar{x}_d - \tilde{x}_d| \to \infty$, the lower bound above has order $|\bar{x}_d-\tilde{x}_d|^{-(2\nu+1)}$, i.e., there exists a constant C and $\delta > 0$ such that

$$\int_{-\infty}^{\infty} \left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx \geq C|\bar{x}_d-\tilde{x}_d|^{-(2\nu+1)}.$$

as long as $|\bar{x}_d - \tilde{x}_d| \geq \delta$.

Recall that when $|\bar{x}_d - \tilde{x}_d| = 0$, the integral above equals to $A > 0$, and the integral is apparently strictly positive for any $\bar{x}_d$ and $\tilde{x}_d$. Therefore, we have the following lower bound for any $\bar{x}_d$ and $\tilde{x}_d$:

$$\int_{-\infty}^{\infty} \left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2 \left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} \left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2} dx \geq \min\{A, C|\bar{x}_d-\tilde{x}_d|^{-(2\nu+1)}\}.$$

Combining all the dimensions, we have

$$\begin{aligned}
&\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})d\mathbf{x} \\
=&\prod_{d=1}^{D}\int_{-\infty}^{\infty}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_T\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})}\right)^2\left[1+\frac{(x-\bar{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}\left[1+\frac{(x-\tilde{x}_d)^2}{(\nu-2)\sigma_T^2}\right]^{-(\nu+1)/2}dx \qquad \text{(EC.18)}\\
\geq&\prod_{d=1}^{D}\min\{A, C|\bar{x}_d-\tilde{x}_d|^{-(2\nu+1)}\}.
\end{aligned}$$

As $\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|\to\infty$, we have $\int q(\mathbf{x}|\bar{\mathbf{x}})q(\mathbf{x}|\tilde{\mathbf{x}})d\mathbf{x}\geq C_1\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|^{-(2\nu+1)D}$, where C_1 is a constant and D is the dimension of $\mathbf{x}$.

B.7. Proof of Proposition 4

Proof: We prove this result by leveraging Lemma 2 and relating the distance between data points to tail probability under the heavy-tailed distribution assumption.

Recall that the definition of $\partial\mathcal{S}'_p$ is the boundary of $\mathcal{S}'_p=\{\mathbf{x}|\|\mathbf{x}\|>\mathrm{VaR}_{data}(p)\}$. Under the heavy-tailed distribution assumption with tail index α, for tail points $\tilde{\mathbf{x}}\in\partial\mathcal{S}'_p$, we have $\mathbb{P}(\|X\|>\|\tilde{\mathbf{x}}\|)\sim p$. This implies that $\|\tilde{\mathbf{x}}\|\sim p^{-1/\alpha}$ as $p\to 0$.

For data points $\bar{\mathbf{x}}$ from the central region and tail points $\tilde{\mathbf{x}}\in\partial\mathcal{S}'_p$, we can bound the distance as:

$$\begin{aligned}
\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|&\geq\|\tilde{\mathbf{x}}\|-\|\bar{\mathbf{x}}\|\sim p^{-1/\alpha}\\
\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|&\leq 2\|\tilde{\mathbf{x}}\|\sim p^{-1/\alpha}
\end{aligned}$$

when $\tilde{\mathbf{x}}$ is sufficiently far in the tail.

From Lemma 2, we have:

$$\begin{aligned}
\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})]&\geq C_3\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|^{-(2\nu+1)D}\\
\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})]&\leq C_4\|\bar{\mathbf{x}}-\tilde{\mathbf{x}}\|^{-(\nu+1)}.
\end{aligned}$$

Substituting our distance bound, there exists constants C'_3 and C'_4 such that for any $\tilde{\mathbf{x}}\in\partial\mathcal{S}'_p$, we have

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})]\geq C'_3(p^{-1/\alpha})^{-(2\nu+1)D}=C'_3p^{(2\nu+1)D/\alpha}.$$

and

$$\mathbb{E}_{q(\mathbf{x}|\bar{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})]\leq C'_4(p^{-1/\alpha})^{-(\nu+1)}=C'_4p^{(\nu+1)/\alpha}$$

B.8. Derivation of the Conditional Score Identity

Here we show that in Algorithm 3, the score function used to update the target dimensions reduces to the conditional score function. Since the unconditional network $\mathbf{s}_\theta(\tilde{\mathbf{x}}_t^{\text{all}},\sigma_t)$ approximates the joint score function, its output can be decomposed as

$$\nabla_{\tilde{\mathbf{x}}_t^{\text{all}}}\log q_{\tilde{\mathbf{X}}_t^{\text{all}}}(\tilde{\mathbf{x}}_t^{\text{all}})=\begin{pmatrix}\nabla_{\tilde{\mathbf{x}}_t^{\text{cond}}}\log q_{(\tilde{\mathbf{X}}_t^{\text{cond}},\tilde{\mathbf{X}}_t^{\text{targ}})}(\tilde{\mathbf{x}}_t^{\text{cond}},\tilde{\mathbf{x}}_t^{\text{targ}})\\ \nabla_{\tilde{\mathbf{x}}_t^{\text{targ}}}\log q_{(\tilde{\mathbf{X}}_t^{\text{cond}},\tilde{\mathbf{X}}_t^{\text{targ}})}(\tilde{\mathbf{x}}_t^{\text{cond}},\tilde{\mathbf{x}}_t^{\text{targ}})\end{pmatrix},$$

where we write $\tilde{\mathbf{x}}_t^{\text{all}} = (\tilde{\mathbf{x}}_t^{\text{cond}}, \tilde{\mathbf{x}}_t^{\text{targ}})$. Since Algorithm 3 replaces the conditional dimensions by $\mathbf{c}$ at each step and only updates the target dimensions, the Langevin iteration at noise level σ_t is

$$\mathbf{x}_{(i)}^{\text{targ}} = \mathbf{x}_{(i-1)}^{\text{targ}} + \frac{s_t}{2}\nabla_{\tilde{\mathbf{x}}_t^{\text{targ}}} \log q_{(\tilde{\mathbf{X}}_t^{\text{cond}}, \tilde{\mathbf{X}}_t^{\text{targ}})}(\tilde{\mathbf{x}}_t^{\text{cond}} = \mathbf{c}, \tilde{\mathbf{x}}_t^{\text{targ}} = \mathbf{x}_{(i-1)}^{\text{targ}}) + \sqrt{s_i}\boldsymbol{\eta}_i^{\text{targ}}.$$

By factoring the joint density as $q_{(\tilde{\mathbf{X}}_t^{\text{cond}}, \tilde{\mathbf{X}}_t^{\text{targ}})}(\tilde{\mathbf{x}}_t^{\text{cond}}, \tilde{\mathbf{x}}_t^{\text{targ}}) = q_{\tilde{\mathbf{X}}_t^{\text{cond}}}(\tilde{\mathbf{x}}_t^{\text{cond}}) \cdot q_{\tilde{\mathbf{X}}_t^{\text{targ}}|\tilde{\mathbf{X}}_t^{\text{cond}}}(\tilde{\mathbf{x}}_t^{\text{targ}}|\tilde{\mathbf{x}}_t^{\text{cond}})$ and taking the gradient with respect to $\tilde{\mathbf{x}}_t^{\text{targ}}$, we obtain

$$\begin{aligned}\nabla_{\tilde{\mathbf{x}}_t^{\text{targ}}} \log q_{(\tilde{\mathbf{X}}_t^{\text{cond}}, \tilde{\mathbf{X}}_t^{\text{targ}})}(\tilde{\mathbf{x}}_t^{\text{cond}} = \mathbf{c}, \tilde{\mathbf{x}}_t^{\text{targ}}) &= \nabla_{\tilde{\mathbf{x}}_t^{\text{targ}}} \log q_{\tilde{\mathbf{X}}_t^{\text{cond}}}(\mathbf{c}) + \nabla_{\tilde{\mathbf{x}}_t^{\text{targ}}} \log q_{\tilde{\mathbf{X}}_t^{\text{targ}}|\tilde{\mathbf{X}}_t^{\text{cond}}}(\tilde{\mathbf{x}}_t^{\text{targ}}|\mathbf{c}) \\ &= \nabla_{\tilde{\mathbf{x}}_t^{\text{targ}}} \log q_{\tilde{\mathbf{X}}_t^{\text{targ}}|\tilde{\mathbf{X}}_t^{\text{cond}}}(\tilde{\mathbf{x}}_t^{\text{targ}}|\mathbf{c}),\end{aligned}$$

where the first term vanishes because $q_{\tilde{\mathbf{X}}_t^{\text{cond}}}(\mathbf{c})$ does not depend on $\tilde{\mathbf{x}}_t^{\text{targ}}$. This gives Equation (16), confirming that the Langevin dynamics in Algorithm 3 effectively uses the conditional score function.

B.9. Proof of Theorem 2

Proof: For simplicity, throughout the proof, we denote the joint distribution of $q_{(\tilde{\mathbf{X}}_1^{\text{targ}}, \tilde{\mathbf{X}}_1^{\text{cond}})}(\tilde{\mathbf{x}}_1^{\text{targ}}, \tilde{\mathbf{x}}_1^{\text{cond}})$ as $\tilde{q}_1^{\text{join}}(\tilde{\mathbf{x}}_1^{\text{targ}}, \tilde{\mathbf{x}}_1^{\text{cond}})$, and the joint distribution of real data $q_{\text{data}}(\mathbf{x}_0^{\text{targ}}, \mathbf{x}_0^{\text{cond}})$ as $q_0^{\text{join}}(\mathbf{x}_0^{\text{targ}}, \mathbf{x}_0^{\text{cond}})$.

Similarly, we denote the corresponding marginal distribution of $q_{\tilde{\mathbf{X}}_1^{\text{cond}}}(\tilde{\mathbf{x}}_1^{\text{cond}})$ as $\tilde{q}_1^{\text{marg}}(\tilde{\mathbf{x}}_1^{\text{cond}})$, and the marginal distribution of real conditions $q_{\text{data}}(\mathbf{x}_0^{\text{cond}})$ as $q_0^{\text{marg}}(\mathbf{x}_0^{\text{cond}})$.

As for the conditional distributions, we denote $q_{\tilde{\mathbf{X}}_1^{\text{targ}}|\tilde{\mathbf{X}}_1^{\text{cond}}}(\tilde{\mathbf{x}}_1^{\text{targ}}|\tilde{\mathbf{x}}_1^{\text{cond}} = \mathbf{c})$ as $\tilde{q}_1^{\text{cond}}(\tilde{\mathbf{x}}_1^{\text{targ}}|\mathbf{c})$, and $q_{\text{data}}(\mathbf{x}_0^{\text{targ}}|\mathbf{x}_0^{\text{cond}} = \mathbf{c})$ as $q_0^{\text{cond}}(\mathbf{x}_0^{\text{targ}}|\mathbf{c})$.

According to the definition of conditional distribution, we have

$$\begin{aligned}q_0^{\text{cond}}(\mathbf{x}_0^{\text{targ}}|\mathbf{c}) &= \frac{q_0^{\text{join}}(\mathbf{x}_0^{\text{targ}}, \mathbf{c})}{q_0^{\text{marg}}(\mathbf{c})}, \\ \tilde{q}_1^{\text{cond}}(\tilde{\mathbf{x}}_1^{\text{targ}}|\mathbf{c}) &= \frac{\tilde{q}_1^{\text{join}}(\tilde{\mathbf{x}}_1^{\text{targ}}, \mathbf{c})}{\tilde{q}_1^{\text{marg}}(\mathbf{c})}\end{aligned}$$

The KL-divergence between q_0^{cond} and $\tilde{q}_1^{\text{cond}}$ is given as

$$\begin{aligned}D_{\text{KL}}(q_0^{\text{cond}} \| \tilde{q}_1^{\text{cond}}) &= \int_{\mathcal{X}^{\text{targ}}} q_0^{\text{cond}}(\mathbf{x}^{\text{targ}}|\mathbf{c}) \log\left(\frac{q_0^{\text{cond}}(\mathbf{x}^{\text{targ}}|\mathbf{c})}{\tilde{q}_1^{\text{cond}}(\mathbf{x}^{\text{targ}}|\mathbf{c})}\right) \mathrm{d}\mathbf{x}^{\text{targ}} \\ &= \int_{\mathcal{X}^{\text{targ}}} \frac{q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c})}{q_0^{\text{marg}}(\mathbf{c})} \log\left(\frac{q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c})}{q_0^{\text{marg}}(\mathbf{c})} \cdot \frac{\tilde{q}_1^{\text{marg}}(\mathbf{c})}{\tilde{q}_1^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c})}\right) \mathrm{d}\mathbf{x}^{\text{targ}}\end{aligned} \tag{EC.19}$$

Now we investigate the joint and marginal distributions of the data perturbed with σ_1, i.e., $\tilde{q}_1^{\text{join}}(\tilde{\mathbf{x}}_1^{\text{targ}}, \tilde{\mathbf{x}}_1^{\text{cond}})$ and $\tilde{q}_1^{\text{marg}}(\tilde{\mathbf{x}}_1^{\text{cond}})$. Since we know that

$$\begin{aligned}\tilde{\mathbf{x}}_1^{\text{targ}} &= \mathbf{x}_0^{\text{targ}} + \sigma_1 \eta^{\text{targ}}, \\ \tilde{\mathbf{x}}_1^{\text{cond}} &= \mathbf{x}_0^{\text{cond}} + \sigma_1 \eta^{\text{cond}},\end{aligned}$$

where $\eta^{\text{all}} = (\eta^{\text{targ}}, \eta^{\text{cond}})$ is the heavy-tailed noise we have chosen to perturb the data. The dimension of η^{cond} is the same as the dimension of $\mathcal{X}^{\text{cond}}$, and the dimension of η^{targ} is the same as the dimension of $\mathcal{X}^{\text{targ}}$. For simplicity, we will denote the joint density function of $\eta^{\text{all}} = (\eta^{\text{targ}}, \eta^{\text{cond}})$ as $p_\eta^{\text{all}}(\eta^{\text{targ}}, \eta^{\text{cond}})$, and the marginal density function of η^{cond} as $p_\eta^{\text{cond}}(\eta^{\text{cond}})$. Since the perturbations to each dimension are i.i.d., we have that

$$p_\eta^{\text{all}}(\eta^{\text{targ}}, \eta^{\text{cond}}) = \prod_{i=1}^{d^{\text{targ}}} p_\eta(\eta_i^{\text{targ}}) \prod_{j=1}^{d^{\text{cond}}} p_\eta(\eta_j^{\text{cond}}),$$
$$p_\eta^{\text{cond}}(\eta^{\text{cond}}) = \prod_{j=1}^{d^{\text{cond}}} p_\eta(\eta_j^{\text{cond}}),$$

where $p_\eta(\cdot)$ denotes the density function of the 1-dimensional heavy-tailed noise added to each dimension, η_i^{targ} and η_j^{cond} denotes the i-th element of η^{targ} and the j-th element of η^{cond}, respectively. In our framework where we use student-t noise to perturb each dimension, $p_\eta(\cdot)$ is the density function of the corresponding student-t noise.

