Diffusion Factor Models: Generating High-Dimensional Returns with Factor Structure

Minshuo Chen*, Renyuan Xu†, Yumin Xu‡, and Ruixun Zhang§

First version: April 2025
This version: January 2026

Abstract

Financial scenario simulation is essential for risk management and portfolio optimization, yet it remains challenging especially in high-dimensional and small data settings common in finance. We propose a *diffusion factor model* that integrates latent factor structure into generative diffusion processes, bridging econometrics with modern generative AI to address the challenges of the curse of dimensionality and data scarcity in financial simulation. By exploiting the low-dimensional factor structure inherent in asset returns, we decompose the score function—a key component in diffusion models—using time-varying orthogonal projections, and this decomposition is incorporated into the design of neural network architectures. We derive rigorous statistical guarantees, establishing non-asymptotic error bounds for both score estimation at $\widetilde{\mathcal{O}}\left(d^{5/2}n^{-\frac{2}{k+5}}\right)$ and generated distribution at $\widetilde{\mathcal{O}}\left(d^{5/4}n^{-\frac{1}{2(k+5)}}\right)$, primarily driven by the intrinsic factor dimension k rather than the number of assets d, surpassing the dimension-dependent limits in the classical nonparametric statistics literature and making the framework viable for markets with thousands of assets. Numerical studies confirm superior performance in mean and covariance estimation as well as latent subspace recovery under small data regimes. Empirical analysis demonstrates the economic significance of our framework in constructing mean-variance optimal portfolios and factor portfolios. This work presents the first theoretical integration of factor structure with diffusion models, offering a principled approach for high-dimensional financial simulation with limited data. Our code is available at https://github.com/xymmmm00/diffusion_factor_model.

Keywords: Generative Modeling; Diffusion Model; Asset Return Generation; Factor Model; Error Bound; Portfolio Construction;

*Department of Industrial Engineering and Management Sciences, Northwestern University. minshuo.chen@northwestern.edu (email).

†Department of Management Science & Engineering, Stanford University. renyuanxu@stanford.edu (email).

‡School of Mathematical Sciences, Peking University. xuyumin@pku.edu.cn (email).

§School of Mathematical Sciences, Center for Statistical Science, Laboratory for Mathematical Economics and Quantitative Finance, and National Engineering Laboratory for Big Data Analysis and Applications, Peking University. zhangruixun@pku.edu.cn (email).

Contents

1 Introduction

Financial scenario simulation, central to quantitative finance and risk management, has evolved significantly over recent decades (Alexander 2005, Eckerli and Osterrieder 2021, Brophy et al. 2023). Generating realistic and diverse financial scenarios is crucial not only for institutional traders to better manage their strategy risks, but also for regulators to ensure market stability (Acharya et al. 2023, Schneider, Strahan, and Yang 2023, Shapiro and Zeng 2024). The US Federal Reserve evaluates market conditions and releases a series of market stress scenarios on an annual basis (Federal Reserve Board 2023). Financial institutions are required to apply these scenarios to their portfolios to estimate and mitigate potential losses during market downturns. With the rise of trading automation and technological advancements, there is a pressing need from both parties to simulate more complex and high-dimensional financial scenarios (Reppen and Soner 2023). This request challenges traditional model-based simulation approaches (Behn, Haselmann, and Vig 2022, Hambly, Xu, and Yang 2023), highlighting the need for sophisticated data-driven techniques.

With the advances in machine learning techniques and computational power, generative AI has become a transformative force and is increasingly popular in financial applications. Its capabilities are now being harnessed for a wide range of tasks, such as generating financial time series (Yoon, Jarrett, and Van der Schaar 2019, Cont et al. 2022, Brophy et al. 2023, Acciaio, Eckstein, and Hou 2024), modeling volatility surfaces (Vuletić and Cont 2025), simulating limit order book dynamics (Coletta et al. 2023, Cont et al. 2023, Hultin et al. 2023), forecasting and imputing missing values (Tashiro et al. 2021, Vuletić, Prenzel, and Cucuringu 2024), and constructing portfolio strategies (Cetingoz and Lehalle 2025).

In recent years, several families of generative models have been explored in this context, including generative adversarial networks (GANs), autoencoders, and variational autoencoders (VAEs). In financial applications, GANs have been the primary workhorse (Yoon, Jarrett, and Van der Schaar 2019, Cont et al. 2022, Liao et al. 2024, Cetingoz and Lehalle 2025, Vuletić and Cont 2025), but are hindered by several issues, including training instability, mode collapse, high computational costs, and evaluation difficulties (Saatci and Wilson 2017, Borji 2019). In addition, developing a theoretical understanding of GANs is challenging due to their minimax structure and complex training process, which has hindered principled analysis and sustainable improvements since their inception (Creswell et al. 2018, Gui et al. 2021). Non-adversarial latent-variable models such as autoencoders and VAEs often face non-identifiable problems (Locatello et al. 2019), which can yield unstable performance on complex data (Saatci and Wilson 2017, Borji 2019, He et al. 2019, Dai, Wang, and Wipf 2020), risk posterior collapse, and residual-risk underestimation (Hoffman and Johnson 2016, He et al. 2019, Dai, Wang, and Wipf 2020).

More recently, generative diffusion models have gained traction as a more robust alternative, offering significant advantages in financial applications (Xiao, Kreis, and Vahdat 2022, Kotelnikov et al. 2023, Coletta et al. 2024, Li, Dai, and Qu 2024, Barancikova, Huang, and Salvi 2025). Compared to autoencoders, VAEs, and GANs, diffusion models can capture complex data distributions with more robust and stable performance, ease of training, and enhanced stability and efficiency

(Nichol and Dhariwal 2021, Dhariwal and Nichol 2021, Karras et al. 2022), achieving state-of-the-art results in practice and making them invaluable tools in advancing generative AI for finance. In particular, it is well documented that the diffusion model works well and beats alternatives in the small-data regime (Kotelnikov et al. 2023, Li, Dai, and Qu 2024). In addition, the diffusion model framework is grounded in rigorous stochastic and statistical analysis (Chen et al. 2024, Gao, Zha, and Zhou 2024, Tang and Zhao 2025), providing a theoretically sound basis for incorporating domain-specific properties, such as those in finance.

1.1 Our Work and Contributions

We develop a deep generative model based on diffusion models to simulate high-dimensional asset returns that follow an *unknown* factor structure, which we term the *diffusion factor model*. The returns of the d-dimensional assets are explained by the linear combination of k *unknown* common factors ($k \ll d$) and an idiosyncratic noise that varies from asset to asset (see Equation (8)).[1] We develop the theory for our diffusion factor model and establish statistical guarantees of the error of diffusion-generated returns, which overcomes the curse of dimensionality in the number of assets. We also conduct numerical and empirical studies to demonstrate its practical relevance.

Our generative model is particularly relevant for the high-dimensional small-data setting, a classical challenge for medium- (e.g., daily) to low- (e.g., weekly or monthly) frequency return data in finance. In empirical applications, the number of assets d often ranges from hundreds to thousands, easily exceeding the number of available observations in a stationary period (Kan and Zhou 2007, Nagel 2013, Gu, Kelly, and Xiu 2020). While machine learning is commonly perceived as a "big data" tool, many core finance questions are hindered by the "small data" nature of economic time series. Our model offers a methodology to tackle this challenge: fitting the diffusion model on the limited data and then generating additional realistic data for downstream tasks.

As a result, our diffusion factor model can be used to simulate realistic financial data for potential applications in a wide range of important contexts, including asset pricing and factor analysis across stock (Fama and French 1993, 2015a), option (Büchner and Kelly 2022), bond (Kelly, Palhares, and Pruitt 2023, Elkamhi, Jo, and Nozawa 2024), and cryptocurrency markets (Liu, Tsyvinski, and Wu 2022), large-scale asset covariance estimation (Bickel and Levina 2008, Fan, Liao, and Liu 2016, Ledoit and Wolf 2022), robust portfolio construction (DeMiguel et al. 2009, Avramov and Zhou 2010, Fabozzi, Huang, and Zhou 2010, Tu and Zhou 2010, Jacquier and Polson 2011), and systematic and institutional risk management (Bisias et al. 2012, Cont et al. 2022, He, Kou, and Peng 2022).

Our contributions are multi-fold. First, our diffusion factor model presents the first theoretical integration of factor models with generative diffusion models. It fully exploits the structural property of factor models, using observations of asset returns with heterogeneous idiosyncratic noises, and without requiring prior knowledge of the exact factors. In particular, our framework addresses the curse of dimensionality issue in the "small data" regime by achieving a sample complexity that scales exponentially in the desired error, with an exponent that primarily depends on the intrinsic

factor dimension k rather than the ambient asset dimension d.

Second, the success of the diffusion factor model hinges on accurately estimating Stein's score function, which we achieve by decomposing the score function via a time-varying projection into a subspace component in a k-dimensional space and a linear component (Lemma 1). This decomposition informs our neural network design—integrating denoising scheme, an encoder-decoder structure, and skip connections—to efficiently approximate the score function (Theorem 1). We establish a statistical guarantee that the L^2 error between the neural score estimator and the ground truth is $\tilde{\mathcal{O}}(d^{\frac{5}{2}} n^{-\frac{2}{k+5}})$ (Theorem 2), demonstrating that the dependence on the sample size n is governed primarily by k rather than d, effectively mitigating the curse of dimensionality.[2]

Third, we establish statistical guarantees for the errors in the generated return distribution as well as the subspace recovery. The output return distribution of our diffusion factor model is close to the true distribution in total variation distance, achieving an error of $\tilde{\mathcal{O}}(d^{\frac{5}{4}} n^{-\frac{1-\delta(n)}{2(k+5)}})$, where $\delta(n)$ vanishes as n grows. By applying singular value decomposition (SVD), we can also achieve latent subspace recovery with an error of order $\tilde{\mathcal{O}}(d^{\frac{5}{4}} n^{-\frac{1-\delta(n)}{k+5}})$ (Theorem 3). These results are achieved by the design of our sampling algorithm (Algorithm 1) and a novel analysis based on matrix concentration inequalities and coupling argument of stochastic processes (Lemmas 2 and 3). Furthermore, our efficient sample complexities hold true under a mild Lipschitz assumption (Assumption 3), demonstrating the generality of our analysis.

Fourth, numerical studies with synthetic data confirm that our diffusion factor model is capable of simulating realistic return data. It delivers substantial improvements in mean and covariance estimation as well as in subspace recovery, especially in the "small data" regime where the number of samples is small relative to the number of assets. These improvements suggest that our diffusion factor model automatically adapts to the (unknown) underlying factor structure and captures patterns of the data distribution more effectively than direct empirical estimation from limited data. From a statistical perspective, our methodology acts as a form of data-dependent regularization, introducing a modest modeling bias while substantially reducing the estimation variance, thus improving the moment estimation.

Finally, empirical analysis of the U.S. stock market shows that data generated by our diffusion factor model improves both mean and covariance estimation, leading to superior mean-variance optimal portfolios and factor portfolios. Portfolios using diffusion-generated data consistently outperform traditional methods, including equal-weight and shrinkage approaches, with higher mean returns and Sharpe ratios. In addition, factors estimated from the generated data capture interpretable economic characteristics and the corresponding tangency portfolios exhibit higher Sharpe ratios, effectively capturing systematic risk. These results demonstrate that our diffusion factor model can serve as a practical and broadly applicable tool for learning return distributions and constructing robust portfolios from limited, heavy-tailed financial data.