Moreover, we also have that $\mathbf{x}_0^{\text{targ}} \perp \eta^{\text{targ}}$ and $\mathbf{x}_0^{\text{cond}} \perp \eta^{\text{cond}}$. Therefore, we have

$$\tilde{q}_1^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}) = \int_{(\mathcal{X}^{\text{targ}}, \mathcal{X}^{\text{cond}})} q_0^{\text{join}}(\mathbf{u}^{\text{targ}}, \mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{all}}\left(\frac{\mathbf{x}^{\text{targ}} - \mathbf{u}^{\text{targ}}}{\sigma_1}, \frac{\mathbf{c} - \mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{targ}} + d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{targ}} \mathrm{d}\mathbf{u}^{\text{cond}}, \tag{EC.20}$$

$$\tilde{q}_1^{\text{marg}}(\mathbf{c}) = \int_{\mathcal{X}^{\text{cond}}} q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c} - \mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}}. \tag{EC.21}$$

Now denote $N_\infty(\mathbf{c}, \delta)$ as the set $\{\mathbf{x} \in \mathcal{X}^{\text{cond}} : \|\mathbf{x} - \mathbf{c}\|_\infty \le \delta\}$ (i.e., the set of x such that $\max_i |x_i - c_i| \le \delta$), and $N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)$ as the set $\{\mathbf{x} \in \mathcal{X}^{\text{targ}} \times \mathcal{X}^{\text{cond}} : \|\mathbf{x} - (\mathbf{x}^{\text{targ}}, \mathbf{c})\|_\infty \le \delta\}$. Since $\log q_0^{\text{marg}}(\cdot)$ and $\log q_0^{\text{join}}(\cdot, \cdot)$ are both uniformly continuous, we have that for each $\varepsilon > 0$, there exists a δ such that

1. for any $\mathbf{u}^{\text{cond}} \in N_\infty(\mathbf{c}, \delta)$, we have $q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \in [(1-\varepsilon) q_0^{\text{marg}}(\mathbf{c}), (1+\varepsilon) q_0^{\text{marg}}(\mathbf{c})]$,

2. for any $(\mathbf{u}^{\text{targ}}, \mathbf{u}^{\text{cond}}) \in N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)$, we have $q_0^{\text{join}}(\mathbf{u}^{\text{targ}}, \mathbf{u}^{\text{cond}}) \in [(1-\varepsilon) q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}), (1+\varepsilon) q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c})]$.

Therefore, we have a lower bound for $\tilde{q}_1^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c})$ [2]:

$$\begin{aligned}
\tilde{q}_1^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}) =& \int_{(\mathcal{X}^{\text{targ}}, \mathcal{X}^{\text{cond}})} q_0^{\text{join}}(\mathbf{u}^{\text{targ}}, \mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{all}}\left(\frac{\mathbf{x}^{\text{targ}}-\mathbf{u}^{\text{targ}}}{\sigma_1}, \frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{targ}}+d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{targ}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
\geq& \int_{N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)} q_0^{\text{join}}(\mathbf{u}^{\text{targ}}, \mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{all}}\left(\frac{\mathbf{x}^{\text{targ}}-\mathbf{u}^{\text{targ}}}{\sigma_1}, \frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{targ}}+d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{targ}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
\geq& (1-\varepsilon) q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}) \int_{N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)} \frac{p_\eta^{\text{all}}\left(\frac{\mathbf{x}^{\text{targ}}-\mathbf{u}^{\text{targ}}}{\sigma_1}, \frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{targ}}+d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{targ}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
=& (1-\varepsilon) q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}) \int_{N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)} \prod_{i=1}^{d^{\text{targ}}} \frac{p_\eta(\frac{x_i^{\text{targ}}-u_i^{\text{targ}}}{\sigma_1})}{\sigma_1} \prod_{j=1}^{d^{\text{cond}}} \frac{p_\eta(\frac{x_j^{\text{cond}}-u_j^{\text{cond}}}{\sigma_1})}{\sigma_1} \mathrm{d}\mathbf{u}^{\text{targ}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
=& (1-\varepsilon) q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}) \prod_{i=1}^{d^{\text{targ}}} \int_{-\delta}^{+\delta} \frac{1}{\sigma_1} p_\eta(\frac{v_i^{\text{targ}}}{\sigma_1}) \mathrm{d}v_i \prod_{j=1}^{d^{\text{cond}}} \int_{-\delta}^{+\delta} \frac{1}{\sigma_1} p_\eta(\frac{v_j^{\text{cond}}}{\sigma_1}) \mathrm{d}v_j \\
=& (1-\varepsilon) q_0^{\text{join}}(\mathbf{x}^{\text{targ}}, \mathbf{c}) \left(\int_{-\frac{\delta}{\sigma_1}}^{\frac{\delta}{\sigma_1}} p_\eta(v) \mathrm{d}v \right)^{d^{\text{targ}}+d^{\text{cond}}}.
\end{aligned} \tag{EC.22}$$

Next, we obtain an upper bound for $\tilde{q}_1^{\text{marg}}(\mathbf{c})$, in which we will use the following lemma. The proof of the lemma is given at the end of this section.

LEMMA EC.1. *If $q_0^{\text{marg}}(\cdot)$ has full support and $\log q_0^{\text{marg}}(\cdot)$ is uniformly continuous, then $q_0^{\text{marg}}(\cdot)$ is bounded. In other words, there exists an $M > 0$ such that $q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \leq M$ for any $\mathbf{u}^{\text{cond}} \in \mathcal{X}^{\text{cond}}$.*

Since $\sup_{\mathbf{u}^{\text{cond}} \in \mathcal{X}^{\text{cond}}} q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \leq M$, we have

$$\begin{aligned}
\tilde{q}_1^{\text{marg}}(\mathbf{c}) =& \int_{\mathcal{X}^{\text{cond}}} q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
=& \int_{N_\infty(\mathbf{c}, \delta)} q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} + \int_{N_\infty(\mathbf{c}, \delta)^{\mathsf{C}}} q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
\leq& (1+\varepsilon) q_0^{\text{marg}}(\mathbf{c}) \int_{N_\infty(\mathbf{c}, \delta)} \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} + \int_{N_\infty(\mathbf{c}, \delta)^{\mathsf{C}}} q_0^{\text{marg}}(\mathbf{u}^{\text{cond}}) \cdot \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
\leq& (1+\varepsilon) q_0^{\text{marg}}(\mathbf{c}) \int_{N_\infty(\mathbf{c}, \delta)} \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} + M \int_{N_\infty(\mathbf{c}, \delta)^{\mathsf{C}}} \frac{p_\eta^{\text{cond}}\left(\frac{\mathbf{c}-\mathbf{u}^{\text{cond}}}{\sigma_1}\right)}{\sigma_1^{d^{\text{cond}}}} \mathrm{d}\mathbf{u}^{\text{cond}} \\
\leq& (1+\varepsilon) q_0^{\text{marg}}(\mathbf{c}) \left(\int_{-\frac{\delta}{\sigma_1}}^{\frac{\delta}{\sigma_1}} p_\eta(v) \mathrm{d}v \right)^{d^{\text{cond}}} + M \left(1 - \left(\int_{-\frac{\delta}{\sigma_1}}^{\frac{\delta}{\sigma_1}} p_\eta(v) \mathrm{d}v \right)^{d^{\text{cond}}} \right),
\end{aligned} \tag{EC.23}$$

where the last inequality can be derived by the same steps as (EC.22).

[2] we denote the complement set of $N_\infty(\mathbf{c}, \delta)$ and $N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)$ as $N_\infty(\mathbf{c}, \delta)^{\mathsf{C}}$ and $N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)^{\mathsf{C}}$, respectively. In other words, $N_\infty(\mathbf{c}, \delta)^{\mathsf{C}} = \{\mathbf{x} \in \mathcal{X}^{\text{cond}} : \|\mathbf{x} - \mathbf{c}\|_\infty > \delta\}$ and $N_\infty((\mathbf{x}^{\text{targ}}, \mathbf{c}), \delta)^{\mathsf{C}} = \{\mathbf{x} \in \mathcal{X}^{\text{targ}} \times \mathcal{X}^{\text{cond}} : \|\mathbf{x} - (\mathbf{x}^{\text{targ}}, \mathbf{c})\|_\infty > \delta\}$.

For simplicity, let's denote $\int_{-\frac{\delta}{\sigma_1}}^{\frac{\delta}{\sigma_1}} p_\eta(v)\mathrm{d}v$ as $G(\frac{\delta}{\sigma_1})$. Notice that this is just $\mathbb{P}(|\eta| \le \frac{\delta}{\sigma_1})$. Thus, for any fixed δ, we have $\lim_{\sigma_1\to 0} G(\frac{\delta}{\sigma_1}) = 1$. Applying (EC.22) and (EC.23), the term inside the logarithm of $D_{\mathrm{KL}}(q_0^{\mathrm{cond}} \| \tilde{q}_1^{\mathrm{cond}})$ is

$$\begin{aligned}
\frac{q_0^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}{q_0^{\mathrm{marg}}(\mathbf{c})} \cdot \frac{\tilde{q}_1^{\mathrm{marg}}(\mathbf{c})}{\tilde{q}_1^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})} \le & \frac{q_0^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}{q_0^{\mathrm{marg}}(\mathbf{c})} \cdot \frac{(1+\varepsilon)q_0^{\mathrm{marg}}(\mathbf{c})G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}} + M\left(1-G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}}\right)}{(1-\varepsilon)q_0^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}+d^{\mathrm{cond}}}} \\
= & \frac{(1+\varepsilon)G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}} + M\frac{\left(1-G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}}\right)}{q_0^{\mathrm{marg}}(\mathbf{c})}}{(1-\varepsilon)G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}+d^{\mathrm{cond}}}} \\
= & \frac{1+\varepsilon}{(1-\varepsilon)G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}}} + \frac{M\left(1-G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}}\right)}{(1-\varepsilon)q_0^{\mathrm{marg}}(\mathbf{c})G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}+d^{\mathrm{cond}}}},
\end{aligned} \tag{EC.24}$$

and notice that the right hand side is not dependent on $\mathbf{x}^{\mathrm{targ}}$.

Therefore, we have

$$\begin{aligned}
D_{\mathrm{KL}}(q_0^{\mathrm{cond}} \| \tilde{q}_1^{\mathrm{cond}}) = & \int_{\mathcal{X}^{\mathrm{targ}}} \frac{q_0^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}{q_0^{\mathrm{marg}}(\mathbf{c})} \log\left(\frac{q_0^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}{q_0^{\mathrm{marg}}(\mathbf{c})} \cdot \frac{\tilde{q}_1^{\mathrm{marg}}(\mathbf{c})}{\tilde{q}_1^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}\right) \mathrm{d}\mathbf{x}^{\mathrm{targ}} \\
\le & \sup_{\mathbf{x}^{\mathrm{targ}}\in\mathcal{X}^{\mathrm{targ}}} \log\left(\frac{q_0^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}{q_0^{\mathrm{marg}}(\mathbf{c})} \cdot \frac{\tilde{q}_1^{\mathrm{marg}}(\mathbf{c})}{\tilde{q}_1^{\mathrm{join}}(\mathbf{x}^{\mathrm{targ}},\mathbf{c})}\right) \\
\le & \log\left(\frac{1+\varepsilon}{(1-\varepsilon)G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}}} + \frac{M\left(1-G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}}\right)}{(1-\varepsilon)q_0^{\mathrm{marg}}(\mathbf{c})G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}+d^{\mathrm{cond}}}}\right).
\end{aligned} \tag{EC.25}$$

Moreover, recall that for fixed ε and δ, we have $\lim_{\sigma_1\to 0} G(\frac{\delta}{\sigma_1}) = 1$, which means that

$$\begin{aligned}
\lim_{\sigma_1\to 0} \frac{1+\varepsilon}{(1-\varepsilon)G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}}} &= \frac{1+\varepsilon}{1-\varepsilon}, \\
\lim_{\sigma_1\to 0} \frac{M\left(1-G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}}\right)}{(1-\varepsilon)q_0^{\mathrm{marg}}(\mathbf{c})G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}+d^{\mathrm{cond}}}} &= 0.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\lim_{\sigma_1\to 0} D_{\mathrm{KL}}(q_0^{\mathrm{cond}} \| \tilde{q}_1^{\mathrm{cond}}) \le & \lim_{\sigma_1\to 0} \log\left(\frac{1+\varepsilon}{(1-\varepsilon)G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}}} + \frac{M\left(1-G(\frac{\delta}{\sigma_1})^{d^{\mathrm{cond}}}\right)}{(1-\varepsilon)q_0^{\mathrm{marg}}(\mathbf{c})G(\frac{\delta}{\sigma_1})^{d^{\mathrm{targ}}+d^{\mathrm{cond}}}}\right) \\
= & \log\frac{1+\varepsilon}{1-\varepsilon}.
\end{aligned} \tag{EC.26}$$

Since this is true for arbitrary $\varepsilon > 0$, letting $\varepsilon \to 0$, we have

$$\lim_{\sigma_1\to 0} D_{\mathrm{KL}}(q_0^{\mathrm{cond}} \| \tilde{q}_1^{\mathrm{cond}}) = 0. \tag{EC.27}$$

B.9.1. Proof of Lemma EC.1

Proof: Since $\log q_0^{\text{marg}}(\cdot)$ is uniformly continuous, we have that for each $0<\varepsilon<1$, there exists a δ such that for any $\mathbf{u},\mathbf{v}\in\mathcal{X}^{\text{cond}}$ such that $\|\mathbf{u}-\mathbf{v}\|_\infty\le\delta$, we have

$$q_0^{\text{marg}}(\mathbf{u})\in[(1-\varepsilon)q_0^{\text{marg}}(\mathbf{v}),(1+\varepsilon)q_0^{\text{marg}}(\mathbf{v})].$$

Suppose in contrast that $q_0^{\text{marg}}(\cdot)$ is not bounded, then there exists $\mathbf{v}\in\mathcal{X}^{\text{cond}}$ such that $q_0^{\text{marg}}(\mathbf{v})>\frac{2}{(1-\varepsilon)(2\delta)^{d^{\text{cond}}}}$. Thus, for any $\|\mathbf{u}-\mathbf{v}\|_\infty\le\delta$, we have $q_0^{\text{marg}}(\mathbf{u})\ge(1-\varepsilon)q_0^{\text{marg}}(\mathbf{v})>\frac{2}{(2\delta)^{d^{\text{cond}}}}$.

Since $q_0^{\text{marg}}(\cdot)$ has full support, we have

$$\int_{\mathcal{X}^{\text{cond}}}q_0^{\text{marg}}(\mathbf{u})\mathrm{d}\mathbf{u}\ge\int_{N_\infty(\mathbf{v},\delta)}q_0^{\text{marg}}(\mathbf{u})\mathrm{d}\mathbf{u}\ge\frac{2}{(2\delta)^{d^{\text{cond}}}}\int_{N_\infty(\mathbf{v},\delta)}\mathrm{d}\mathbf{u}=\frac{2}{(2\delta)^{d^{\text{cond}}}}(2\delta)^{d^{\text{cond}}}=2.$$

However, since $q_0^{\text{marg}}(\cdot)$ is a probability density function, we must have $\int_{\mathcal{X}^{\text{cond}}}q_0^{\text{marg}}(\mathbf{u})\mathrm{d}\mathbf{u}=1$, which contradicts to the inequality above.

Therefore, $q_0^{\text{marg}}(\cdot)$ must be bounded.

B.10. Proof of Proposition 5

Proof: We need to prove that for any $\varepsilon>0$, there exists $\delta>0$ such that for any $\|\mathbf{x}-\mathbf{y}\|\le\delta$, we have

$$|\log f(\mathbf{x})-\log f(\mathbf{y})|\le\varepsilon.$$

First, since the distribution has polynomial tail order, we have

$$\lim_{\|x\|\to\infty}\frac{f(x)}{\|x\|^{-k}}=C.$$

Therefore, there exists $r>1$ such that for any $\mathbf{x}$ with $\|\mathbf{x}\|\ge r$, we have

$$C\left(1-\frac{e^{\frac{\varepsilon}{2}}-1}{e^{\frac{\varepsilon}{2}}+1}\right)\le\frac{f(\mathbf{x})}{\|\mathbf{x}\|^{-k}}\le C\left(1+\frac{e^{\frac{\varepsilon}{2}}-1}{e^{\frac{\varepsilon}{2}}+1}\right),$$

meaning that

$$\log\left(C-C\frac{e^{\frac{\varepsilon}{2}}-1}{e^{\frac{\varepsilon}{2}}+1}\right)-k\log\|\mathbf{x}\|\le f(\mathbf{x})\le\log\left(C+C\frac{e^{\frac{\varepsilon}{2}}-1}{e^{\frac{\varepsilon}{2}}+1}\right)-k\log\|\mathbf{x}\|,$$

Thus, for any $\mathbf{x}$ and $\mathbf{y}$ such that $\|\mathbf{x}\|\ge r$ and $\|\mathbf{y}\|\ge r$, we have

$$\begin{aligned}|\log f(\mathbf{x})-\log f(\mathbf{y})|\le&\log\left(C+C\frac{e^{\frac{\varepsilon}{2}}-1}{e^{\frac{\varepsilon}{2}}+1}\right)-\log\left(C-C\frac{e^{\frac{\varepsilon}{2}}-1}{e^{\frac{\varepsilon}{2}}+1}\right)+k\left|\log\|\mathbf{x}\|-\log\|\mathbf{y}\|\right|\\=&\log\left(C\frac{2e^{\frac{\varepsilon}{2}}}{e^{\frac{\varepsilon}{2}}+1}\right)-\log\left(C\frac{2}{e^{\frac{\varepsilon}{2}}+1}\right)+k\left|\log\|\mathbf{x}\|-\log\|\mathbf{y}\|\right|\\=&\frac{\varepsilon}{2}+k\left|\log\|\mathbf{x}\|-\log\|\mathbf{y}\|\right|\le\frac{\varepsilon}{2}+k\left|\|\mathbf{x}\|-\|\mathbf{y}\|\right|\le\frac{\varepsilon}{2}+k\|\mathbf{x}-\mathbf{y}\|.\end{aligned}\tag{EC.28}$$

Therefore, when $\|\mathbf{x}\|\ge r$ and $\|\mathbf{y}\|\ge r$, as long as $\|\mathbf{x}-\mathbf{y}\|\le\frac{\varepsilon}{4k}$, we have $|\log f(\mathbf{x})-\log f(\mathbf{y})|\le\frac{3}{4}\varepsilon$.

On the other hand, since $\{\mathbf{x} : \|\mathbf{x}\| \le r\}$ is a compact set, we know that a continuous $\log f(x)$ implies that it is also uniformly continuous on $\{\mathbf{x} : \|\mathbf{x}\| \le r\}$. Thus, there exists δ_1 such that, for all $\|\mathbf{x}\| \le r$ and $\|\mathbf{y}\| \le r$ with $\|\mathbf{x}-\mathbf{y}\| \le \delta_1$, we have $|\log f(\mathbf{x}) - \log f(\mathbf{y})| \le \frac{1}{4}\varepsilon$.

Therefore, for arbitrary $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ with $\|\mathbf{x}-\mathbf{y}\| \le \min\{\frac{\varepsilon}{4k}, \delta_1\}$, we have

$$|\log f(\mathbf{x}) - \log f(\mathbf{y})| \le \frac{1}{4}\varepsilon + \frac{3}{4}\varepsilon = \varepsilon,$$

which implies that $\log f(x)$ is uniformly continuous.

Appendix C: Framework Details

In this section, we describe the details of our framework, including the architectures of our score network $\mathbf{s}_\theta(\mathbf{x}^{\text{all}}, \sigma)$, as well as the training and generation details.

C.1. Architecture

Our architecture is based on the settings for forecasting of CSDI (Tashiro et al. (2021)). Details of the network structure is provided in Figure EC.1.

Input to the Network. Here we emphasize the input of our network. Following the ideas in Tashiro et al. (2021), the input has three components: data input (including condition data $\mathbf{x}^{\text{cond}}$ and target data $\mathbf{x}^{\text{targ}}$), noise label t (corresponding to noise level σ_t), as well as side information including time embedding and feature embedding.

As for data input, we concatenate the condition dimensions $\mathbf{x}^{\text{cond}} \in \mathcal{X}^{\text{cond}}$ with the target dimensions $\mathbf{x}^{\text{targ}} \in \mathcal{X}^{\text{targ}}$, creating a full matrix $\mathbf{x}^{\text{all}} = (\mathbf{x}^{\text{cond}}, \mathbf{x}^{\text{targ}})$. For the two experiments mentioned in Section 5, after data normalization, we have $\mathcal{X}^{\text{cond}} = \mathcal{X}^{\text{targ}} = \mathbb{R}^{1\times D}$, where $D = 10$ for the correlated Pareto experiment and $D = 20$ for the queueing system experiment. In other words, the condition space is the same as the target space, and thus the concatenated data $\mathbf{x}^{\text{all}} \in \mathbb{R}^{2\times D}$, with the first element being the condition space and the second element being the target space. We regard this $2 \times D$ total input matrix as "time series" data, with time length being 2 and number of features being D. In other words, we regard the conditional data $\mathbf{x}^{\text{cond}}$ as the observation of the D features at time 0, and $\mathbf{x}^{\text{targ}}$ as the observation of the D features at time 1.[3] In this way, we can easily fit our settings into the CSDI architecture.

C.2. Training Process.

In the training process, $\mathbf{x}^{\text{cond}}$ and $\mathbf{x}^{\text{targ}}$ are perturbed together with the same noise level σ_t, and we input the perturbed concatenated data $\tilde{\mathbf{x}}_t^{\text{all}} = (\tilde{\mathbf{x}}_t^{\text{cond}}, \tilde{\mathbf{x}}_t^{\text{targ}})$ into our network $\mathbf{s}_\theta(\mathbf{x}_t^{\text{all}}, \sigma_t)$. The output of the network is also a $2 \times D$ matrix, aiming to approximate the score function of the perturbed data distribution $\tilde{\mathbf{x}}_t^{\text{all}}$, namely $\nabla_{\tilde{\mathbf{x}}_t^{\text{all}}} q_t(\tilde{\mathbf{x}}_t^{\text{all}})$. In other words, we hope to match the score of the *joint distribution* of $(\tilde{\mathbf{x}}_t^{\text{cond}}, \tilde{\mathbf{x}}_t^{\text{targ}})$, and the loss function is an unified loss function across all $\{\sigma_t\}_{t=1}^T$, as described in (14).

[3] In fact, in the queueing system experiment, our data is already time series data, with $\mathbf{x}^{\text{cond}}$ being the numbers of customers at $\tau_0 = 50$ and $\mathbf{x}^{\text{targ}}$ being the numbers of customers at $\tau_1 = 100$. For consistency, we also regard our copula data as "time series" so that the same architecture can be applied.