1.2 Related Literature

Our work is broadly related to two strands of the literature on factor models and diffusion models.

Factor Models. There is a vast econometric literature on factor models. Classic factor-based asset pricing models primarily focus on risk premium estimation, time-varying factors, model validity, and factor structure interpretability. Recent methodological advances have pioneered techniques for analyzing large, high-dimensional datasets, incorporating semiparametric estimation, robust inference, machine learning techniques, and time-varying risk premiums (Ferson and Harvey 1991, Connor, Hagmann, and Linton 2012, Feng, Giglio, and Xiu 2020, Gu, Kelly, and Xiu 2020, Raponi, Robotti, and Zaffaroni 2020, Chen, Pelger, and Zhu 2024, Feng et al. 2024, Giglio, Xiu, and Zhang 2025). We refer interested readers to survey papers such as Fama and French (2004), Giglio, Kelly, and Xiu (2022), Kelly, Xiu et al. (2023), and Bagnara (2024).

While we assume the (target) data distribution follows a factor model structure, the implementation and analysis of the diffusion models *do not require observing* the factors. In fact, our goal is to uncover the latent low-dimensional factor space through the data generation process. This is extremely valuable for financial applications, particularly in identifying effective factors, which is often challenging using traditional methods, see, for example, Chen, Roll, and Ross (1986), Jegadeesh and Titman (1993), Jagannathan and Wang (1996), Lettau and Ludvigson (2001), Pástor and Stambaugh (2003), Yogo (2006), Adrian, Etula, and Muir (2014), Hou, Xue, and Zhang (2015), He, Kelly, and Manela (2017), Lettau and Pelger (2020a) and Lettau and Pelger (2020b).

Diffusion Models and Their Theoretical Underpinnings. Diffusion models have shown groundbreaking success and quickly become the state-of-the-art method in diverse domains (Yang et al. 2023, Cao et al. 2024, Guo et al. 2024, Liu et al. 2024).

Despite significant empirical advances, the development of theoretical foundations for diffusion models falls behind. Recently, intriguing statistical and sampling theories emerged for deciphering, improving, and harnessing the power of diffusion models. Specifically, sampling theory considers whether diffusion models can generate a distribution that closely mimics the data distribution, given that the diffusion model is well-trained (De Bortoli et al. 2021, De Bortoli 2022, Albergo, Boffi, and Vanden-Eijnden 2023, Chen et al. 2023b, Chen, Daras, and Dimakis 2023, Benton et al. 2024, Huang, Huang, and Lin 2025, Li et al. 2024a, Li, Di, and Gu 2025).

Complementary to sampling theory, statistical theory of diffusion models mainly concerns how well the score function can be learned given finitely many training samples (Yang and Wibisono 2022, Koehler, Heckett, and Risteski 2023, Chen et al. 2023a, Oko, Akiyama, and Suzuki 2023, Dou et al. 2024, Wibisono, Wu, and Yang 2024, Zhang et al. 2024). Later, end-to-end analyses in Chen et al. (2023a), Oko, Akiyama, and Suzuki (2023), Azangulov, Deligiannidis, and Rousseau (2024), Fu et al. (2024), Tang and Yang (2024), Zhang et al. (2024), Yakovlev and Puchkin (2025) present statistical complexities of diffusion models for estimating nonparametric data distributions. It is worth noting that Chen et al. (2023a), Oko, Akiyama, and Suzuki (2023), Azangulov, Deligiannidis, and Rousseau (2024), Tang and Yang (2024), Wang et al. (2024) prove the adaptivity of diffusion models to the intrinsic structures of data—they can circumvent the curse of ambient dimensionality when data are exactly concentrated on a low-dimensional space.

Two works most closely related to ours are Chen et al. (2023a) and Wang et al. (2024), both of which consider subspace-structured data. Chen et al. (2023a) assume that each data point $\mathbf{X}$ lies exactly on a low-dimensional subspace, i.e., $\mathbf{X} = \mathbf{AZ}$ for some unknown matrix $\mathbf{A} \in \mathbb{R}^{d\times k}$ and latent variable $\mathbf{Z} \in \mathbb{R}^k$. In contrast, our factor model (Equation (8)) relaxes this strict subspace assumption by allowing idiosyncratic noise in the asset returns. The neural network architecture and parts of our analysis are inspired by Chen et al. (2023a), but the presence of high-dimensional idiosyncratic noise introduces substantial technical challenges in our setting. We discuss these technical novelties in detail in Section 5.2. Wang et al. (2024) also consider noisy subspace data, but assume that the latent variable $\mathbf{Z}$ follows a Gaussian mixture distribution. By comparison, we only require that the distribution of the latent variable satisfies a general sub-Gaussian tail condition. During the preparation of this manuscript, Yakovlev and Puchkin (2025) generalize the study to noisy nonlinear low-dimensional data structures. They assume that the data follow a transformation on a latent variable, which is uniformly distributed in a hypercube. This is very different from our study on the factor model structure.

1.3 Notation

We denote vectors and matrices by bold letters. For a vector $\mathbf{v}$, we denote $\|\mathbf{v}\|_2$, $\|\mathbf{v}\|_\infty$ as its ℓ^2 and ℓ^∞ norm, respectively. For a matrix $\mathbf{M}$, we denote $\mathrm{tr}(\mathbf{M})$, $\|\mathbf{M}\|_{\mathrm{F}}$ and $\|\mathbf{M}\|_{\mathrm{op}}$ as its trace, Frobenius norm, and operator norm, respectively. When $\mathbf{M}$ is symmetric, we denote $\lambda_{\max}(\mathbf{M})$ and $\lambda_k(\mathbf{M})$ as the maximal and the k-th largest eigenvalues. We also denote a matrix-induced norm as $\|\mathbf{v}\|_{\mathbf{M}}^2 = \mathbf{v}^\top\mathbf{M}\mathbf{v}$. For two symmetric matrices, we associate a partial ordering $\mathbf{M} \succeq \mathbf{N}$ if $\mathbf{M} - \mathbf{N}$ is positive semi-definite. For a random vector $\mathbf{X}$ following distribution P, we denote $\|\mathbf{X}\|_{L^2(P)}^2 = \mathbb{E}[\|\mathbf{X}\|_2^2]$. We denote $\phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ as the Gaussian density function with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. Throughout the paper, we use uppercase letters (e.g., $\mathbf{X}$) for random variables and lowercase letters (e.g., $\mathbf{x}$) for their realizations.

2 Problem Set-up for Diffusion Factor Models

Given limited market data, our objective is to design and train a diffusion-based factor model capable of simulating realistic, high-dimensional asset returns. This section introduces the two core components of our approach: generative diffusion models and the underlying factor structure. Section 2.1 defines diffusion models and emphasizes the central role of score functions in their construction. Section 2.2 presents a framework for modeling high-dimensional asset returns with an unknown low-dimensional latent structure, typically captured by a factor model—an essential feature for enabling efficient and robust modeling in small-data environments.

2.1 Generative Diffusion Models

Diffusion models consist of two interconnected processes: a forward process progressively injects noise into data over time, and a time-reverse process that constructs new data by progressively

removing noise (Anderson 1982, Haussmann and Pardoux 1986, Song and Ermon 2019, Ho, Jain, and Abbeel 2020, Song et al. 2021). The forward process is employed during training, while *the time-reverse process is used for data generation.* In the following, we formulate both processes using stochastic differential equations (SDEs) and detail the training methodology for diffusion models.

Forward and Time-Reverse SDEs. For ease of theoretical analysis, we follow the convention in the literature (Ho, Jain, and Abbeel 2020, Song and Ermon 2020) and adopt Ornstein-Ulhenbeck (O-U) process for the forward process. In particular, we study a simple O-U process with a deterministic and nondecreasing weight function $\eta(t) > 0$ as

$$\mathrm{d}\mathbf{R}_t = -\frac{1}{2}\eta(t)\mathbf{R}_t\mathrm{d}t + \sqrt{\eta(t)}\mathrm{d}\mathbf{W}_t \quad \text{with} \quad \mathbf{R}_0 \sim P_{\text{data}} \text{ and } t \in [0, T], \tag{1}$$

where $(\mathbf{W}_t)_{t\geq 0}$ is a standard Wiener process, T is a terminal time and P_{data} is the data distribution, i.e., the distribution of high-dimensional asset returns. We also denote P_t as the marginal distribution of $\mathbf{R}_t$ with a corresponding density function p_t. Given an initial value $\mathbf{R}_0 = \mathbf{r}$, at time t, the conditional distribution of $\mathbf{R}_t \,|\, \mathbf{R}_0 = \mathbf{r}$ is Gaussian, i.e.,

$$\mathbf{R}_t \,|\, \mathbf{R}_0 = \mathbf{r} \;\sim \mathcal{N}(\alpha_t \mathbf{r}, h_t \mathbf{I}_d), \tag{2}$$

where $\alpha_t = \exp\left(-\int_0^t \frac{1}{2}\eta(s)\mathrm{d}s\right)$ is the shrinkage ratio and $h_t = 1 - \alpha_t^2$ is the variance of the added Gaussian noise. For simplicity, we take $\eta(t) = 1$ throughout the paper. Note that the terminal distribution P_T is close to $P_\infty = \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ when T is sufficiently large, since the marginal distribution of an O-U process converges exponentially fast to its stationary distribution (Bakry et al. 2014, Chen et al. 2023b).

To design a procedure to generate new samples, we reverse the forward process in time (Anderson 1982, Song et al. 2021). Under mild regularity conditions (Haussmann and Pardoux 1986), this yields a well-defined backward process that transforms (white) noise into data. We denote the time-reversed SDE (backward process) associated with (1) as

$$\mathrm{d}\mathbf{R}_t^{\leftarrow} = \left(\frac{1}{2}\mathbf{R}_t^{\leftarrow} + \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right)\mathrm{d}t + \mathrm{d}\overline{\mathbf{W}}_t \quad \text{with} \quad \mathbf{R}_0^{\leftarrow} \sim Q_0 \text{ and } t \in [0, T], \tag{3}$$

where $(\overline{\mathbf{W}}_t)_{t\geq 0}$ is another Wiener process independent of $(\mathbf{W}_t)_{t\geq 0}$, $\nabla \log p_t(\cdot)$ is known as the *score function* and Q_0 is the initial distribution of the backward process. If we set $Q_0 = P_T$, under mild assumptions, the time-reverse process has the *same marginal distribution* as the forward process in the sense of $\text{Law}(\mathbf{R}_t^{\leftarrow}) = \text{Law}(\mathbf{R}_{T-t})$; see Anderson (1982) and Haussmann and Pardoux (1986) for details. In particular, we have $\text{Law}(\mathbf{R}_T^{\leftarrow}) = P_{\text{data}}$, which leads us to recover the data distribution.