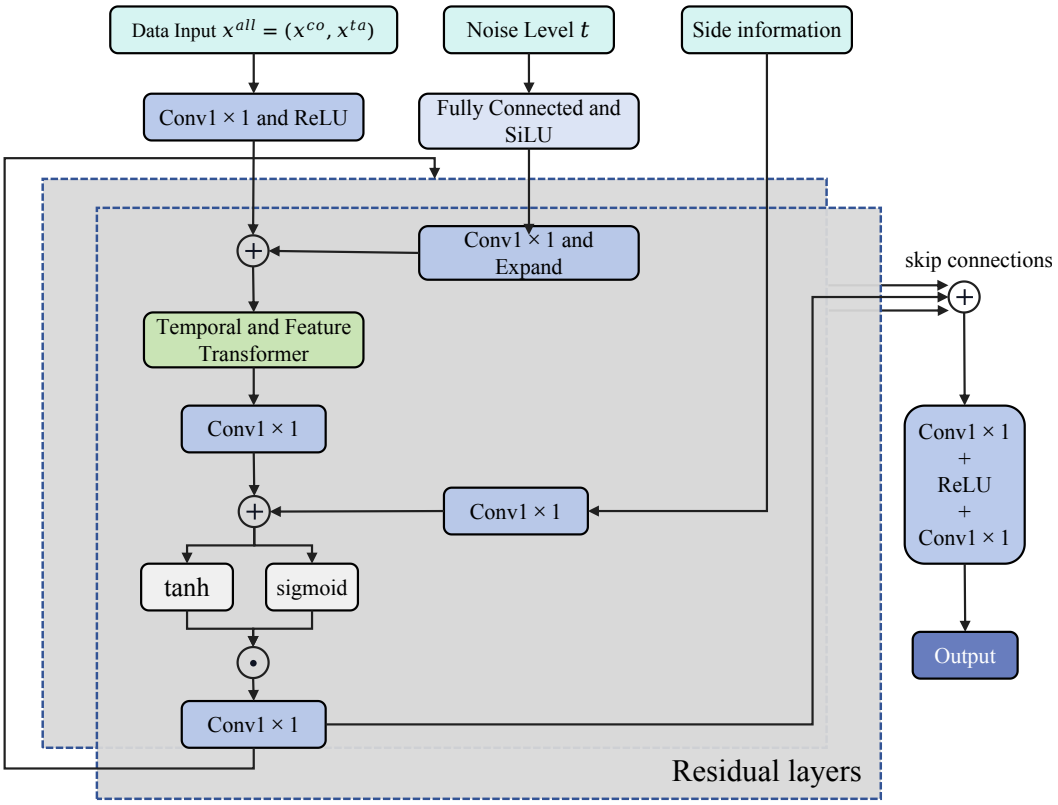


Figure EC.1 The Architecture of our networks. This architecture is based on the settings of CSDI (Tashiro et al. (2021))

C.3. Generation Process.

After the network $\mathbf{s}_\theta$ is trained, we use *annealed Langevin dynamics* with conditional constraints for the generation process, as described in Algorithm 3. The difference with the standard Langevin dynamics (shown in Algorithm 1) is that, after each step i, we manually replace the conditional dimensions of the generated data $\hat{\mathbf{x}}_i^{\text{cond}}$ with the real value of the condition $\mathbf{x}_i^{\text{cond}}$ before going to the next step. This achieves our goal of "generating conditional distribution from joint distribution". Also, $\hat{\mathbf{x}}_0$ is no longer initialized from $\mathcal{N}(0,1)$. Instead, it is drawn from a pre-specified heavy-tailed distribution, such as t-distribution.

Appendix D: Illustrative Experiments on the Choice of ν of the Perturbation Noise

This section investigates how the choice of the degrees of freedom parameter ν in the perturbation noise affects the performance of our SHD method proposed in Section 4. We conduct controlled experiments on a one-dimensional synthetic dataset to isolate the effect of this key hyperparameter. For more comprehensive experiments on realistic, multi-dimensional datasets, please refer to Section 5.

D.1. Dataset and Experimental Setup

Dataset: We construct a synthetic dataset by sampling from a one-dimensional location-scale t-distribution $\mathcal{T}(\nu, \mu, \tau^2)$ with parameters $\nu = 6$, $\mu = 0$, and $\tau^2 = \frac{1}{3}$. The scale parameter τ^2 is specifically chosen to ensure unit variance, providing a standardized setting for comparison. This choice of $\nu = 6$ corresponds to a tail order of $\alpha = 6$, representing heavy-tailed behavior. We generate 50,000 independent samples for training and 1,000 samples for testing.

Model Training: To systematically study the effect of the perturbation noise parameter, we train 9 different models: 8 SHD models with varying ν values $\{1.5, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20\}$ for the student-t perturbation noise, and one baseline model using the vanilla SGM method with Gaussian perturbation noise. All models use identical architectures and hyperparameters to ensure fair comparison.

Training Configuration: We set the number of noise levels to $T = 20$, with noise scale boundaries $\sigma_1 = 0.01$ and $\sigma_T = 5.0$. Training uses a batch size of 1,024 over 25 epochs. For sample generation, we employ annealed Langevin dynamics with 100 steps and step size 1×10^{-4} for each noise level t (from $T = 20$ down to 1), generating 1,000 samples for evaluation.

D.2. Evaluation Metrics

We assess model performance using two complementary metrics that capture different aspects of distributional accuracy:

Following the evaluation framework in Section 5, we compute (i) the Wasserstein distance between generated samples and the true distribution, measuring overall distributional similarity, and (ii) the absolute percentage error (APE) of the Value at Risk (VaR) at different confidence levels ($p = 0.95, 0.99, 0.995$), assessing tail behavior accuracy. For detailed metric definitions, see Section 5.2.2.[4] Lower values indicate better performance for both metrics.

D.3. Key Findings

Table EC.1 reveals several important insights about the choice of ν:

Performance stability across moderate ν values: SHD models with $\nu \in \{3, 4, 5, 6, 7, 8, 9\}$ achieve comparable performance across both distributional similarity (Wasserstein distance ≈ 0.18-0.20) and tail accuracy (VaR errors typically below 0.4). All these models substantially outperform the vanilla SGM baseline, which shows notably higher errors particularly in tail estimation. This demonstrates the critical importance of heavy-tailed perturbation noise for modeling heavy-tailed data.

Performance degradation for $\nu \leq 2$: Models with $\nu \in \{1.5, 2\}$ exhibit significantly worse performance, with Wasserstein distances exceeding 1.0 and VaR errors reaching 0.7-1.1. This degradation occurs because

[4] In Section 5.2.2, we defined $\mathrm{MAPE}_{\mathrm{VaR}}$ for multi-dimensional data as the average APE across dimensions. Since our experiments here are one-dimensional, we report the APE directly without the need of averaging across dimensions.

Table EC.1 Wasserstein distance and APE_{VaR} of different ν values on the one-dimensional experiment.

METHOD	WASSERSTEIN DISTANCE	$p=0.95$	$p=0.99$	$p=0.995$
SHD ($\nu=1.5$)	1.077	1.763	6.173	7.926
SHD ($\nu=2.0$)	1.065	1.007	7.862	11.265
SHD ($\nu=3.0$)	0.190	0.349	0.387	0.356
SHD ($\nu=4.0$)	0.162	0.251	0.329	0.368
SHD ($\nu=5.0$)	0.196	0.278	0.204	0.130
SHD ($\nu=6.0$)	0.196	0.191	0.234	0.202
SHD ($\nu=7.0$)	0.196	0.270	0.225	0.203
SHD ($\nu=8.0$)	0.179	0.336	0.273	0.204
SHD ($\nu=9.0$)	0.205	0.295	0.187	0.136
SHD ($\nu=10.0$)	0.224	0.238	0.223	0.160
SHD ($\nu=15.0$)	0.285	0.339	0.308	0.279
SHD ($\nu=20.0$)	0.282	0.290	0.293	0.302
GAUSSIAN	0.472	0.705	0.514	0.424

student-t distributions with $\nu \leq 2$ does not have finite variance, leading to highly unstable training dynamics. Therefore, we recommend $\nu > 2$ as a practical requirement for stable SHD training.

Performance degradation for large ν: Models with $\nu > 10$ exhibit worse performance than the ones with $\nu \in \{3,4,5,6,7,8,9\}$, especially in terms of Wasserstein distance. Specifically, the Gaussian noise can be viewed as a special case of the student-t noise with $\nu = \infty$. This degradation of performance for large ν is consistent with the theoretical analysis in Proposition 4 that when the noise is much more light-tailed than the data, and tail regions are not sufficiently explored and thus the performance of SHD is degraded.

Theory-Practice Connection: The relationship between our theoretical framework and empirical results deserves careful discussion. The ground truth distribution has tail index $\alpha = 6$. According to Proposition 4, to achieve the ideal $O(p)$ decay rate for the connectivity measure $\mathbb{E}_{q(\mathbf{x}|\tilde{\mathbf{x}})}[r(\mathbf{x}|\tilde{\mathbf{x}})]$, the theory requires $\nu \in [\frac{\alpha-1}{2}, \alpha-1] = [2.5, 5]$. Within this range (i.e., $\nu \in \{3,4,5\}$), the results indeed confirm strong performance, consistent with the theoretical prediction.

Values moderately above this range (e.g., $\nu \in \{6,7,8,9\}$) also yield comparable empirical performance in our experiment. We hereby provide the explanation for this result: when $\nu > \alpha - 1 = 5$, Proposition 4 implies that the connectivity measure decays as $O(p^{(\nu+1)/\alpha})$ with an exponent $(\nu+1)/\alpha > 1$, i.e., faster than $O(p)$. In principle, this means tail exploration becomes less effective relative to the theoretical benchmark. In practice, the observed robustness for ν slightly above the theoretical range can be attributed to two factors: (i) the bounds in Proposition 4 involve multiplicative constants that may compensate for a moderately suboptimal exponent, and (ii) in this one-dimensional, single-distribution setting with ample training data, the tail region may be sufficiently covered even with a somewhat faster-decaying connectivity measure. We caution that this empirical robustness may not extend to higher-dimensional or more challenging settings, where adhering to the theoretical range becomes more important.

Importantly, the clear performance degradation for $\nu > 10$ and for Gaussian noise ($\nu = \infty$) confirms the theoretical prediction that excessively light-tailed perturbation noise fundamentally impairs tail exploration.

The theoretical range $[(\alpha-1)/2, \alpha-1]$ should therefore be understood as providing a *principled starting point* for selecting ν, with moderate deviations above the upper bound tolerable in benign settings but not guaranteed to be safe in general.

Appendix E: Experiment Details and Additional Results

In this section, we add more details about the settings of the experiments in Section 5, as well as some additional demonstrations of the results.

E.1. Correlated Pareto Sample Experiment

E.1.1. Copula Generation Method Here we describe how to generate the Pareto distributed samples with inter-correlation, as well as samples from the conditional distribution.

Generating joint samples (training set). We first sample from a multi-dimensional normal distribution $\boldsymbol{Y} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, where the diagonals of $\boldsymbol{\Sigma}$ are set to 1. This results in the marginal distribution of each element of $\boldsymbol{Y}$ following the standard normal distribution. By applying the standard normal cdf to each component, the new random vector

$$\mathbf{U} = (U_1, \cdots, U_{20}) = (\Phi(Y_1), \cdots, \Phi(Y_{20}))$$

has marginals that are uniformly distributed on $[0,1]$. Then we apply the inverse of Pareto cdf, (i.e., $F_i^{-1}(u) = (1-u)^{-\frac{1}{\alpha_i}}$), on each component of $\mathbf{U}$, which generates the new vector

$$\mathbf{X} = (X_1, \cdots, X_{20}) = (F_1^{-1}(U_1), \cdots, F_{20}^{-1}(U_{20})).$$

The marginal distribution of X_i then following Pareto distribution with tail order α_i and domain $[1, +\infty)$. And the elements of $\mathbf{X}$ are correlated as long as the original $\boldsymbol{\Sigma}$ matrix is not diagonal.

Combining all the steps above, the correlated Pareto samples are generated by

$$\mathbf{X} = (X_1, \cdots, X_{20}) = (F_1^{-1}(\Phi(Y_1)), \cdots, F_{20}^{-1}(\Phi(Y_{20}))), \tag{EC.29}$$

where $\boldsymbol{Y} = (Y_1 n \cdots, Y_{20}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$. We then shift each X_i by -1 so that the new X_i is distributed on $[0, +\infty)$.

Generating conditional samples (evaluation set). To generate samples following the conditional distribution $(X_{11}, \cdots, X_{20} \mid X_1 = x_1, \cdots, X_{10} = x_{10})$, we first calculate

$$(y_1, \cdots, y_{10}) = (\Phi^{-1}(F_1(x_1)), \cdots, \Phi^{-1}(F_{10}(x_{10}))),$$

then we sample $Y_{11}, \cdots, Y_{20}$ from the conditional distribution $(Y_{11}, \cdots, Y_{20} | Y_1 = y_1, \cdots, Y_{20} = y_{20})$. Assume that

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \quad \text{with sizes} \quad \begin{bmatrix} 10 \times 10 & 10 \times 10 \\ 10 \times 10 & 10 \times 10 \end{bmatrix},$$

then the conditional distribution is given by

$$(Y_{11}, \cdots, Y_{20} | Y_1 = y_1, \cdots, Y_{20} = y_{20}) \sim \mathcal{N}\left(\bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\Sigma}}\right),$$

where

$$\bar{\boldsymbol{\mu}} = \boldsymbol{\Sigma_{21}}\boldsymbol{\Sigma_{11}^{-1}}(y_1, \cdots, y_{10})^{\mathrm{T}}, \quad \bar{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma_{21}} - \boldsymbol{\Sigma_{21}}\boldsymbol{\Sigma_{11}^{-1}}\boldsymbol{\Sigma_{12}}.$$

Then, by applying the same transformation in (EC.29), we get

$$(X_{11}, \cdots, X_{20}) = (F_1^{-1}(\Phi(Y_{11})), \cdots, F_{20}^{-1}(\Phi(Y_{20}))), \tag{EC.30}$$

which follows the conditional distribution $(X_{11}, \cdots, X_{20} \mid X_1 = x_1, \cdots, X_{10} = x_{10})$.

E.1.2. Details about the Datasets We use the above-mentioned copula method to generate 20-dimensional Pareto samples. The Pareto parameters α_i, $i = 1, \cdots, 20$ of our dataset is randomly chosen from $\{6, 7, 8, 9, 10\}$, aiming at creating different tail orders at different dimensions. As for the $\boldsymbol{\Sigma}$, we first generate a 20×20 matrix A with all elements uniformly selected from $[-1, 1]$. Then we convert it into a symmetric positive semi-definite matrix by the transformation $\mathcal{F}(A) = \frac{1}{4}(A + A^{\mathrm{T}})(A + A^{\mathrm{T}})$. After that, we normalize the new matrix so that the diagonals are all 1's. We then round all the matrix elements to 2 decimals, and make sure that the matrix is still positive semi-definite. Then we use the selected $\boldsymbol{\alpha}$ and $\boldsymbol{\Sigma}$ to simulate 10^5 samples for the training process.

As for the evaluation, we simulate 10^4 conditional benchmark samples using the conditional generation method, with the first 10 dimensions fixed to a given vector $(x_1, \cdots, x_{10})$. The x_d's ($d = 1, \cdots, 10$) are chosen around the unconditional mean of X_d.

E.1.3. Training and Sampling Details First, we describe the hyperparameters used for the implementation of SHD. We set $\nu = 3.5$ for the student-t noises added to the real samples, and the number of noise levels $T = 20$, with $\sigma_1 = 0.01$ and $\sigma_T = 1.0$. As mentioned in Section 5, in order to increase the importance weight of data at the tail, the noise added to the d-th dimension of the sample is scaled by $2(1 + \exp(-\frac{1}{5}|x_d|))^{-1}$ for any noise level $t = 1, \cdots, T$. As for the training samples, we split them into train/validation set with a ratio of $9:1$, with the first 9×10^4 samples as the training dataset and the other 10^4 for validation after every 10 epochs. We set the batch size as 1024 and number of epochs as 20. In each epoch, we iterate along the whole training set, meaning that we have $\lceil 90000/1024 \rceil = 88$ iterations in each epoch. We use Adam optimizer with learning rate starting from 1×10^{-3} and exponentially decay to 1×10^{-5}. As for the model, we set number of residual layers as 4, residual channels as 8, and attention heads as 2. We use 32-dimensions embedding for the diffusion step. Besides, we use 16-dimensions temporal embedding and 8-dimensions feature embedding in the side information. Since the data is distributed at $[0, +\infty)$, we randomly add a negative sign to the training samples, which creates a data distribution with full support on $\mathbb{R}^{20}$.

In the generation process, the starting point $\hat{\mathbf{x}}_0$ is initialized from $\mathcal{T}(\nu = 3.5, 0, \frac{\nu-2}{\nu} = \frac{3}{7})$, which is a t-distributed variable with $\nu = 3.5$, mean 0, and variance 1. Then, for each t from $T = 20$ to 1, we set $L_t = 50$ and $\varepsilon = 0.0001$ in the annealed Langevin dynamics. We generate 10^4 samples for evaluation. We then take the absolute value of each generated samples to convert the domain back to $[0, +\infty)$.

As for the implementation of SGM, to make everything consistent and comparable, we use the same parameters and settings as SDH, but with all the noises changed to standard Gaussian, including the noises added in the training process and the initial distribution in the generation process.

E.1.4. Additional Results Here we present some additional results about the comparison between SHD and SGM, which help to demonstrace the advantages of heavy-tail noise compared with Gaussian noise.

The QQ-plots and density plots of all the 10 target dimensions (namely $X_{11}, \cdots, X_{20}$) are presented in Figure EC.2 and Figure EC.3, respectively. The QQ-plots of SHD are closer to the 45-degree line in most of the dimensions, and the SGM results have lighter tails in almost all the dimensions. This implies that heavy-tailed noise scheme generates a better result on the dimensions where Gaussian noise can fail to capture the tail pattern.

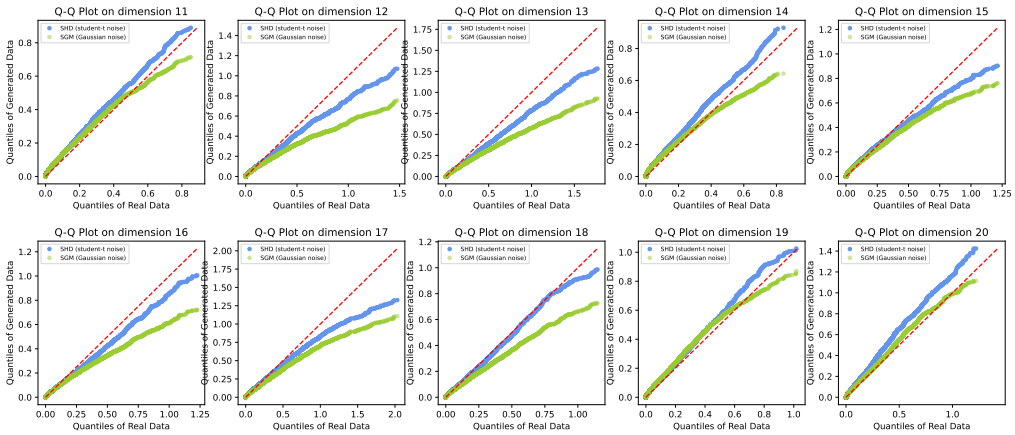


Figure EC.2 QQ-plot of between generated data and real data for all target dimensions in the correlated Pareto experiment. The green dots correspond to the standard SGM method, and the blue dots correspond to our SHD method. For each dataset, we take the smallest 99.5% of the data points for plotting, because the last 0.5% percent are empirically unstable and less informative.

In addition, we also plot the $\log(1 - F_X(x))$ for real data and generated data to directly compare the tail order, where $F_X(x)$ is the cdf of X. We present the plot for all dimensions in Figure EC.4. According to the results, using student-t noises works better in general at capturing the conditional distribution of real data, especially at the tail.

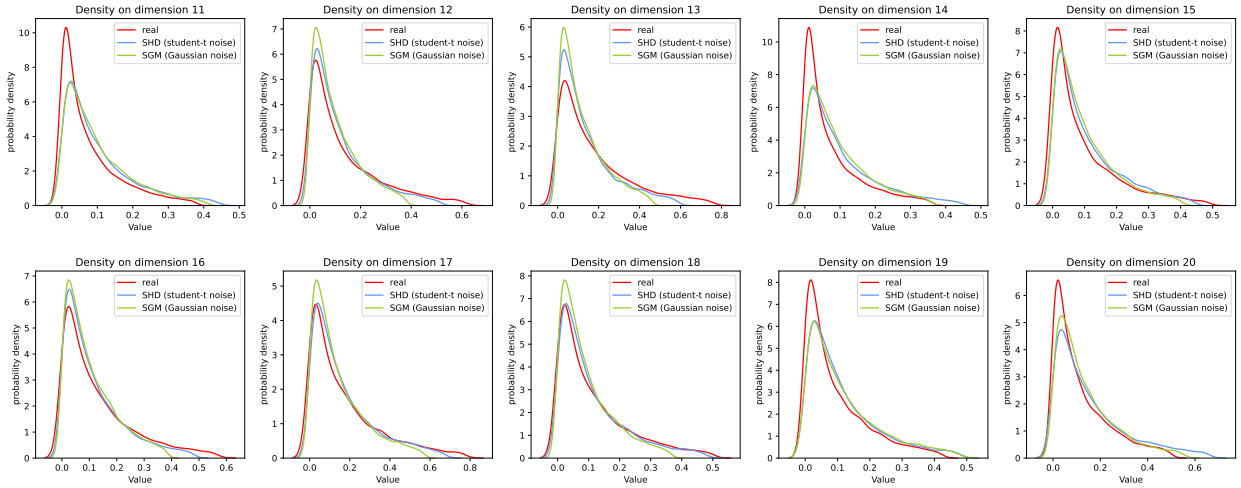


Figure EC.3 Density plot of between generated data and real data for all target dimensions in the correlated Pareto experiment. For each dataset, we take the smallest 95% of the data points for plotting, because the density at the last 5% percent is too low to be clear on the plots.

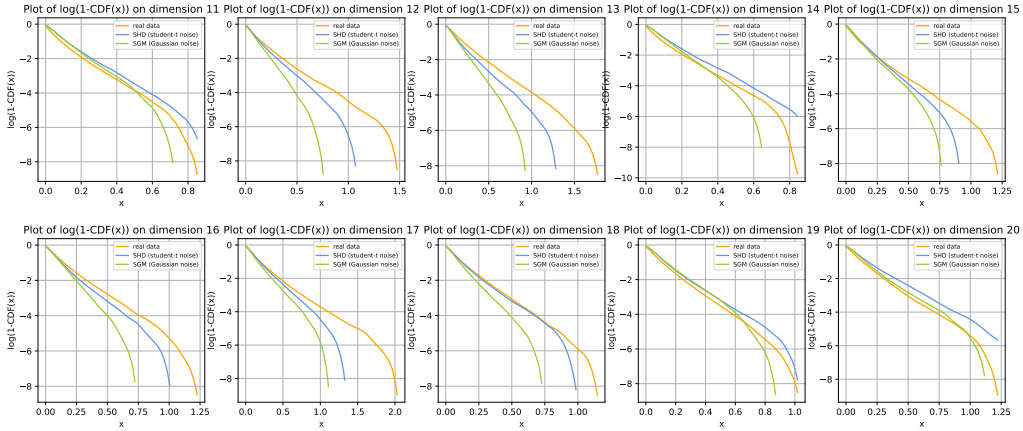


Figure EC.4 Plot of $\log(1-\mathrm{CDF}(x))$. The orange line corresponds to the real data; the green line corresponds to standard SGM method; and the blue line corresponds to our SHD method. $\mathrm{CDF}(x)$ is estimated using the kernel density method. For each dataset, we take the smallest 99.5% of the data points for plotting, because the last 0.5% percent are unstable and less informative.

E.2. Experiment on Vector Autoregressive Model

E.2.1. Details about the Vector Autoregressive Model Our synthetic dataset is generated from a 3-dimensional vector autoregressive model with $p = 3$ lags of the endogenous variable. The model can be expressed as

$$\boldsymbol{y}_t = A_1\boldsymbol{y}_{t-1} + A_2\boldsymbol{y}_{t-2} + A_3\boldsymbol{y}_{t-3} + \boldsymbol{\varepsilon}_t, \tag{EC.31}$$

where $\boldsymbol{y}_t = (y_{1t}, y_{2t}, y_{3t})$ is the endogenous variable and $\boldsymbol{\varepsilon}_t = (\varepsilon_{1t}, \varepsilon_{2t}, \varepsilon_{3t})$ is the random noise.

The coefficient matrices take the following values

$$A1 = \begin{pmatrix} 0.7 & -0.1 & 0.3 \\ -0.1 & 0.4 & 0.1 \\ 0.3 & 0.1 & -0.8 \end{pmatrix} \quad A2 = \begin{pmatrix} 0.1 & 0.05 & 0.05 \\ 0.05 & -0.4 & 0.05 \\ 0.05 & 0.05 & 0.1 \end{pmatrix} \quad A3 = \begin{pmatrix} 0.05 & 0.02 & 0.02 \\ 0.02 & 0.05 & 0.02 \\ 0.02 & 0.02 & 0.05 \end{pmatrix},$$

and the random noises $\boldsymbol{\varepsilon}_t$'s are i.i.d. across time. For each t, the three dimensions of $\boldsymbol{\varepsilon}_t$ are also i.i.d., and $\varepsilon_{it} \sim \mathcal{ZSP}(3.0)$.

E.2.2. Details about the Dataset and Evaluation In this experiment, the task is to generate the distribution of the next 5 steps conditioning on the data of the previous 5 steps. We simulate a path of $\boldsymbol{y}_t$ using the above-mentioned model. We first simulate for 1000 steps so that the system gets close to its stationary distribution. Then, starting with the current time point, we continue to simulate for 10^5 steps to create a long sample path. Then we create samples of length 10 from each 10 consecutive time points, which in total generates 10^5 samples. Each sample is a sequence of 10 vectors $\{\boldsymbol{y}_t\}_{t=1}^{10}$ with $\boldsymbol{y}_t = (y_{1t}, y_{2t}, y_{3t})$. For each sample, we use the data of the first 5 steps $\{\boldsymbol{y}_t\}_{t=1}^{5}$ as condition, and the last 5 steps $\{\boldsymbol{y}_t\}_{t=6}^{10}$ as target.

For the training process, we use the Hill estimator to estimate the tail order of the data distribution, and the estimated average tail order is 3.50. We treat ν as a tunable hyperparameter and expand our search space to $\nu \in [2.5, 5.0]$, and select the one with the best performance on the training set as our final choice. Similar to the correlated Pareto experiment, we apply the adaptive noise scaling strategy by scaling the noise added to the d-th dimension by $2(1+\exp(-\frac{1}{5}|x_d|))^{-1}$ to enhance tail exploration.

For evaluation, we generate 10^4 conditional samples, with the conditions $\{\boldsymbol{y}_t\}_{t=1}^{5}$ being fixed at a pre-specified value. We evaluate the performance on the target points $\{\boldsymbol{y}_t\}_{t=6}^{10}$, which in total contains 3 (dimension of $\boldsymbol{y}_t$) $\times$ 5 (number of steps) $= 15$ dimensions.