In practice, however, (3) cannot be directly used to generate samples from the data distribution P_{data} as both the score function and the distribution P_T are *unknown*. To train a simulator that generates data (closely) from P_{data}, the key is to accurately learn the score function. With a score estimator $\widehat{\mathbf{s}}$ that approximates $\nabla \log p_t$ and an initial distribution $Q_0 := \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ that is easy to

sample, we specify the following implementable process for data generation

$$\mathrm{d}\widehat{\mathbf{R}}_t^{\leftarrow} = \left(\frac{1}{2}\widehat{\mathbf{R}}_t^{\leftarrow} + \widehat{\mathbf{s}}\left(\widehat{\mathbf{R}}_t^{\leftarrow}, T-t\right)\right)\mathrm{d}t + \mathrm{d}\overline{\mathbf{W}}_t \quad \text{with} \quad \widehat{\mathbf{R}}_0^{\leftarrow} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d). \tag{4}$$

For O-U processes, the error introduced by taking $Q_0 = \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ usually decays exponentially with respect to T (Chen et al. 2023b, Lee, Lu, and Tan 2023, Tang and Zhao 2024, Gao, Nguyen, and Zhu 2025).

Training by Score Matching. To learn the score function $\nabla \log p_t$ in (3), a natural method is to minimize a mean-squared error between the estimated and true scores (Hyvärinen and Dayan 2005), i.e.,

$$\min_{\mathbf{s}\in\mathcal{S}} \int_{t_0}^{T} w(t)\mathbb{E}_{\mathbf{R}_t}\left[\|\mathbf{s}(\mathbf{R}_t, t) - \nabla \log p_t(\mathbf{R}_t)\|_2^2\right]\mathrm{d}t, \tag{5}$$

where $w(t)$ is a positive weighting function and $\mathbf{s}$ is a parameterized estimator of the score function from a class $\mathcal{S}$ such as neural networks. Here, $t_0 > 0$ is a small early-stopping time to prevent the score function from blowing up as $t \to 0$ (Song and Ermon 2019, Chen et al. 2023a).

A key challenge in minimizing the score-matching loss (5) is that the target term, $\nabla \log p_t$, is generally intractable—it cannot be computed directly from observed data. Alternatively, one can equivalently minimize the following denoising score matching proposed in Vincent (2011), Song et al. (2020), which utilizes the conditional density of $\mathbf{R}_t \,|\, \mathbf{R}_0$ in (2):

$$\min_{\mathbf{s}\in\mathcal{S}} \int_{t_0}^{T} w(t)\mathbb{E}_{\mathbf{R}_0}\left[\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0}\left[\|\mathbf{s}(\mathbf{R}_t, t) - \nabla \log \phi(\mathbf{R}_t; \alpha_t\mathbf{R}_0, h_t\mathbf{I}_d)\|_2^2\right]\right]\mathrm{d}t. \tag{6}$$

Here ϕ is the Gaussian density function defined at the end of Section 1. For technical convenience, we choose a uniform weight $w(t) = 1/(T - t_0)$. Note that under the forward dynamics (1), $\nabla \log \phi(\mathbf{r}_t; \alpha_t\mathbf{r}_0, h_t\mathbf{I}_d)$ in (6) has an analytical form,

$$\nabla \log \phi(\mathbf{r}_t; \alpha_t\mathbf{r}_0, h_t\mathbf{I}_d) = -\frac{\mathbf{r}_t - \alpha_t\mathbf{r}_0}{h_t}.$$

In practice, we can only observe a finite sample of asset returns $\{\mathbf{r}^i\}_{i=1}^n$ from P_{data}. Therefore, we train the diffusion model using the following empirical score-matching objective:

$$\min_{\mathbf{s}\in\mathcal{S}} \widehat{\mathcal{L}}(\mathbf{s}) := \frac{1}{n}\sum_{i=1}^{n} \ell(\mathbf{r}^i, \mathbf{s}) \quad \text{with} \quad \ell(\mathbf{r}^i, \mathbf{s}) = \frac{1}{T - t_0}\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}^i}\left\|\mathbf{s}(\mathbf{R}_t, t) + \frac{\mathbf{R}_t - \alpha_t\mathbf{r}^i}{h_t}\right\|_2^2 \mathrm{d}t. \tag{7}$$

Henceforth we write the population loss function in (6) as $\mathcal{L}(\mathbf{s}) := \mathbb{E}[\widehat{\mathcal{L}}(\mathbf{s})]$.

2.2 Asset Returns and Unknown Factor Structure

To improve sample efficiency, especially in small-data regimes, the central idea is to incorporate domain knowledge into the diffusion model. Specifically, we leverage a key insight from the finance

literature: a relatively small set of latent factors—reflecting both macroeconomic and firm-specific variables—can effectively explain a broad class of asset returns (Ross 2013, Fan, Liao, and Wang 2016, Aït-Sahalia and Xiu 2019, Giglio and Xiu 2021, Bryzgalova et al. 2023, Kelly, Malamud, and Pedersen 2023). Following these studies, we consider the asset return $\mathbf{R} \sim P_{\text{data}}$ satisfying the following factor model structure:

$$\mathbf{R} = \boldsymbol{\beta}\,\mathbf{F} + \boldsymbol{\varepsilon}, \tag{8}$$

where $\mathbf{F} \in \mathbb{R}^k$ are *unknown* factors with $k \ll d$, $\boldsymbol{\beta} \in \mathbb{R}^{d\times k}$ is a factor loading matrix, and $\boldsymbol{\varepsilon} \in \mathbb{R}^d$ is the vector of idiosyncratic residuals.

We want to emphasize that, while we assume the data distribution P_{data} follows a factor model structure (8), the implementation and analysis of the diffusion models *do not require observing* the factors. Instead, our approach is capable of uncovering the latent low-dimensional factor space through the data generation process; see Section 5 for more details.

Under the unknown factor scenario, factors and their loadings are identifiable only up to an invertible linear transformation, e.g., rescaling and rotation (Kelly, Xiu et al. 2023). Thus, it is reasonable to assume that $\boldsymbol{\beta}$ has orthonormal columns. Otherwise, one can perform a QR decomposition to write $\boldsymbol{\beta} = \boldsymbol{\beta}'\mathbf{H}$, where $\boldsymbol{\beta}' \in \mathbb{R}^{d\times k}$ has orthonormal columns and $\mathbf{H} \in \mathbb{R}^{k\times k}$ is an upper triangular matrix.

In light of the factor structure in (8), we aim to develop a diffusion model framework that explicitly exploits this low-dimensional representation. Crucially, the statistical guarantees of our approach depend primarily on the number of latent factors k, rather than the ambient data dimension d. This dimensionality reduction enables the diffusion model to be trained effectively on a limited number of observations, while still generating realistic high-dimensional samples. As a result, the proposed framework addresses two central challenges in modeling financial data: the curse of dimensionality and data scarcity.

3 Score Decomposition under Diffusion Factor Model

To simulate high-dimensional asset returns using diffusion factor models, the key challenge is accurately learning the score function via neural networks. However, due to the high dimensionality of asset returns and limited market data, directly estimating the score function is impractical as it suffers from the curse of dimensionality. To overcome this, we analyze the structural properties of score functions under factor models, deriving a tractable decomposition. This decomposition informs a neural network architecture designed to perform effectively in small-data regimes.

3.1 Score Decomposition

With factor model structure in (8), we show that the score function $\nabla \log p_t$ can be decomposed into a subspace score in a k-dimensional space and a complementary component, each possessing distinct properties.

To ensure the decomposition is well-defined, we impose the following assumption.

Assumption 1 (Factor model)**.** *We assume the following conditions on the factor model (8):*

(i) The factor loading $\boldsymbol{\beta} \in \mathbb{R}^{d\times k}$ has orthonormal columns.

(ii) The factor $\mathbf{F} \in \mathbb{R}^k$ follows a distribution that has a density function denoted as p_{fac} and has a finite second moment, i.e., $\int \|\mathbf{f}\|_2^2 \, p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f} < \infty$.

(iii) The residual $\boldsymbol{\varepsilon}$ is Gaussian with density $\phi(\cdot; \mathbf{0}, \operatorname{diag}\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_d^2\})$ and there exists a positive constant $\sigma_{\max} > 0$ such that

$$\sigma_{\max} \geq \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > 0.$$

(iv) $\mathbf{F}$ and $\boldsymbol{\varepsilon}$ are independent.

As a result, $\mathbf{R}$ has a positive definite covariance matrix, defined as

$$\boldsymbol{\Sigma}_0 := \mathbb{E}[\mathbf{R}\mathbf{R}^\top] - \mathbb{E}[\mathbf{R}]\mathbb{E}[\mathbf{R}]^\top. \tag{9}$$

Next, for an arbitrary time $t \in [0, T]$, we consider a linear subspace $\mathcal{V}_t$ spanned by the column vectors of $\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}$, with $\boldsymbol{\Lambda}_t$ defined as

$$\boldsymbol{\Lambda}_t := \operatorname{diag}\left\{h_t + \sigma_1^2\alpha_t^2, h_t + \sigma_2^2\alpha_t^2, \ldots, h_t + \sigma_d^2\alpha_t^2\right\}. \tag{10}$$

We further define $\mathbf{T}_t$ as the matrix of orthogonal projection onto $\mathcal{V}_t$:

$$\mathbf{T}_t := \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-\frac{1}{2}} \quad \text{with} \quad \boldsymbol{\Gamma}_t := (\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta})^{-1}. \tag{11}$$

Matrix $\boldsymbol{\Gamma}_t$ is well-defined as $\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}$ is positive definite due to Assumption 1. The following lemma presents the score decomposition.

Lemma 1. *Suppose Assumption 1 holds. The score function $\nabla \log p_t(\mathbf{r})$ can be decomposed into a subspace score and a complement score as*

$$\nabla \log p_t(\mathbf{r}) = \underbrace{\mathbf{T}_t\boldsymbol{\Lambda}_t^{\frac{1}{2}}\boldsymbol{\beta} \cdot \nabla \log p_t^{\text{fac}}(\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{\frac{1}{2}}\mathbf{T}_t \cdot \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r})}_{\textit{Subspace score}} \underbrace{-\boldsymbol{\Lambda}_t^{-\frac{1}{2}}(\mathbf{I} - \mathbf{T}_t) \cdot \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}}_{\textit{Complement score}}, \tag{12}$$

where $p_t^{\text{fac}}(\cdot) := \int \phi(\cdot; \alpha_t\mathbf{f}, \boldsymbol{\Gamma}_t)p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}$ and $\boldsymbol{\Lambda}_t$, $\boldsymbol{\Gamma}_t$, $\mathbf{T}_t$ are defined in (10) and (11).

For future convenience, we denote the subspace score as $\mathbf{s}_{\text{sub}} : \mathbb{R}^k \times [0, T] \to \mathbb{R}^d$ and the complement score as $\mathbf{s}_{\text{comp}} : \mathbb{R}^d \times [0, T] \to \mathbb{R}^d$:

$$\mathbf{s}_{\text{sub}}(\mathbf{z}, t) := \mathbf{T}_t\boldsymbol{\Lambda}_t^{\frac{1}{2}}\boldsymbol{\beta} \cdot \nabla \log p_t^{\text{fac}}(\mathbf{z}), \quad \text{and} \tag{13}$$

$$\mathbf{s}_{\text{comp}}(\mathbf{r}, t) := -\boldsymbol{\Lambda}_t^{-\frac{1}{2}}(\mathbf{I} - \mathbf{T}_t)\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}. \tag{14}$$

We defer the proof to Appendix A. A few discussions are in place.