E.2.3. Training and Sampling Details We set $\nu = 4.5$ for the student-t noises added to the real samples in our SHD training process. The number of noise levels is selected as $T = 20$, with $\sigma_1 = 0.01$ and $\sigma_T = 1.0$. We split the 10^5 training samples into training and validation set with a ratio of $9:1$. The batch size is set to 1024 and the number of epochs is set as 40. We use Adam optimizer with learning rate starting from 1×10^{-2} and exponentially decay to 1×10^{-5}. Other hyperparameters of the model architecture are set the same as the correlated Pareto experiment, with details mentioned in Appendix E.1.3.

E.2.4. Additional Results Here we present some additional results about the comparison between SHD and SGM. The QQ-plots and density plots of (y_{1t}, y_{2t}, y_{3t}) for $t = 6, \cdots, 10$ are presented in Figure EC.5 and Figure EC.6, respectively. We can see that, in general, the SHD method performs better in capturing the tail pattern of the real data.

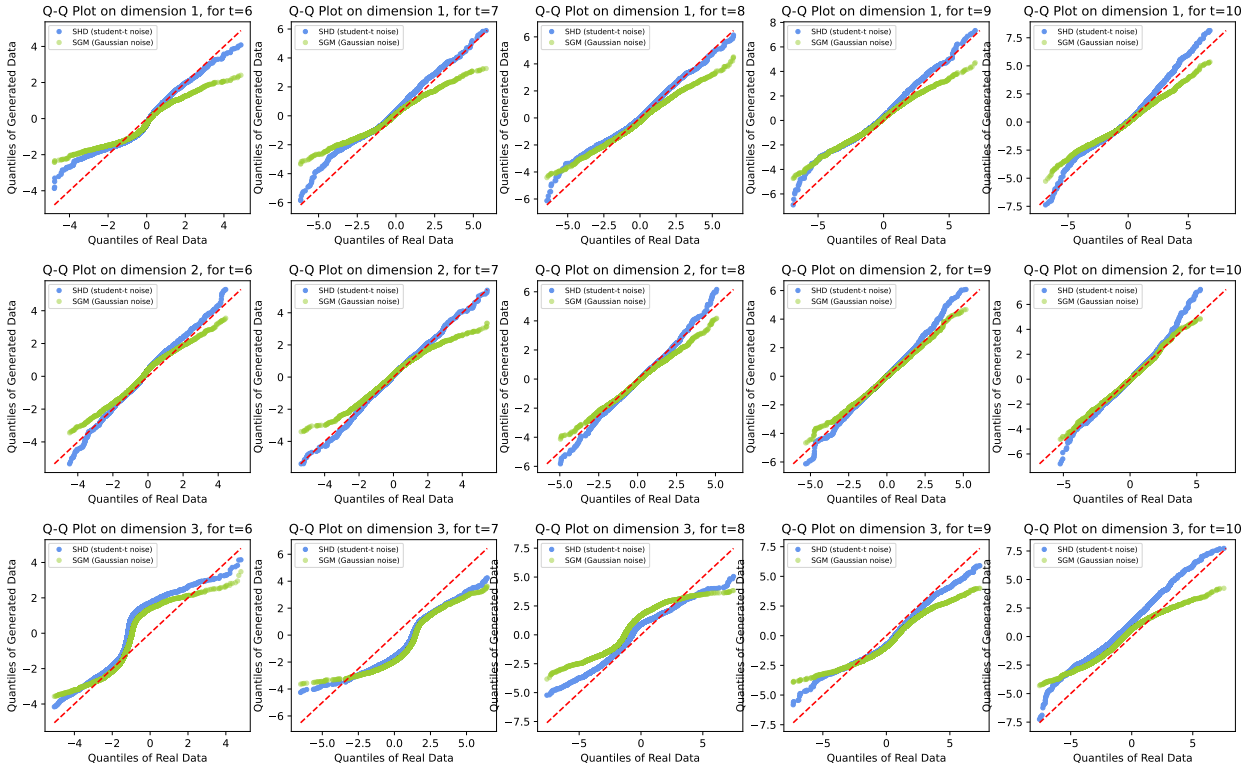


Figure EC.5 QQ-plot of between generated data and real data for all dimensions and time steps in the vector autoregressive model experiment. The green dots correspond to the standard SGM method, and the blue dots correspond to our SHD method. For each dataset, we take the smallest 99.5% of the data points for plotting, because the last 0.5% percent are empirically unstable and less informative.

E.3. Experiment on queueing System

E.3.1. Additional Details of the queueing System Our queueing system has $D = 20$ service stations. Each station d has n_d homogeneous servers. We randomly select n_d from $\{1, 2, 3, 4, 5, 6\}$ for our 20 stations. The service time S_d in station d follows log-normal distribution $\text{Lognormal}(0, \sigma_d^2)$, where σ_d^2 is randomly selected from $\{2, 3, 4, 5\}$. The exogenous arrival process of customers to each station follows a *Non-Homogeneous Compound Poisson Process* (NHCPP), with the cumulative arrival number until time τ being $Y(\tau) = \sum_{i=1}^{N(\tau)} A_i$, where $N(\tau)$ is a *Non-Homogeneous Poisson Process* with intensity

$$\lambda(\tau) = \frac{1}{10}\left(0.75 + 0.25\sin\left(\frac{2\pi\tau}{100}\right)\right),$$

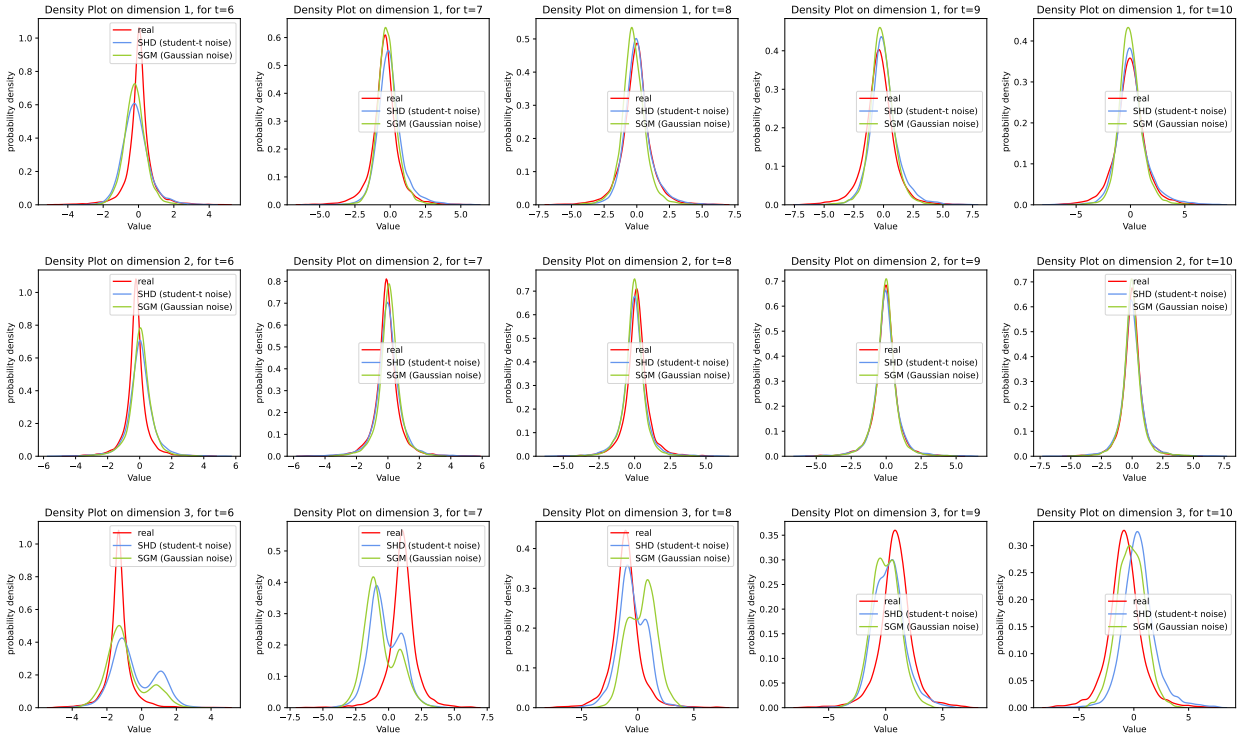


Figure EC.6 Density plot of between generated data and real data for all dimensions and time steps in the vector autoregressive model experiment. For each dataset, we take the smallest 99% of the data points for plotting, because the density at the last 1% percent is too low to be clear on the plots.

and A_i follows $\text{Poisson}(X_i)$ with X_i follows Pareto distribution $\mathcal{P}(2,1)$. The Poisson distribution is used to make sure that A_i's are integers. Thus, the distribution of both the service time and arrival rate has heavy-tail characteristics.

When a customer arrives at the service system, he randomly selects a station d to enter. At station d, if all servers are busy, he waits in a First-In-First-Out (FIFO) queue to be served, and is otherwise served immediately. A server can only serve for one customer at the same time. After finishing his service at station d, the customer either leaves the system, or immediately randomly enters another station to be served. The choice of the customer after finishing service at a certain station is governed by a transition matrix $\mathbf{P} = (p_{i,j})$, where p_{ij} is the probability that a customer leaving station i immediately goes to station j. We have $\sum_{j=1}^{20} p_{ij} \leq 1$ for any $i = 1, \cdots, 20$, with $1 - \sum_{j=1}^{20} p_{ij}$ being the probability of leaving the system. The $\mathbf{P}$ matrix in our experiment is randomly generated with the conditions above being satisfied.

We focus on the number of customers in each station at time τ. Note that the customers under service is also counted as "customer in the station".

E.3.2. Simulating the queueing System Now we briefly describe how to simulate sample paths of customer numbers in each station d from time $\tau = \tau_0$ to $\tau = \tau_1$. If number of customers at $\tau = \tau_0$ is not 0,

Algorithm 1: Simulating the Non-Homogeneous Compound Poisson Arrivals

Input: Start time τ_0, end time τ_1, intensity function $\lambda(\tau)$, maximum intensity rate $\lambda_{\max} > \max_t \lambda(t)$, Pareto distribution shape parameter α, previous arrivals (before τ_0) list $L^{\text{pre}}_{\text{arrival}}$

Output: Full sorted arrival list L^{d}_{arrival}

Initialize $\tau \leftarrow \tau_0$;
Initialize $L^{d}_{\text{arrival}} \leftarrow L^{\text{pre}}_{\text{arrival}}$;
while $\tau < \tau_1$ **do**
    Draw $u \sim \text{Uniform}(0, 1)$;
    $\tau \leftarrow \tau - \log(u)/\lambda_{\max}$;
    if $\tau < \tau_1$ **then**
        $\rho = \lambda(\tau)/\lambda_{\max}$; `// ρ is the acceptance probability`
        Draw $v \sim \text{Uniform}(0, 1)$;
        if $v < \rho$ **then**
            Generate $X \sim \mathcal{P}(2, 1)$;
            Generate $A \sim \text{Poisson}(X)$;
            for $t = T$ **to** 1 **do**
                Append (τ, d) pair into L^{d}_{arrival};
return $L^{d}_{arrival}$;

we have to input the arrival time of each customer, because the service time distribution $\text{Lognormal}(0, \sigma_d^2)$ is *not* memoryless, meaning that the remaining service time depends on how long a customer has been served. We then simulate the exogenous arrivals between τ_0 and τ_1 at each station. To simulate the Non-Homogeneous Poisson Process, we use the *thinning method*Lewis and Shedler (1979). After the exogenous arrivals are generated, we combine them with the previous arrivals (before τ_0) into a full arrival list, with each element being a (τ_{arr}, d) pair, where τ_{arr} is the arrival time and d is the station number. After that, we sort the arrival list according to arrival time τ in ascending order, creating the sorted arrival list L^{d}_{arrival}. The entire process of generating L^{d}_{arrival} is summarized in Algorithm 1

In each of the service station, we record the "next available time" τ_{next} of each server, which is the time when its current service is finished. The "next available time" is initialized as 0 for all servers. Besides, we create an "departure list" L^{d}_{depart} for each station. L^{d}_{arrival} and L^{d}_{depart} will be used to calculate number of customers. Then we create an "event list" L_{event} by concatenating all L^{d}_{arrival} and then sort by the arrival times.

Then we go through each of the elements in the event list L_{event} from the beginning to the end. For each element (τ_{arr}, d), we send it to station d, and assign this customer to the server that has the smallest "next available time". The starting time of the service is then $\tau_{\text{start}} = \max\{\tau_{\text{arr}}, \tau_{\text{next}}\}$. We then sample the service

time from $\tau_{\text{service}} = \text{Lognormal}(0, \sigma_d^2)$, and the service end time is thus $\tau_{\text{end}} = \tau_{\text{start}} + \tau_{\text{service}}$. After that, we set the "next available time" τ_{next} of this server as τ_{end}, append (τ_{end}, d) into the departure list L^d_{depart} of station d, and decide the next destination of the current customer according to the transition matrix $\mathbf{P}$. If the customer leaves the system, we go to the next element in the sorted arrival list; if the customer goes to station d', we regard this event as a "new arrival" at station d', and insert the element (τ_{end}, d') into our event list L_{event} as well as L^d_{arrival} at the right place so that the ascending order of arrival time in the two lists are kept.