Motivation and Insights of Score Decomposition. Lemma 1 is proved using an orthogonal decomposition of the rescaled noisy data, $\mathbf{\Lambda}_t^{-1/2}\mathbf{r} = \mathbf{T}_t \cdot \mathbf{\Lambda}_t^{-1/2}\mathbf{r} + (\mathbf{I} - \mathbf{T}_t) \cdot \mathbf{\Lambda}_t^{-1/2}\mathbf{r}$, with the two decomposed terms serving different roles. Specifically, $\mathbf{s}_{\text{sub}}$ is responsible for recovering the distribution of the low-dimensional factors, while $\mathbf{s}_{\text{comp}}$ progressively adjusts the covariance of the generated returns to match that of the heterogeneous noise.

Furthermore, Lemma 1 provides key insights into an efficient representation of the score function. As observed, the subspace score depends only on a k-dimensional input, while the complement score is linear, suggesting a natural dimension reduction in representing the score. Learning a low-dimensional nonlinear function together with a linear component is substantially more efficient—in terms of both sample complexity and computational cost—than learning a high-dimensional nonlinear function. Indeed, the score network architecture introduced in Section 3.2, along with the subsequent statistical analysis in Sections 4 and 5, reflects this critical observation.

3.2 Choosing Score Network Architecture

When training a diffusion model, we parameterize the score function using neural networks, where a properly chosen network architecture plays a vital role in effective training. The score decomposition in Lemma 1 suggests a well-informed network architecture design. Before we introduce our network architecture, we briefly summarize our notion of ReLU networks considered in this paper.

Let $\mathcal{S}_{\text{ReLU}}$ be a family of neural networks with ReLU activations determined by a set of hyper-parameters L, M, J, K, κ, γ_1, and γ_2. Roughly speaking, L is the depth of the network, M is the width of the network, J is the number of non-zero weight parameters, K is the range of network output, κ is the largest magnitude of weight parameters, and γ_1 as well as γ_2 are both Lipschitz coefficients as we detail below. Formally, considering that a score network takes noisy data $\mathbf{r}$ and time t as input, we define $\mathcal{S}_{\text{ReLU}}$ as

$$
\begin{aligned}
&\mathcal{S}_{\text{ReLU}}(L, M, J, K, \kappa, \gamma_1, \gamma_2) \\
&= \Big\{ \mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{r}, t) = \mathbf{W}_L \cdot \text{ReLU}(\cdots \text{ReLU}(\mathbf{W}_1[\mathbf{z}^\top, t]^\top + \mathbf{b}_1) \cdots) + \mathbf{b}_L \text{ with } \boldsymbol{\zeta} := \{\mathbf{W}_\ell, \mathbf{b}_\ell\}_{\ell=1}^L : \\
&\qquad \text{network width bounded by } M,\ \sup_{\mathbf{r},t} \|\mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{r}, t)\|_2 \le K, \\
&\qquad \max\{\|\mathbf{b}_\ell\|_\infty, \|\mathbf{W}_\ell\|_\infty\} \le \kappa \text{ for } \ell = 1, \ldots, L,\ \sum_{\ell=1}^{L} (\|\mathbf{b}_\ell\|_0 + \|\mathbf{W}_\ell\|_0) \le J, \\
&\qquad \|\mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{r}_1, t) - \mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{r}_2, t)\|_2 \le \gamma_1 \|\mathbf{r}_1 - \mathbf{r}_2\|_2 \text{ for any } t \in (0, T], \\
&\qquad \|\mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{r}, t_1) - \mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{r}, t_2)\|_2 \le \gamma_2 |t_1 - t_2| \text{ for any } \mathbf{r} \Big\},
\end{aligned}
\tag{15}
$$

where ReLU activation is applied entrywise, and each weight matrix $\mathbf{W}_\ell$ is of dimension $d_\ell \times d_{\ell+1}$. Correspondingly, the width of the network is denoted by $M = \max_\ell d_\ell$. Here, the uniform bound $\sup_{\mathbf{r},t} \|\mathbf{g}_\zeta(\mathbf{r}, t)\|_2 \le K$ and the sparsity constraint $\sum_{\ell=1}^{L} (\|\mathbf{b}_\ell\|_0 + \|\mathbf{W}_\ell\|_0) \le J$ are standard and practically assumptions for ReLU networks (Bartlett, Foster, and Telgarsky 2017, Louizos, Welling,

and Kingma 2018, Hoefler et al. 2021, Song et al. 2021).[3] The Lipschitz continuity on $\mathbf{g}_{\boldsymbol{\zeta}}$ is often enforced by Lipschitz network training (Gouk et al. 2021) or induced by implicit bias of the training algorithm (Soudry et al. 2018, Bartlett et al. 2020).

Now, using $\mathcal{S}_{\mathrm{ReLU}}$, we design our score network architecture by first rearranging terms in (12) as

$$\begin{aligned}\nabla \log p_t(\mathbf{r}) &= \boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\frac{\int \alpha_t \mathbf{f}\cdot\phi(\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}}{\int \phi(\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}} - \boldsymbol{\Lambda}_t^{-1}\mathbf{r} \\ &= \alpha_t\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\cdot\boldsymbol{\xi}(\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r},t) - \boldsymbol{\Lambda}_t^{-1}\mathbf{r}, \end{aligned} \tag{16}$$

where $\boldsymbol{\xi}:\mathbb{R}^k\times[0,T]\to\mathbb{R}^k$ is defined as

$$\boldsymbol{\xi}(\mathbf{z},t) := \frac{\int \mathbf{f}\cdot\phi(\boldsymbol{\Gamma}_t\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}}{\int \phi(\boldsymbol{\Gamma}_t\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}} \quad \text{for} \quad \mathbf{z}\in\mathbb{R}^k. \tag{17}$$

The i-th element of $\boldsymbol{\xi}(\mathbf{z},t)$ is denoted as $\xi_i(\mathbf{z},t)$. Note that the coefficient α_t forces the first term to decay exponentially. Therefore, for sufficiently large t, the score function $\nabla\log p_t(\mathbf{r})$ is approximately a linear function, corresponding to the second term in (16).

When choosing the score network architecture, we aim to reproduce the functional form in (16). Accordingly, we define a class of neural networks built upon $\mathcal{S}_{\mathrm{ReLU}}$ as

$$\begin{aligned}&\mathcal{S}_{\mathrm{NN}}(L,M,J,K,\kappa,\gamma_1,\gamma_2,\sigma_{\max}) \\ =&\Big\{\mathbf{s}_\theta(\mathbf{r},t) = \alpha_t\mathbf{D}_t\mathbf{V}\cdot\mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{V}^\top\mathbf{D}_t\mathbf{r},t) - \mathbf{D}_t\mathbf{r} \text{ with } \boldsymbol{\theta} := \{\mathbf{c},\mathbf{V},\boldsymbol{\zeta}\}: \\ &\qquad \mathbf{c} := [c_1,c_2,\ldots,c_d]^\top \in [0,\sigma_{\max}]^d, \quad \mathbf{V}\in\mathbb{R}^{d\times k} \text{with orthogonal columns}, \\ &\qquad \mathbf{D}_t := \mathrm{diag}\left\{1/(h_t+\alpha_t^2c_1),\ldots,1/(h_t+\alpha_t^2c_d)\right\} \text{induced by } \mathbf{c}, \\ &\qquad \mathbf{g}_{\boldsymbol{\zeta}}\in\mathcal{S}_{\mathrm{ReLU}}(L,M,J,K,\kappa,\gamma_1,\gamma_2)\Big\}. \end{aligned} \tag{18}$$

In (18), $\mathbf{V}$ represents the unknown factor loading $\boldsymbol{\beta}$ and $\mathbf{D}_t$ represents $\boldsymbol{\Lambda}_t^{-1}$. The ReLU network $\mathbf{g}_{\boldsymbol{\zeta}}$ is responsible for implementing $\boldsymbol{\xi}$. We remark that $\mathbf{V}^\top$ and $\mathbf{V}$ serve as the linear encoder and decoder, respectively, and $-\mathbf{D}_t\mathbf{r}$ is incorporated as a shortcut connection within the U-Net framework (Ronneberger, Fischer, and Brox 2015). When there is no confusion, we drop the hyper-parameters and denote the network classes in (15) and (18) as $\mathcal{S}_{\mathrm{ReLU}}$ and $\mathcal{S}_{\mathrm{NN}}$, respectively.

4 Score Approximation and Estimation

Given the score decomposition and score network architecture $\mathcal{S}_{\mathrm{NN}}$, this section establishes two intriguing properties: 1) with appropriate hyper-parameters, $\mathcal{S}_{\mathrm{NN}}$ can well approximate any score function in the form (12), and 2) learning the score function within $\mathcal{S}_{\mathrm{NN}}$ leads to an efficient sample complexity. Specifically, we establish an approximation theory to the score function in Section 4.1. Building on the approximation guarantee, Section 4.2 derives bounds on the statistical

error, providing finite-sample guarantees for score estimation, where the sample complexity bounds depend primarily on the number of factors k rather than ambient dimension d.

4.1 Theory of Score Approximation

The following assumptions on the factor distribution and score function are needed to establish our score approximation guarantee.

Assumption 2 (Factor distribution)**.** *The density function for the factors, $p_{\text{fac}}(\cdot)$, is non-negative and twice continuously differentiable. In addition $p_{\text{fac}}(\cdot)$ has sub-Gaussian tail, namely, there exist constants B, C_1, and C_2 such that*

$$p_{\text{fac}}(\mathbf{f}) \leq (2\pi)^{-\frac{k}{2}} C_1 \exp(-C_2\|\mathbf{f}\|_2^2/2) \text{ when } \|\mathbf{f}\|_2 \geq B. \tag{19}$$

Assumption 2 is commonly adopted both in the literature of high-dimensional statistics (Vershynin 2018, Wainwright 2019) and in recent work on diffusion-model theory (De Bortoli et al. 2021, Chen et al. 2023a, Oko, Akiyama, and Suzuki 2023, Cole and Lu 2024). In finance, Assumption 2 is standard in the factor/econometrics literature (e.g., Bai and Ng 2002, 2023) for modeling low-frequency returns, which are well known to exhibit the aggregated Gaussianity property (Fan and Yao 2003). We also need the following regularity assumption on the score function.

Assumption 3. *The subspace score function $\mathbf{s}_{sub}(\mathbf{z}, t)$ is L_s-Lipschitz in $\mathbf{z}$ for any $t \in [0, T]$.*

The Lipschitz assumption on the score function is a standard assumption in the diffusion model literature (Lee, Lu, and Tan 2022, Chen et al. 2023b, Han, Razaviyayn, and Xu 2024). Note that Assumption 3 only requires the Lipschitz continuity for the subspace score. But it implies that $\nabla \log p_t$ is Lipschitz with coefficient $\left(L_s \cdot \frac{h_t+\sigma_1^2\alpha_t^2}{h_t+\sigma_d^2\alpha_t^2} + \frac{1}{h_t+\sigma_d^2\alpha_t^2}\right)$, which is in a similar spirit to the condition proposed in Lee, Lu, and Tan (2022). As a concrete example, a Gaussian distribution with a nondegenerate covariance satisfies Assumption 3.