We stop when the current τ in the current element is greater than τ_1, or when we have gone through all the elements in L_{arrival}. To calculate the number of customers in station d at time τ_1, we can just count the number of $(\tau_{\text{arr}}, \tau_{\text{end}})$ pairs in L^d_{event} that satisfies $\tau_{\text{arr}} \le \tau_1$ and $\tau_{\text{end}} > \tau_1$. The entire process of simulating number of customers $\mathbf{X}_{\tau_1}$ at τ_1 from τ_0 is summarized in Algorithm 2

E.3.3. Details about the Datasets We use the above-mentioned simulation method to simulate the number of customers at $\tau_0 = 50$ and $\tau_1 = 100$. For the training set, our simulation starts at τ_0 with no existing customers in the system. We simulate 10^5 samples for the training process.

As for the evaluation, we fix the number of customers at $\tau_0 = 50$ as integers around the unconditional mean of $\mathbf{X}_{\tau_0} = (X_{\tau_0,1}, \cdots, X_{\tau_0}, D)$, where $X_{\tau_0,d}$ is the number of customers in station d at time τ_0 and $D = 20$ is the number of stations. Our simulation of the evaluation set then starts at $\tau_0 = 50$, with number of existing customers fixed to the given numbers. For simplicity, the arrival time of all the existing customers are set as 50, equal to τ_0. We simulate 10^4 conditional samples for evaluation.

The Hill estimator on the training set yields an average tail order of 3.75. Following the same procedure as in the correlated Pareto experiment, we conduct a grid search over $\nu \in [2.5, 5.0]$ and select the best-performing value on the training set.

E.3.4. Training and Sampling Details For the implementation of SHD in our queueing experiment, we set $\nu = 4.5$ for the student-t noises added to the real samples, and the number of noise levels $T = 20$, with $\sigma_1 = 0.01$ and $\sigma_T = 1.0$. We split the 10^5 training samples into training and validation set with a ratio of $9:1$. The batch size is set to 1024 and number of epochs is set as 30. We use Adam optimizer with learning rate starting from 1×10^{-3} and exponentially decay to 1×10^{-6}. The hyperparameters of the model architecture are set the same as the correlated Pareto experiment, with details mentioned in Appendix E.1.3.

E.3.5. Additional Results Again, here we present some additional results about the comparison between SHD and SGM. The QQ-plots and density plots for all the 20 stations are presented in Figure EC.7 and Figure EC.8, respectively. According to the results, heavy-tailed noise scheme generates a better result on the dimensions where Gaussian noise can fail to capture the tail pattern (especially the 9th and 15th stations), while keeps working well on the dimensions where Gaussian noise already works well.

In addition, we also plot the $\log(1 - F_X(x))$ for real data and generated data to directly compare the tail order, where $F_X(x)$ is the cdf of X. We present the plot for all dimensions in Figure EC.4. In general, the $\log(1 - F_X(x))$ curve of SHD is closer to the curve of real data, especially at the tail area.

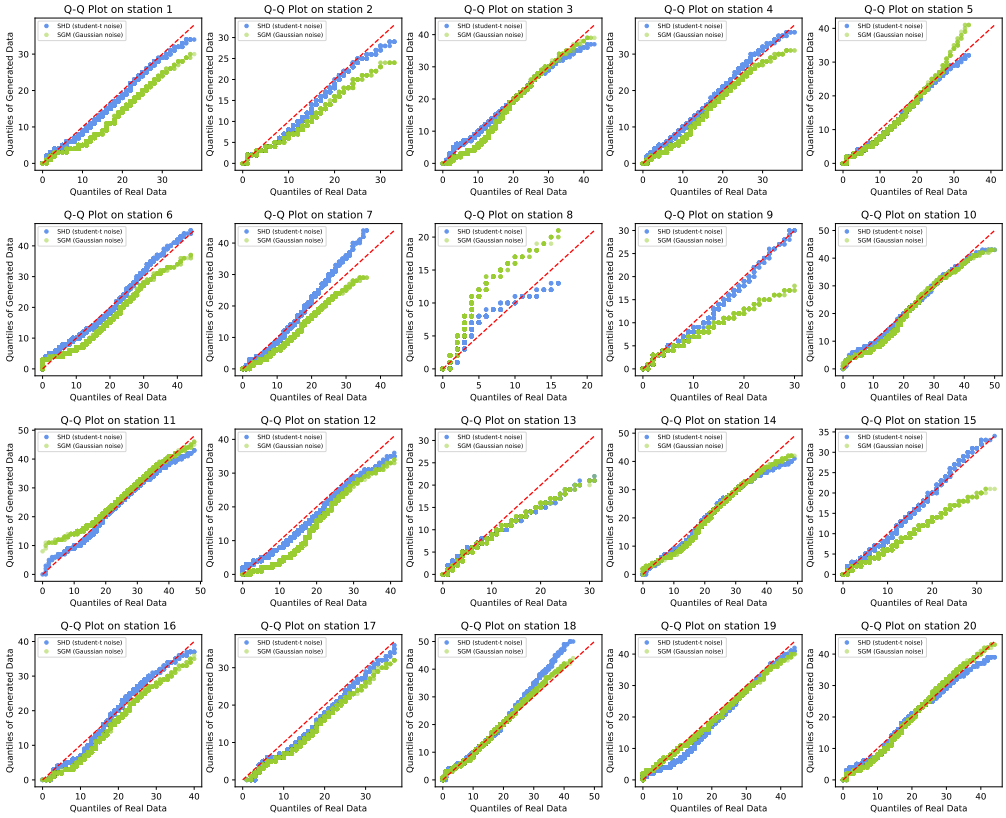


Figure EC.7 QQ-plot of between generated data and real data for all stations in the queueing system experiment. The green dots correspond to the standard SGM method, and the blue dots correspond to our SHD method. For each dataset, we take the smallest 99.5% of the data points for plotting, because the last 0.5% percent are empirically unstable and less informative.

E.4. Experiment on Bike-sharing Dataset

E.4.1. Details about the Dataset The entire dataset contains hourly rental count data of shared bicycles from January 1, 2011 to December 19, 2012. The dataset is provided by Kaggle[5] (Fanaee-T and Gama 2014). The data is split into daily observations. Each observation is a 24-dimensional vector $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \cdots, x_{24}^{(i)})$, where $x_j^{(i)}$ represents the number of rental counts in the j-th hour on the i-th day. The dataset also provides other features including weather, windspeed, etc, but we only focus on the number of rentals in each hour.

[5] Data source: `https://www.kaggle.com/competitions/bike-sharing-demand/data`

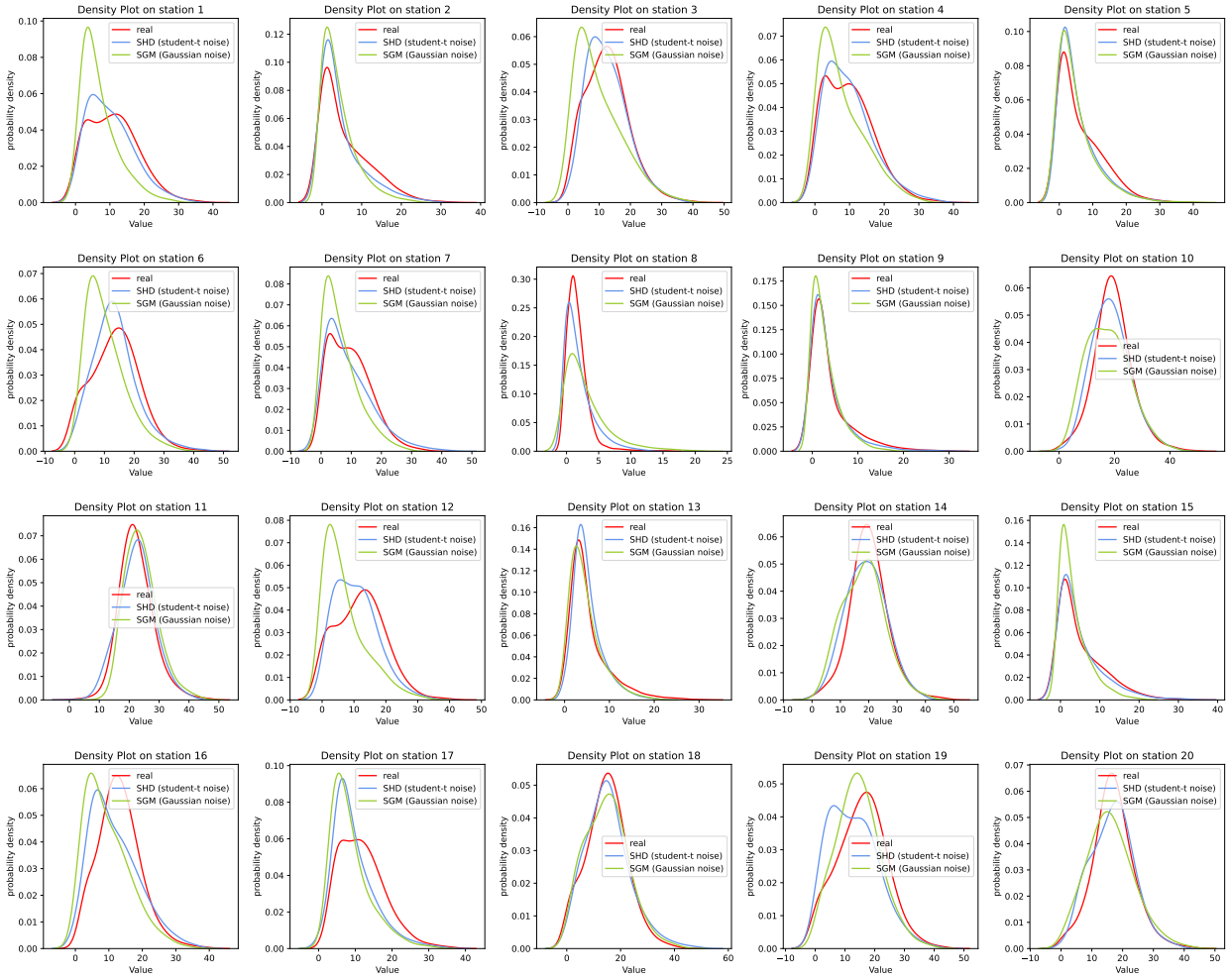


Figure EC.8 Density plot of between generated data and real data for all target dimensions in the queueing system experiment. For each dataset, we take the smallest 99% of the data points for plotting, because the density at the last 1% percent is too low to be clear on the plots.

We aim at learning the conditional distribution of hourly rentals from 12:00 to 24:00, with the hourly rentals from 0:00 to 12:00 being the condition. In other words, we train our model to generate samples of $(x_{13}, \cdots, x_{24})$ conditioning on $(x_1, \cdots, x_{12})$.

We remark that the pattern of hourly rentals on weekends are quite different from the pattern on weekdays. On weekdays, the hourly rental counts peak at around 8:00 and 17:00, which are the morning and evening rush hours. On weekends, the hourly rental counts has a single peak at around 13:00. However, we will train a unified model for all days and we do not specify whether a day is a weekday or weekend. We will let the model to extrapolate such pattern from the entire dataset by itself. The results in Figure 3 indicates that our model can successfully capture the different patterns based on the information about rentals from 0:00 to 12:00.