Example 1 (Gaussian factors)**.** *Assume the factor $\mathbf{F}$ follows a nondegenerate Gaussian distribution, i.e.,*

$$\mathbf{F} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \quad \textit{with} \quad \mathbf{\Sigma} = \text{diag}\{\varsigma_1, \ldots, \varsigma_k\} \succ \mathbf{0}. \tag{20}$$

Then, an explicit calculation gives rise to

$$\nabla \log p_t(\mathbf{r}) = (-\mathbf{\Lambda}_t^{-1}\boldsymbol{\beta}\mathbf{\Gamma}_t(\mathbf{\Gamma}_t + \alpha_t^2\mathbf{\Sigma})^{-1})\boldsymbol{\beta}^\top \mathbf{\Lambda}_t^{\frac{1}{2}}\mathbf{T}_t \cdot \mathbf{\Lambda}_t^{-\frac{1}{2}}\mathbf{r} - \mathbf{\Lambda}_t^{-\frac{1}{2}}(\mathbf{I} - \mathbf{T}_t) \cdot \mathbf{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}.$$

Correspondingly, the subspace score $\mathbf{s}_{sub}$ is written as

$$\mathbf{s}_{sub}(\mathbf{z}, t) = (-\mathbf{\Lambda}_t^{-1}\boldsymbol{\beta}\mathbf{\Gamma}_t(\mathbf{\Gamma}_t + \alpha_t^2\mathbf{\Sigma})^{-1})\mathbf{z},$$

which is Lipschitz in $\mathbf{z}$.

We state our theory of score approximation as follows.

Theorem 1. *Suppose Assumptions 1-3 hold. Given an approximation error $\epsilon > 0$, there exists a network $\bar{\mathbf{s}}_{\boldsymbol{\theta}} \in \mathcal{S}_{\mathrm{NN}}$ such that for any $t \in [0, T]$, it presents an upper bound*

$$\mathbb{E}_{\mathbf{R}_t \sim P_t} \|\bar{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t, t) - \nabla \log p_t(\mathbf{R}_t)\|_2 \leq \frac{(\sqrt{k}+1)\epsilon}{\min\{\sigma_d^2, 1\}}. \tag{21}$$

The configuration of the network architecture $\mathcal{S}_{\mathrm{NN}}$ satisfies

$$\begin{aligned}
&M = \mathcal{O}\bigg(T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\big(\log\frac{1}{\epsilon}+k\big)^{\frac{k}{2}}\bigg),\ \gamma_1 = 20k(1+L_s)(1+\sigma_{\max}^4),\\
&L = \mathcal{O}\bigg(\log\frac{1}{\epsilon}+k\bigg),\ J = \mathcal{O}\bigg(T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\big(\log\frac{1}{\epsilon}+k\big)^{\frac{k+2}{2}}\bigg),\ \gamma_2 = 10\tau,\\
&K = \mathcal{O}\bigg((1+L_s)(1+\sigma_{\max}^4)\left(\log\frac{1}{\epsilon}+k\right)^{\frac{1}{2}}\bigg),\ \kappa = \max\big\{(1+L_s)(1+\sigma_{\max}^4)\big(\log\frac{1}{\epsilon}+k\big)^{\frac{1}{2}}, T\tau\big\},
\end{aligned} \tag{22}$$

where

$$\tau = \sup_{t\in[0,T],\|\mathbf{z}\|_\infty \leq \sqrt{(1+\sigma_{\max}^2)(k+(\log 1/\epsilon))}} \left\|\frac{\partial}{\partial t}\boldsymbol{\xi}(\mathbf{z},t)\right\|_2 \quad \textit{with } \boldsymbol{\xi} \textit{ defined in (17).}$$

As shown in (21), the approximation error has a benign dependence on the dimension. It primarily depends on $\min\{\sigma_d^2, 1\}$ and k, rather than d. The proof is deferred to Appendix B.1. Below, we provide key insights offered by Theorem 1, along with a proof sketch and a discussion of the main technical challenges.

Discussion on Network Architecture. In contrast to conventional neural network designs for universal approximation, such as those in Yarotsky (2017), our network employs only Lipschitz functions $\mathbf{g}_{\boldsymbol{\zeta}}$ rather than a broad family of unrestricted functions. As illustrated in (18), we incorporate time t as an additional input, and the network size is determined solely by the k-dimensional space due to the encoder-decoder architecture. Our results indicate that the error bound is determined by k and remains free of the Lipschitz parameters γ_1 and γ_2.

Technical Challenges and Proof Overview. One key challenge lies in approximating the score function under the factor model (8) when data presents high-dimensional noise $\boldsymbol{\varepsilon}$. To address this challenge, we utilize the score function decomposition in (16) to separately approximate the low-dimensional term $\boldsymbol{\xi}(\mathbf{z}, t)$ and the noise-related term $\boldsymbol{\Lambda}_t^{-1/2}\mathbf{r}$. With the designed network architecture in (18), the noise-related term can be perfectly captured by setting $\mathbf{D}_t = \boldsymbol{\Lambda}_t^{-1}$. For the low-dimensional term, we provide an approximation based on a partition of $\mathbb{R}^k$ into a compact subset $\mathcal{C} = \{\mathbf{z} \in \mathbb{R}^k : \|\mathbf{z}\|_2 \leq S\}$ with a radius $S = \mathcal{O}(\sqrt{(1+\sigma_{\max}^2)(k+\log(1/\epsilon))})$ and its complement. Specifically, we construct a network $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$ to achieve an L^∞ approximation guarantee within the set $\mathcal{C} \times [0, T]$, and take $\overline{\mathbf{g}}_{\boldsymbol{\zeta}} = 0$ in the complement of $\mathcal{C} \times [0, T]$.

To construct $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$ as an approximation to $\boldsymbol{\xi}(\mathbf{z}, t)$ over the domain $\mathcal{C} \times [0, T]$, we begin by forming a uniform grid of hypercubes covering $\mathcal{C} \times [0, T]$ and build local approximations within each hypercube.

For the i-th component ξ_i of $\boldsymbol{\xi}$, we use a Taylor polynomial $\bar{g}_i$ to obtain a local approximation satisfying $\|\bar{g}_i - \xi_i\|_\infty = \mathcal{O}(\epsilon)$ on each hypercube. Since ReLU networks can approximate polynomials to arbitrary accuracy in the L^∞ norm, we construct a network $\bar{g}_{\boldsymbol{\zeta},i}$ that approximates $\bar{g}_i$ within error $\epsilon/2$. By combining these approximations across all hypercubes, we obtain a network $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$ that achieves an L^∞ approximation of $\boldsymbol{\xi}$ on $\mathcal{C} \times [0, T]$.

Finally, the proof of Theorem 1 is completed by showing that the L^2 approximation error on the complement of $\mathcal{C} \times [0, T]$ can be well controlled due to the sub-Gaussian tail property assumed in Assumption 2. Note that the designed network architecture takes the form $\overline{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{r}, t) = \alpha_t \boldsymbol{\Lambda}_t^{-1} \boldsymbol{\beta}\, \overline{\mathbf{g}}_{\boldsymbol{\zeta}}(\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{-1} \mathbf{r}, t) - \boldsymbol{\Lambda}_t^{-1} \mathbf{r}$. See the details in Appendix B.1.

4.2 Theory of Score Estimation

We now turn to the estimation of score functions using a finite number of samples. With the score function parameterized by $\mathcal{S}_{\mathrm{NN}}$ in (18), we can express the score matching objective as

$$\widehat{\mathbf{s}}_{\boldsymbol{\theta}} = \underset{\mathbf{s}_{\boldsymbol{\theta}} \in \mathcal{S}_{\mathrm{NN}}}{\arg\min}\, \widehat{\mathcal{L}}(\mathbf{s}_{\boldsymbol{\theta}}), \tag{23}$$

where recall $\widehat{\mathcal{L}}$ is defined in (7). Given n i.i.d. samples, we provide an L^2 error bound for the neural score estimator $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$. The result is presented in the following theorem.

Theorem 2. *Suppose Assumptions 1-3 hold. We choose $\mathcal{S}_{\mathrm{NN}}$ in Theorem 1 with $\epsilon = n^{-\frac{1-\delta(n)}{k+5}}$ for $\delta(n) = \frac{(k+10)\log(\log n)}{2\log n}$. Given n i.i.d. samples from P_{data}, with probability $1 - \frac{1}{n}$, it holds that*

$$\frac{1}{T - t_0} \int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t \sim P_t} \left[\|\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t, t) - \nabla \log p_t(\mathbf{R}_t)\|_2^2 \right] \mathrm{d}t = \tilde{\mathcal{O}}\left(\frac{1}{t_0} (1 + \sigma_{\max}^{2k}) d^{\frac{5}{2}} k^{\frac{k+10}{2}} n^{-\frac{2-2\delta(n)}{k+5}} \log^4 n \right),$$

where $\tilde{\mathcal{O}}(\cdot)$ omits factors associated with L_s and polynomial factors on $\log t_0$, $\log d$, and $\log k$.

Discussion on the Convergence Rate. The convergence rate in Theorem 2 depends not only on the intrinsic factor dimension k but also mildly on the asset return dimension d, which appears only in a non-leading polynomial term. This polynomial dependency arises because the noise term $\boldsymbol{\varepsilon}$ spans the entire $\mathbb{R}^d$ space, introducing a truncation error component that scales with d. Fortunately, this dependency does not appear in the leading term $n^{-\frac{2-2\delta(n)}{k+5}}$, where $\delta(n) = \frac{(k+10)\log\log n}{2\log n}$. This suggests that the convergence rate is primarily dominated by the sample size n and the latent factor dimensionality k, rather than the ambient dimensionality d. When n is sufficiently large, $\delta(n)$ becomes negligible, indicating the squared L^2 estimation error to converge at the rate of $\tilde{\mathcal{O}}\left(\frac{1}{t_0} (1 + \sigma_{\max}^{2k}) d^{\frac{5}{2}} k^{\frac{k+10}{2}} n^{-\frac{2}{k+5}} \log^4 n \right)$.

Proof Sketch. The full proof is deferred to Appendix B.2; here, we present a sketch of the main argument. The proof relies on a decomposition of the population loss $\mathcal{L}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}})$. Specifically, for any

$a \in (0,1)$, it holds that

$$\mathcal{L}\left(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\right) \leq \underbrace{\mathcal{L}^{\text{trunc}}\left(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\right) - (1+a)\widehat{\mathcal{L}}^{\text{trunc}}\left(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\right)}_{(A)} + \underbrace{\mathcal{L}\left(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\right) - \mathcal{L}^{\text{trunc}}\left(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\right)}_{(B)} + (1+a) \underbrace{\inf_{\mathbf{s}_{\boldsymbol{\theta}} \in \mathcal{S}_{\text{NN}}} \widehat{\mathcal{L}}\left(\mathbf{s}_{\boldsymbol{\theta}}\right)}_{(C)},$$

where $\mathcal{L}^{\text{trunc}}$ is defined as

$$\mathcal{L}^{\text{trunc}}\left(\mathbf{s}_{\boldsymbol{\theta}}\right) := \int \ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) p_t(\mathbf{r}) \mathrm{d}\mathbf{r} \quad \text{with} \quad \ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) := \ell(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) \mathbb{1}\left\{\|\mathbf{r}\|_2 \leq \rho\right\},$$

and a truncation radius ρ to be determined. Here, the term (A) captures the statistical error due to finite (training) samples, while terms (B) and (C) represent sources of *bias* in the estimation of the score function. Specifically, (B) captures the domain truncation error, while (C) accounts for the approximation error of $\mathcal{S}_{\text{NN}}$. We bound terms $(A), (B)$, and (C) separately. For term (A), we utilize a Bernstein-type concentration inequality on a compact domain. In addition, we show that the term (B) is non-leading for sufficiently large radius ρ, thanks to the sub-Gaussian tail conditions. Then, we show that term (C) is bounded by the network approximation error (21) in Theorem 1. To balance these three terms, we choose $\rho = \mathcal{O}(\sqrt{d + \log n})$, $a = n^{-\frac{1-\delta(n)}{k+5}}$, and set $\mathcal{S}_{\text{NN}}$ in Theorem 1 with $\epsilon = n^{-\frac{1-\delta(n)}{k+5}}$ to obtain the desired result.

5 Theory of Distribution Estimation

This section establishes statistical guarantees for the estimation of high-dimensional return distribution. Given the neural score estimator $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$ in Theorem 2, we define the learned distribution $\widehat{P}_{t_0}$ as the marginal distribution of $\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}$ in (4), starting from $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. To assess the quality of $\widehat{P}_{t_0}$, we examine two key aspects: the estimation error relative to the ground-truth distribution P_{data} and the accuracy of reconstructing the latent factor space.

5.1 Main Results

We estimate the latent subspace using generated samples as described in Algorithm 1.

The following theorem shows that the simulated distribution and the recovered latent factor subspace are accurate with high probability.

Theorem 3. *Given the neural score estimator $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$ in Theorem 2, we choose $T = \frac{(4\gamma_1+2)(1-\delta(n))}{k+5} \log n$ and $t_0 = n^{-\frac{1-\delta(n)}{k+5}}$, where γ_1 is the Lipschitz parameter in Theorem 1. Denote $\texttt{Eigen-gap}(k) = \lambda_k(\boldsymbol{\Sigma}_0) - \lambda_{k+1}(\boldsymbol{\Sigma}_0)$ with the covariance matrix of returns $\boldsymbol{\Sigma}_0$ defined in (9). Further denote $\mathbf{U} \in \mathbb{R}^{d \times k}$ as the k-dimensional leading eigenspace of $\boldsymbol{\Sigma}_0$. Then, the following two results hold.*

1. ***Estimation of return distribution.*** *With probability $1 - 1/n$, the total variation distance*

Algorithm 1 Sampling and Singular Value Decomposition (SVD)

Require: Score network $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$ in Theorem 2, number of generated data m, and time t_0 and T.

1: Generate m random samples $\{\mathbf{R}_1, \ldots, \mathbf{R}_m\}$ at early stopping time t_0 via the backward process (4).[4]

2: Perform SVD on sample covariance matrix:

$$\widehat{\mathbf{\Sigma}}_0 := \frac{1}{m-1}\sum_{i=1}^{m}(\mathbf{R}_i - \bar{\mathbf{R}})(\mathbf{R}_i - \bar{\mathbf{R}})^\top \quad \text{with} \quad \bar{\mathbf{R}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{R}_i. \tag{24}$$

3: Obtain the largest k eigenvalues $\{\widehat{\lambda}_1, \ldots, \widehat{\lambda}_k\}$ and the corresponding k-dimensional eigenspace $\widehat{\mathbf{U}} \in \mathbb{R}^{d\times k}$.

4: **return** $\{\mathbf{R}_1, \ldots, \mathbf{R}_m\}$, $\widehat{\mathbf{\Sigma}}_0$, $\{\widehat{\lambda}_1, \ldots, \widehat{\lambda}_k\}$, and $\widehat{\mathbf{U}}$.

between $\widehat{P}_{t_0}$ and P_{data} satisfies

$$\mathrm{TV}\left(P_{data}, \widehat{P}_{t_0}\right) = \tilde{\mathcal{O}}\left((1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{2(k+5)}} \log^{\frac{5}{2}} n\right).$$

2. ***Latent subspace recovery.*** *Set $m = \tilde{\mathcal{O}}\left(\lambda_{\max}^{-2}(\mathbf{\Sigma}_0) d n^{\frac{2(1-\delta(n))}{k+5}} \log n\right)$. For any $1 \le i \le k$, with probability $1 - 1/n$, it holds that*

$$\left|\frac{\lambda_i(\widehat{\mathbf{\Sigma}}_0)}{\lambda_i(\mathbf{\Sigma}_0)} - 1\right| = \tilde{\mathcal{O}}\left(\frac{\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}}}{\lambda_i(\mathbf{\Sigma}_0)} \cdot n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\right).$$

Meanwhile, the corresponding k-dimensional eigenspace can be recovered with

$$\|\widehat{\mathbf{U}}\widehat{\mathbf{U}}^\top - \mathbf{U}\mathbf{U}^\top\|_{\mathrm{F}} = \tilde{\mathcal{O}}\left(\frac{\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+12}{4}}}{\texttt{Eigen-gap}(k)} \cdot n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\right),$$

where recall that $\widehat{\mathbf{U}}$ is the k-dimensional leading eigenspaces of $\widehat{\mathbf{\Sigma}}_0$.

A few explanations are in line.

Trade-off on Early Stopping. The distribution estimation in Theorem 3 highlights a trade-off associated with t_0. Specifically, we can upper bound $\mathrm{TV}(P_{\mathrm{data}}, \widehat{P}_{t_0})$ by three terms

$$\mathrm{TV}(P_{\mathrm{data}}, \widehat{P}_{t_0}) \le \mathrm{TV}(P_{\mathrm{data}}, P_{t_0}) + \mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0}) + \mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0}), \tag{25}$$

where $\widetilde{P}_{t_0}$ is the distribution of $\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}$, defined in (4), initialized with $\widehat{\mathbf{R}}^{\leftarrow}_0 \sim P_T$. As shown in (25), the latent distribution error $\mathrm{TV}(P_{\mathrm{data}}, \widehat{P}_{t_0})$ arises from early stopping, score network estimation, and the mixing of forward process (1). As t_0 increases, the score estimation error decreases according to Theorem 2. As a result, the error term $\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0})$ decreases. However, the early stopping error $\mathrm{TV}(P_{\mathrm{data}}, P_{t_0})$ increases due to the heavier injected Gaussian noise. Under a training horizon

of $T = \tilde{\mathcal{O}}(\log n)$, the choice of $t_0 = n^{-\frac{1-\delta(n)}{k+5}}$ optimally balances the early stopping error and the score estimation error.

Eigenspace Estimation using Generated Samples. The latent subspace estimation in Theorem 3 shows that the subspace can be accurately recovered with high probability. Specifically, generating $\tilde{\mathcal{O}}\big(dn^{\frac{2(1-\delta(n))}{k+5}} \log n\big)$ samples from the trained diffusion model ensures that the eigenvalues and eigenspace of the sample covariance matrix $\widehat{\mathbf{\Sigma}}_0$ closely approximate those of $\mathbf{\Sigma}_0$, with the error proportional to the score estimation error. Moreover, if $\texttt{Eigen-gap}(k)$ increases—indicating an improvement in the factor model identification, then the estimation error of the k-dimensional eigenspace decreases.

Further Discussion on Dimension Dependence. Our sample complexity bounds in Theorem 3 circumvent the curse of ambient dimensionality d under very mild assumptions, namely, score function being Lipschitz and the distribution of factors being sub-Gaussian. In the meantime, we emphasize that our focus is not on optimizing the non-leading term (e.g., $k^{\frac{k+10}{4}}$) to derive a sharp bound, but on the structural dependence on k (versus d). As a result, these bounds characterize learning efficiency being adaptive to the subspace dimension k even in the most challenging scenarios. In practical applications, however, data distributions often possess more favorable regularity properties—such as higher-order smoothness in the score function or the distribution of returns—which may lead to better learning efficiency compared to the theoretical bound. For example, if one assumes that the k-dimensional subspace score function is Hölder-s continuous, the error bound can be improved to $n^{-\frac{s}{2s+k}}$. As $s \to \infty$, the rate is approximately $n^{-\frac{1}{2}}$, which has been shown to be optimal (Tsybakov 2009). While refining our bounds under such additional properties is beyond the scope of this paper, we present comprehensive numerical and empirical results in Sections 6 and 7 to illustrate the strong performance of diffusion factor models, particularly in the small-data regime.

Proof Sketch. The proof is deferred to Appendix C.1; here, we highlight its main ideas. The outline has two parts: (I) the key steps in establishing the distribution estimation result in Theorem 3, and (II) the technical components for proving the latent subspace recovery results, emphasizing novel coupling and concentration arguments.

(I) Estimation of return distribution. We bound each term in the decomposition (25) separately.

1. Term $\mathrm{TV}(P_{\text{data}}, P_{t_0})$ is the early-stopping error. By direct calculations using the Gaussian transition kernel, we show that it is bounded by $\mathcal{O}(dt_0)$.

2. Term $\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0})$ captures the statistical estimation error. We apply Girsanov's Theorem (Karatzas and Shreve 1991, Theorem 5.1; Revuz and Yor 2013, Theorem 1.4) to show that the KL divergence $\mathrm{KL}(P_{t_0}, \widetilde{P}_{t_0})$ is bounded by the L^2 score estimation error developed in Theorem 2. Further, by Pinsker's inequality (Tsybakov 2009, Lemma 2.5), we convert the KL divergence bound into a total variation distance bound.

3. Term $\mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0})$ reflects the mixing error of the forward process (1). Using the data processing inequality (Thomas and Joy 2006, Theorem 2.8.1), we show that it is a non-leading error term of order $\tilde{\mathcal{O}}(\exp(-T))$.

(II) Latent subspace recovery. The crux is to bound the covariance estimation error $\|\widehat{\mathbf{\Sigma}}_0 - \mathbf{\Sigma}_0\|_{\mathrm{op}}$ by the following lemma.

Lemma 2. *Assume the same assumptions as in Theorem 3 and take $\widehat{\mathbf{\Sigma}}_0$ as the estimator in (24) with m samples from Algorithm 1. It holds that, with probability at least $1-\delta$,*

$$\|\widehat{\mathbf{\Sigma}}_0 - \mathbf{\Sigma}_0\|_{\mathrm{op}} = \mathcal{O}\bigg(\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\bigg). \tag{26}$$

Here, m satisfies

$$m = \mathcal{O}\left(\lambda_{\max}^{-2}(\mathbf{\Sigma}_0) d n^{\frac{2(1-\delta(n))}{k+5}} \log n\right). \tag{27}$$

The complete proof of Lemma 2 is deferred to Appendix C.2.1. Using Lemma 2 in combination with Weyl's theorem and Davis-Kahan theorem (Davis and Kahan 1970), we derive the desired results for latent subspace recovery.