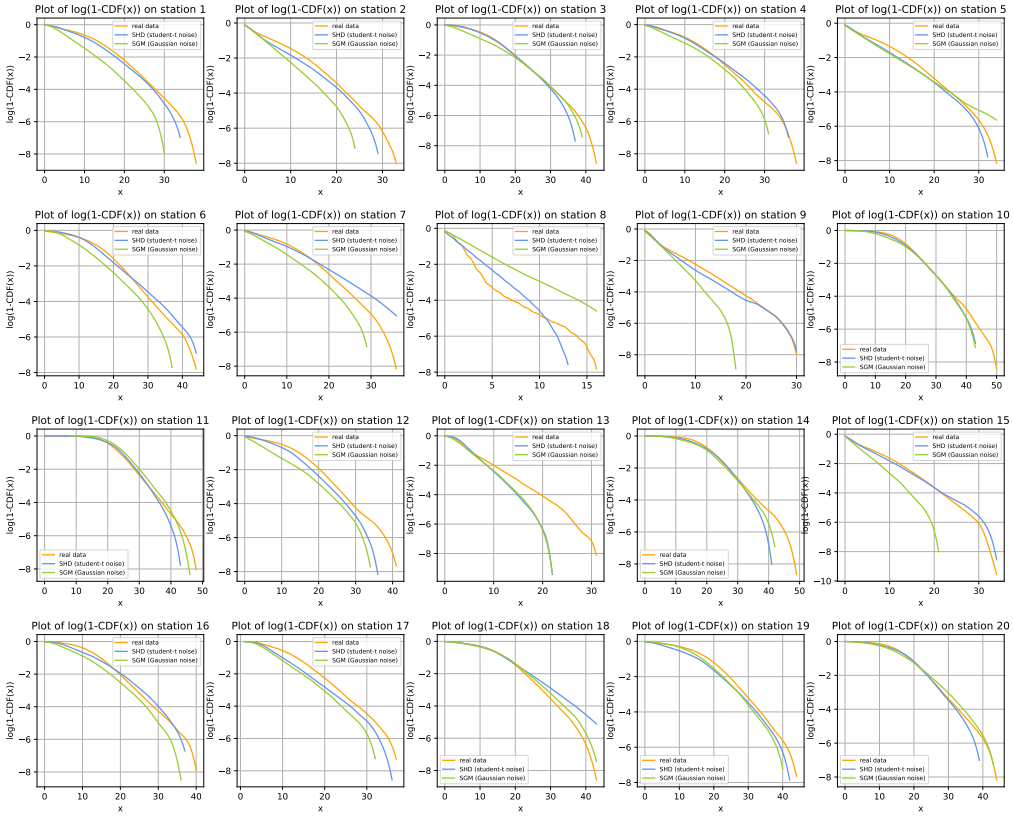


Figure EC.9 Plot of $\log(1-\mathrm{CDF}(x))$. The orange line corresponds to the real data; the green line corresponds to standard SGM method; and the blue line corresponds to our SHD method. $\mathrm{CDF}(x)$ is estimated using the kernel density method. For each dataset, we take the smallest 99.5% of the data points for plotting, because the last 0.5% percent are empirically unstable and less informative.

E.4.2. Training and Sampling Details The Hill estimator on the training set gives an average tail order of 7.91. We perform a grid search over $\nu \in [2.5, 9]$ and choose the best-performing value. We set $\nu = 4.5$ for the student-t noises added to the real samples, and the number of noise levels $T = 20$, with $\sigma_1 = 0.01$ and $\sigma_T = 0.5$. We use the randomly picked 400 daily observations as the training set, and the rest of the data as test set. The batch size is set to 128 and number of epochs is set as 200. In each epoch, we iterate along the whole training set, meaning that we have $\lceil \frac{400}{128} \rceil = 4$ iterations in each epoch. We use Adam optimizer with learning rate starting from 1×10^{-2} and exponentially decay to 1×10^{-5}. The hyperparameters of the model architecture are set the same as the correlated Pareto experiment, with details mentioned in Appendix E.1.3.

E.4.3. Evaluation: CRPS Now we describe the definition of continuous ranked probability score CRPS Matheson and Winkler (1976), which measures the compatibility of an estimated probability distribution $F(X)$ with an observation x. For a 1-dimensional variable, CRPS is defined as

$$\mathrm{CRPS}(F^{-1}, x) = \int_0^1 2(x - F^{-1}(p))(\alpha - \mathbb{I}_{x<F^{-1}(p)})d\alpha. \tag{EC.32}$$

We approximate the integral by discretizing with 0.05 ticks, which leads to

$$\mathrm{CRPS}(F^{-1}, x) \approx \sum_{i=1}^{19} 2(x - F^{-1}(0.05i))(0.05i - \mathbb{I}_{x<F^{-1}(0.05i)}). \tag{EC.33}$$

Then, following Tashiro et al. (2021), we evaluate the normalized average of CRPS for all target dimensions (in our bike-sharing experiment, the target dimensions are $x_{13}, \cdots, x_{24}$):

$$\frac{\sum_d \mathrm{CRPS}(F_d^{-1}, x_d)}{\sum_d |x_d|}, \tag{EC.34}$$

where F_d is the marginal distribution of x_d.

E.4.4. Additional Results Here we present the full version of Figure 3 with all dates in the evaluation set included. For each date, we construct the 50% and 90% confidence intervals for the number of rentals in each hour from the generated samples. Then we plot them against the real sample. The full plot is shown in Figure EC.10.

E.5. Experiment on Stock Dataset

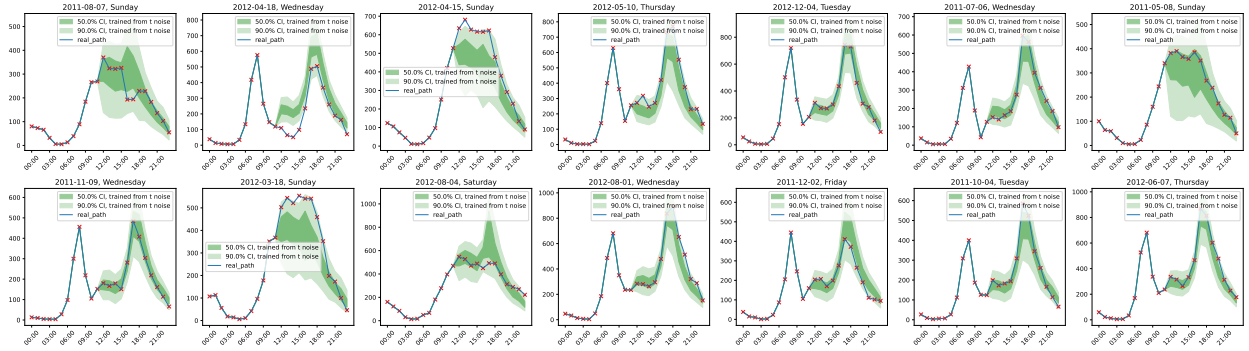
E.5.1. Details about the Dataset The entire dataset contains daily return data of 20 stocks from July 1, 2015 to July 30, 2021. The daily price data is obtained from Yahoo Finance.[6] The daily return of each stock is calculated using its close price. The entire dataset contains 1482 observations of daily returns. Each observation is a 20-dimensional vector $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \cdots, x_{20}^{(i)})$, where $x_j^{(i)}$ represents the return of the j-th stock on the i-th day. The full list of the stock codes is

sz.000069	sz.000725	sh.601607	sz.002271	sh.600061
sh.601985	sz.002142	sz.300122	sh.601166	sz.000596
sh.601766	sh.600309	sz.002466	sh.600029	sh.600999
sz.300433	sz.002202	sh.600104	sz.002601	sh.600958,

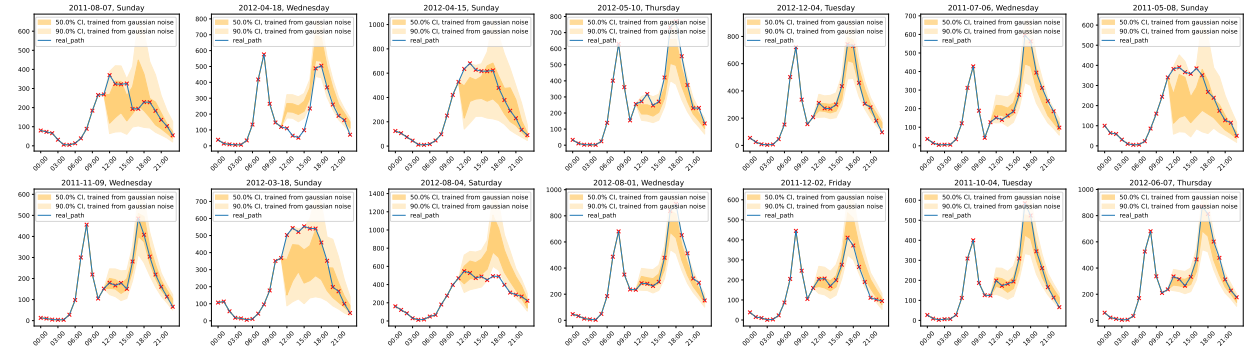
where "sh" represents Shanghai Stock Exchange and "sz" represents Shenzhen Stock Exchange.

We aim at learning the joint distribution of daily returns across different stocks. In other words, we train our unconditional model to generate new samples of $(x_1, \cdots, x_{20})$.

[6] Data source: Yahoo Finance (`https://finance.yahoo.com`)



(a) student-t noise



(b) Gaussian noise

Figure EC.10 Ground truth and generated confidence intervals on *bike rental dataset* for all dates in the evaluation set. The first 12 hours are conditions and the last 12 hours are targets.

E.5.2. Training and Sampling Details The Hill estimator on the training set gives an average tail order of 3.19. We perform a grid search over $\nu \in [2.5, 5]$ and choose the best-performing value. We set $\nu = 3.5$ for the student-t noises added to the real samples, and the number of noise levels $T = 20$, with $\sigma_1 = 0.01$ and $\sigma_T = 0.8$. We use the randomly picked 1200 daily observations as the training set, and the rest of the data as test set. The number of epochs is set as 300, and the batch size is set at 1200. In other words, we use the entire training set in each epoch. We use Adam optimizer with learning rate starting from 1×10^{-2} and exponentially decay to 1×10^{-5}. The hyperparameters of the model architecture are set the same as the correlated Pareto experiment, with details mentioned in Appendix E.1.3.

E.5.3. Evaluation Details We unconditionally generate 1000 samples of stock return and evaluate the performance of different methods by comparing with the 282 held-out test samples. First, we evaluate the performance in capturing the marginal distribution of the daily return for each stock by calculating the Wasserstein distance between the empirical distribution of the real evaluation set and the empirical distribution of generated samples, and then averaging across all 20 stocks. Second, to further demonstrate the ability

to capture cross-stock correlations, we create an equally-weighted portfolio using the 20 stocks. We calculate the daily return of the portfolio from the individual stock returns (for both real and generated samples), and then compute the Wasserstein distance between the real and generated portfolio return distributions.

References

Borjesson P, Sundberg CE (1979) Simple approximations of the error function q (x) for communications applications. *IEEE Transactions on Communications* 27(3):639–643.

Fanaee-T H, Gama J (2014) Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence* 2(2-3):113–127.

Lewis PW, Shedler GS (1979) Simulation of nonhomogeneous poisson processes by thinning. *Naval research logistics quarterly* 26(3):403–413.

Matheson JE, Winkler RL (1976) Scoring rules for continuous probability distributions. *Management science* 22(10):1087–1096.

Nair J, Wierman A, Zwart B (2013) The fundamentals of heavy-tails: Properties, emergence, and identification. *Proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*, 387–388.

Roberts GO, Tweedie RL (1996) Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli* 341–363.

Song Y, Ermon S (2020) Improved techniques for training score-based generative models. *Advances in neural information processing systems*, volume 33, 12438–12448.

Tashiro Y, Song J, Song Y, Ermon S (2021) Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems* 34:24804–24816.

Algorithm 2: Simulating $\mathbf{X}_{\tau_1}$ from τ_0

Input: Start time τ_0, sorted full arrival lists L^d_{arrival} with $d = 1, \cdots, D$, station transition matrix $\mathbf{P}$

Output: $line\_length\_list$: A list of line lengths at specified observation times

Initialization

  Initialize $\tau \leftarrow 0$; // current time

  Initialize $i \leftarrow 0$; // index for events

  Initialize L^d_{depart} as an empty list for $d = 1, \cdots, D$;

  Initialize $\tau^{d,k}_{\text{next}} \leftarrow 0$ for all d and k ; // index d,k means the kth server in station d

  Initialize L_{events} by concatenating all L^d_{arrival} and then sort by the arrival times;

Event Processing Loop

  while $\tau < T$ ***and*** $i < \text{len}(L_{events})$ **do**

    $\tau \leftarrow L_{\text{events}}[i][0]$; // Move τ forward to the next arrival time

    $d \leftarrow L_{\text{events}}[i][1]$; // Arrival is at station d

    $i \leftarrow i + 1$;

    $k \leftarrow \arg\min_k \tau^{d,k}_{\text{next}}$, $\tau_{\text{next}} = \min_k \tau^{d,k}_{\text{next}}$;

    $\tau_{\text{start}} \leftarrow \max\{\tau, \tau_{\text{next}}\}$;

    Draw $\tau_{\text{service}} \sim \text{Lognormal}(0, \sigma_d^2)$;

    $\tau_{\text{end}} \leftarrow \tau_{\text{start}} + \tau_{\text{service}}$;

    /* For those arrived before τ_0, the service time must end after τ_0 */

    while $\tau_{end} < \tau_0$ **do**

      Draw $\tau_{\text{service}} \sim \text{Lognormal}(0, \sigma_d^2)$;

      $\tau_{\text{end}} \leftarrow \tau_{\text{start}} + \tau_{\text{service}}$;

    $\tau^{d,k}_{\text{next}} \leftarrow \tau_{\text{end}}$;

    Insert (τ_{end}, d) into L^d_{depart};

    Draw d' according to $\mathbf{P}$; // $d' = 0$ if leaving the system

    if $d' > 0$ **then**

      Insert (τ_{end}, d') into L_{events};

Calculate number of customers at τ_1

  for $d = 1$ **to** D **do**

    $n_{\text{arrival}} =$ number of τ in L^d_{arrival} with $\tau \le \tau_1$;

    $n_{\text{depart}} =$ number of τ in L^d_{depart} with $\tau > \tau_1$;

    $X_{\tau_1, d} = n_{\text{arrival}} - n_{\text{depart}}$;

  return $\mathbf{X}_{\tau_1}$;