Proving Lemma 2 is similar to that for the estimation of return distribution. We upper bound $\|\widehat{\mathbf{\Sigma}}_0 - \mathbf{\Sigma}_0\|_{\mathrm{op}}$ by

$$\left\|\widehat{\mathbf{\Sigma}}_0 - \mathbf{\Sigma}_0\right\|_{\mathrm{op}} \le \underbrace{\left\|\mathbf{\Sigma}_0 - \mathbf{\Sigma}_{t_0}\right\|_{\mathrm{op}}}_{(A)} + \underbrace{\left\|\mathbf{\Sigma}_{t_0} - \widetilde{\mathbf{\Sigma}}_{t_0}\right\|_{\mathrm{op}}}_{(B)} + \underbrace{\left\|\widetilde{\mathbf{\Sigma}}_{t_0} - \check{\mathbf{\Sigma}}_{t_0}\right\|_{\mathrm{op}}}_{(C)} + \underbrace{\left\|\widehat{\mathbf{\Sigma}}_0 - \check{\mathbf{\Sigma}}_{t_0}\right\|_{\mathrm{op}}}_{(D)},$$

where $\mathbf{\Sigma}_{t_0}$, $\widetilde{\mathbf{\Sigma}}_{t_0}$, $\check{\mathbf{\Sigma}}_{t_0}$ are the covariance of P_{t_0}, $\widetilde{P}_{t_0}$ and $\widehat{P}_{t_0}$, respectively. Analogous to the upper bound of the total variation distance in (25), term (A) corresponds to the early-stopping error; term (B) captures the statistical estimation error; and term (C) reflects the mixing error. The additional term (D) represents a finite-sample concentration error arising from the use of m samples in Algorithm 1.

We bound each term separately. Term (A) can be bounded by direct calculations using the Gaussian transition kernel; term (D) is bounded using matrix concentration inequalities (Vershynin 2018, Theorems 3.1.1 and 4.6.1). However, bounding terms (B) and (C) requires a novel analysis, as small total variation distances $\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0})$ and $\mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0})$ do not immediately imply small error bounds on the covariance matrix. In fact, we show the following L^2 bound based on a coupling between two backward SDEs, which converts to bounds on (B) and (C).

Lemma 3. *Assume the same assumptions as in Theorem 3. Consider the following coupled SDEs:*

$$\begin{cases} \mathrm{d}\mathbf{R}_t^{\leftarrow} &= \left(\frac{1}{2}\mathbf{R}_t^{\leftarrow} + \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right)\mathrm{d}t + \mathrm{d}\bar{\mathbf{W}}_t, \quad \text{with} \quad \mathbf{R}_0^{\leftarrow} \sim P_T, \\ \mathrm{d}\widehat{\mathbf{R}}_t^{\leftarrow} &= \left(\frac{1}{2}\widehat{\mathbf{R}}_t^{\leftarrow} + \widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\widehat{\mathbf{R}}_t^{\leftarrow}, T-t)\right)\mathrm{d}t + \mathrm{d}\bar{\mathbf{W}}_t, \quad \text{with} \quad \widehat{\mathbf{R}}_0^{\leftarrow} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d) \text{ or } P_T, \end{cases} \tag{28}$$

where P_T is the terminal distribution of the forward SDE (1). It holds that

$$\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2 = \mathcal{O}\left((1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\right). \tag{29}$$

The proof of Lemma 3 is deferred to Appendix C.2.2. By the Cauchy-Schwarz inequality and Lemma 3, we bound $\|\mathbf{\Sigma}_{t_0} - \widetilde{\mathbf{\Sigma}}_{t_0}\|_{\text{op}}$ as well as $\|\widetilde{\mathbf{\Sigma}}_{t_0} - \check{\mathbf{\Sigma}}_{t_0}\|_{\text{op}}$ by $\mathcal{O}\big(\sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \cdot \big(\sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2} + \sqrt{\mathbb{E}\|\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2}\big)\big)$, where the second moments of $\mathbf{R}^{\leftarrow}_{T-t_0}$ and $\hat{\mathbf{R}}^{\leftarrow}_{T-t_0}$ are clearly finite. Putting together all the error terms, we complete the proof of Lemma 2.

5.2 Highlights of Technical Novelties

With the statements of our theoretical results in place, we now summarize the main differences between our setting and that of Chen et al. (2023a), and highlight the corresponding technical novelties.

Difference in Data Structure. Our setting differs fundamentally from Chen et al. (2023a) because our data follow a factor-model structure with heterogeneous idiosyncratic noise that spans the full high-dimensional space, rather than lying on a noise-free low-dimensional subspace, which is fundamental and crucial for financial applications. The perturbed data $\mathbf{R}_t$ therefore contain two sources of noise—a homogeneous diffusion noise with variance h_t and a heterogeneous residual noise $\alpha_t^2\sigma_i^2$—so the i-th coordinate is perturbed by Gaussian noise with a total variance of $h_t + \alpha_t^2\sigma_i^2$. This heterogeneity introduces substantial technical challenges, particularly in controlling how this noise propagates into the final approximation error.

Technical Differences/Novelties. To address the challenges posed by high-dimensional idiosyncratic noise, we develop a time-varying score decomposition alongside a tailored neural network architecture. Specifically, we derive a time-varying orthogonal decomposition of the score into subspace and complementary components, which we operationalize through a factor-aware, time-varying encoder-decoder with skip connections. We further introduce a time-dependent projection matrix that allows the model to effectively accommodate heterogeneous, high-dimensional noise. In contrast, in the noise-free linear setting of Chen et al. (2023a), a fixed projection matrix suffices for both score decomposition and network design.

From a technical standpoint, Lemmas 2 and 3 are both novel and essential to our analysis, and they highlight a key distinction between our work and Chen et al. (2023a):

- Lemma 2 establishes an error bound for covariance estimation and latent subspace recovery using samples generated by Algorithm 1, with only mild dependence on d (non-leading order) despite the presence of heterogeneous high-dimensional noise. Because the idiosyncratic noise is full-rank, the problem cannot be reduced to a k-dimensional setting, making the analysis substantially more challenging. Consequently, our theoretical guarantees are derived in the

full d-dimensional space, in contrast to Chen et al. (2023a), who work entirely within the k-dimensional latent subspace.

- Lemma 3 quantifies the discrepancy between the reverse processes driven by the true score $\nabla \log p_t$ and the learned score $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$, thereby converting score estimation error into distributional error (e.g., total variation). Unlike Chen et al. (2023a), which analyzes a time-reverse process projected onto a time-invariant subspace, our setting involves a time-varying latent subspace induced by the projections $\boldsymbol{\Lambda}_t^{-1/2}\boldsymbol{\beta}$. This introduces significant analytical challenges in characterizing the relationship between the corresponding time-varying projected backward processes. To overcome these difficulties, we develop a coupling argument that enables a general comparison between the true and estimated reverse dynamics $\mathbf{R}^{\leftarrow}_{T-t_0}$ and $\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}$ in (3) and (4).

6 Numerical Study with Synthetic Data

In this section, we use our diffusion factor model to learn high-dimensional asset returns under a synthetic factor model setup. We numerically evaluate its effectiveness in terms of recovering both the latent subspace and the return distribution (as in Theorem 3).

To simulate a practically challenging scenario, we follow the widely used practice in the econometrics literature (Bai and Ng 2002, 2023) to simulate a latent factor model with $d = 2^{11} = 2048$ assets and $k = 16$ factors, which satisfies Assumptions 1–3 in our framework. For diffusion models, we use a U-Net (Ronneberger, Fischer, and Brox 2015) as a practical implementation of our theoretical neural network $\mathcal{S}_{\text{NN}}$ in (18), and set its bottleneck width as the factor dimension k. Appendix D provides more details on how we construct simulated returns and train diffusion models.

We denote $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ as the ground truth mean and covariance matrix of returns. Similarly, we denote $\boldsymbol{\mu}_{\text{Diff}}$ and $\boldsymbol{\Sigma}_{\text{Diff}}$ as the mean and covariance matrix estimated using diffusion-generated data, and $\boldsymbol{\mu}_{\text{Emp}}$ and $\boldsymbol{\Sigma}_{\text{Emp}}$ as the empirical mean and covariance matrix estimated using training data.

Latent Subspace Recovery. We compare the following two methods to recover the latent subspace:

1. `Diff Method`: Our proposed diffusion factor model—we first estimate the return distribution using our diffusion factor model trained on the training dataset, then generate a large set of new data, and finally apply principal component analysis (PCA) on the generated data to estimate the eigenvalues and eigenspaces.

2. `Emp Method`: A naïve PCA method—we directly perform PCA on the training data and extract the leading eigenvalues and eigenspaces.

It is worth noting that the comparison between two methods is fair because the above two methods have access to exactly the same training data.

We denote $\{\lambda_i\}_{1\le i\le k}$ as the top-k eigenvalues and $\mathbf{U}\mathbf{U}^\top$ as the leading k-dimensional principal components of the ground-truth $\mathbf{\Sigma}$. We perform SVD on $\mathbf{\Sigma}_{\text{Diff}}$ (resp. $\mathbf{\Sigma}_{\text{Emp}}$) to extract the top-$k$ eigenvalues $\{\lambda_i^{\text{Diff}}\}_{1\le i\le k}$ (resp. $\{\lambda_i^{\text{Emp}}\}_{1\le i\le k}$) and the leading k-dimensional principal components $\left(\mathbf{U}\mathbf{U}^\top\right)_{\text{Diff}}$ (resp. $\left(\mathbf{U}\mathbf{U}^\top\right)_{\text{Emp}}$).

To assess the accuracy of the eigenvalue estimation, we compute the ℓ^1 relative error for `Diff Method` and `Emp Method` as

$$\text{Diff RE}_1 = \frac{1}{k}\sum_{i=1}^{k}\left|\frac{\lambda_i^{\text{Diff}}}{\lambda_i} - 1\right| \quad \text{and} \quad \text{Emp RE}_1 = \frac{1}{k}\sum_{i=1}^{k}\left|\frac{\lambda_i^{\text{Emp}}}{\lambda_i} - 1\right|. \tag{30}$$

To evaluate the recovery of the principal components, we compute the relative Frobenius norm errors for the two methods as

$$\text{Diff RE}_2 = \frac{\left\|\left(\mathbf{U}\mathbf{U}^\top\right)_{\text{Diff}} - \mathbf{U}\mathbf{U}^\top\right\|_{\text{F}}}{\left\|\mathbf{U}\mathbf{U}^\top\right\|_{\text{F}}} \quad \text{and} \quad \text{Emp RE}_2 = \frac{\left\|\left(\mathbf{U}\mathbf{U}^\top\right)_{\text{Emp}} - \mathbf{U}\mathbf{U}^\top\right\|_{\text{F}}}{\left\|\mathbf{U}\mathbf{U}^\top\right\|_{\text{F}}}. \tag{31}$$

Table 1 reports the errors in estimating the top-k eigenvalues (30) in Panel A and the errors in recovering the k-dimensional principal components (31) in Panel B for both the `Diff Method` and `Emp Method`, for a variety of training sample sizes $N = 2^9, 2^{10}, \ldots, 2^{13}$.

Table 1: Relative error of the estimated top-k eigenvalues (30) and k-dimensional principal components (31) for varying sample sizes (standard deviations in parentheses).

Panel A: Eigenvalues			
N	**Diff RE**$_1$	**Emp RE**$_1$	**Diff RE**$_1$**/Emp RE**$_1$
$2^9 = 512$	0.144 ($\pm$ 0.011)	0.160	0.898 ($\pm$ 0.069)
$2^{10} = 1024$	0.130 ($\pm$ 0.008)	0.141	0.919 ($\pm$ 0.056)
$2^{11} = 2048$	0.116 ($\pm$ 0.005)	0.121	0.957 ($\pm$ 0.041)
$2^{12} = 4096$	0.081 ($\pm$ 0.004)	0.081	1.003 ($\pm$ 0.047)
$2^{13} = 8192$	0.069 ($\pm$ 0.003)	0.067	1.024 ($\pm$ 0.045)
Panel B: Principal Components			
N	**Diff RE**$_2$	**Emp RE**$_2$	**Diff RE**$_2$**/Emp RE**$_2$
$2^9 = 512$	0.247 ($\pm$ 0.012)	0.274	0.901 ($\pm$ 0.044)
$2^{10} = 1024$	0.202 ($\pm$ 0.006)	0.218	0.926 ($\pm$ 0.028)
$2^{11} = 2048$	0.153 ($\pm$ 0.005)	0.159	0.960 ($\pm$ 0.033)
$2^{12} = 4096$	0.110 ($\pm$ 0.004)	0.109	1.009 ($\pm$ 0.036)
$2^{13} = 8192$	0.085 ($\pm$ 0.004)	0.084	1.012 ($\pm$ 0.047)

Table 1 reveals the advantage of our method in small-data regimes ($N \le d$), which is particularly important for practical applications. In particular, when $N \le 2048$, `Diff Method` consistently outperforms `Emp Method`, as shown by error ratios being statistically below 1. When there is enough sample ($N \ge 4096$), simply using empirical estimates suffices to yield good subspace recovery. It is worth highlighting that $N = 2048$ corresponds to approximately 8 years of daily return observations or 39 years of weekly return observations. It is rarely the case that one enjoys the luxury of having

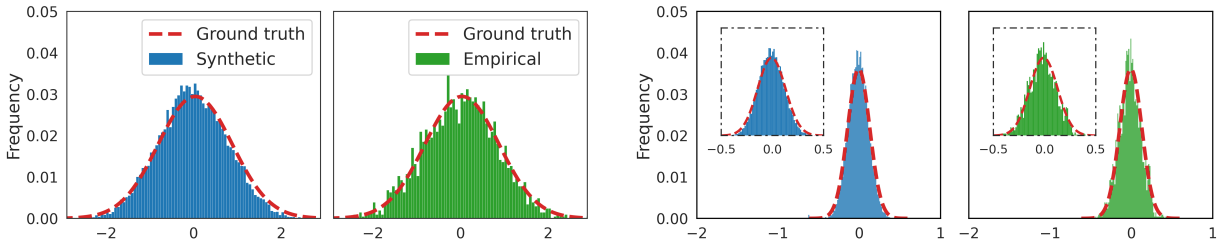
that much data to estimate a factor model, because return distributions do not remain stable over such a long period of time.

Generated Return Distribution. In Figure 1, we visualize the (empirical) return distribution generated by our diffusion factor model (trained on 2^{11} samples) for a few selected assets, which is compared with direct sampling from the ground truth. With the same number of 2^{11} samples, `Diff Method` produces a *smoother* empirical distribution that more closely approximates the ground truth. This suggests that our diffusion factor model may be more effective at capturing patterns and regularities of the underlying distribution than direct sampling.

Figure 1: Examples of asset return distribution (the blue is constructed using output samples from the diffusion model and the green is based on samples from the ground truth.)

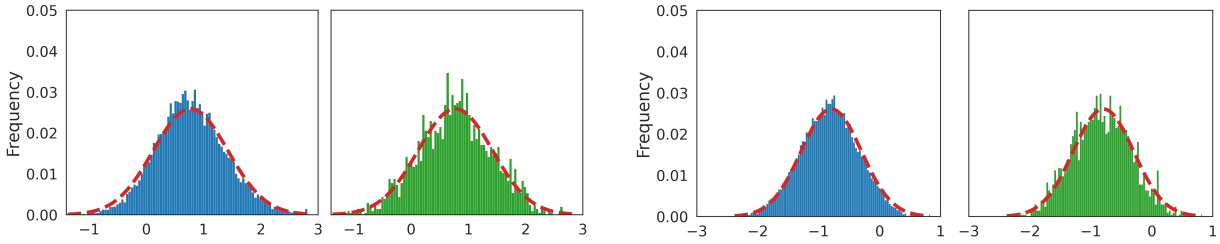
(a) Asset with the largest variance.

(b) Asset with the smallest variance.



(c) Asset with the largest mean.

(d) Asset with the smallest mean.



Moreover, to evaluate the accuracy of estimating the first two moments, Table 2 reports the relative estimation errors for the mean and the covariance matrix, across a range of sample sizes $N = 2^9, 2^{10}, \ldots, 2^{13}$. Specifically, we compute the ℓ^2 relative error of the mean and the relative Frobenius-norm error of the covariance matrix for `Diff Method` and `Emp Method` as follows:

$$\text{Diff RE}_3 = \frac{\|\boldsymbol{\mu}_{\text{Diff}} - \boldsymbol{\mu}\|_2}{\|\boldsymbol{\mu}\|_2}, \quad \text{Emp RE}_3 = \frac{\|\boldsymbol{\mu}_{\text{Emp}} - \boldsymbol{\mu}\|_2}{\|\boldsymbol{\mu}\|_2}, \tag{32}$$

$$\text{Diff RE}_4 = \frac{\|\boldsymbol{\Sigma}_{\text{Diff}} - \boldsymbol{\Sigma}\|_{\text{F}}}{\|\boldsymbol{\Sigma}\|_{\text{F}}}, \quad \text{and} \quad \text{Emp RE}_4 = \frac{\|\boldsymbol{\Sigma}_{\text{Emp}} - \boldsymbol{\Sigma}\|_{\text{F}}}{\|\boldsymbol{\Sigma}\|_{\text{F}}}. \tag{33}$$

Table 2 further demonstrates that, in small-sample regimes ($N \leq 2048$), the `Diff Method` consistently attains lower estimation errors for both the mean and covariance than the `Emp Method`, as evidenced by error ratios below 1. For $N \geq 4096$, the two methods become indistinguishable.

Table 2: Relative error of the estimated mean (32) and covariance matrix (33) for varying sample sizes (standard deviations in parentheses).

Panel A: Mean			
N	**Diff RE**$_3$	**Emp RE**$_3$	**Diff RE**$_3$**/Emp RE**$_3$
$2^9 = 512$	0.089 ($\pm$ 0.007)	0.097	0.916 ($\pm$ 0.070)
$2^{10} = 1024$	0.057 ($\pm$ 0.002)	0.060	0.954 ($\pm$ 0.033)
$2^{11} = 2048$	0.047 ($\pm$ 0.001)	0.048	0.966 ($\pm$ 0.023)
$2^{12} = 4096$	0.026 ($\pm$ 0.001)	0.026	1.001 ($\pm$ 0.021)
$2^{13} = 8192$	0.022 ($\pm$ 0.001)	0.022	1.009 ($\pm$ 0.020)
Panel B: Covariance Matrix			
N	**Diff RE**$_4$	**Emp RE**$_4$	**Diff RE**$_4$**/Emp RE**$_4$
$2^9 = 512$	0.257 ($\pm$ 0.016)	0.290	0.885 ($\pm$ 0.055)
$2^{10} = 1024$	0.209 ($\pm$ 0.011)	0.230	0.911 ($\pm$ 0.046)
$2^{11} = 2048$	0.157 ($\pm$ 0.005)	0.167	0.941 ($\pm$ 0.027)
$2^{12} = 4096$	0.115 ($\pm$ 0.002)	0.115	0.997 ($\pm$ 0.018)
$2^{13} = 8192$	0.089 ($\pm$ 0.001)	0.088	1.007 ($\pm$ 0.009)

Statistical Interpretation. From a statistical perspective, our superior performance can be explained by the bias–variance trade-off. The `Emp Method` has small bias but often suffers from high variance in small-data regimes. In contrast, the `Diff Method` first fits a diffusion model to the available data and then uses the trained model to generate a large number of samples for downstream estimation tasks. Once the diffusion model is well trained (using observed data) to represent the underlying unknown distribution, our method can generate arbitrarily many additional samples. The `Diff Method` can therefore be viewed as a data-dependent regularization of the `Emp Method`, increasing the effective sample size to reduce estimation variance at the cost of a small modeling bias (Li et al. 2025). In small-data regimes, estimation variance typically dominates the error (Jorion 1986, Ledoit and Wolf 2003, 2004), and the resulting variance reduction leads the `Diff Method` to outperform the `Emp Method`. This insight is further developed in the next section through empirical analysis.

7 Empirical Analysis

In this section, we apply our diffusion factor model to real-world data and evaluate its economic relevance in constructing both mean-variance optimal portfolios (Zhou and Li 2000, DeMiguel et al. 2009) and factor portfolios (Giglio, Kelly, and Xiu 2022, Feng et al. 2023). Although our theoretical analysis relies on distributional assumptions of sub-Gaussian tails, we challenge our framework with real asset returns, which are well documented to be heavy-tailed while still well captured by factor models (Chamberlain and Rothschild 1983, Cont 2001, Fan, Liao, and Mincheva 2013). Specifically, Section 7.1 compares mean-variance optimal portfolios derived from diffusion-generated data with those based on other robust portfolio rules in the literature. Section 7.2 assesses the performance of factor portfolios estimated using diffusion-generated data and benchmarks them against other

prominent factor models in the literature.

We use daily excess return data for U.S. stocks from May 1, 2001, to April 30, 2024.[5] The dataset is obtained from the Center for Research in Security Prices (CRSP), available through Wharton Research Data Services. We use a five-year rolling window to estimate the diffusion model, and we update the estimation quarterly. Specifically, at the beginning of each rebalancing quarter T (i.e., on February 1, May 1, August 1, and November 1), we update model parameters using training data from the preceding five years ($T-20$ to T offset quarters). We test the model on data in the next quarter $T+1$ to evaluate out-of-sample performance. Appendix E.1 provides more details on data preprocessing and training of diffusion models. Appendix E.4 provides a robustness check with annually updated diffusion models.[6]

7.1 Mean-Variance Optimal Portfolio

We follow the literature to consider the mean-variance optimization problem with a norm constraint (Jagannathan and Ma 2003, Bertsimas, Pachamanova, and Sim 2004, DeMiguel et al. 2009, Gotoh and Takeda 2011) to yield a fully invested and reasonably diversified portfolio:

$$\max_{\boldsymbol{\omega}} \ \boldsymbol{\omega}^\top \boldsymbol{\mu} - \frac{\eta}{2}\boldsymbol{\omega}^\top \boldsymbol{\Sigma}\boldsymbol{\omega}, \quad \text{subject to } \boldsymbol{\omega}^\top \mathbf{1} = 1 \text{ and } \|\boldsymbol{\omega}\|_\infty \leq 0.05, \tag{34}$$

where $\boldsymbol{\omega}$ denotes the portfolio weights, $\boldsymbol{\mu}$ is the expected return in excess of the risk-free rate, $\boldsymbol{\Sigma}$ is the covariance matrix, and $\eta > 0$ is the risk aversion parameter. As a robustness check, we also consider an ℓ_1-norm constraint when solving the portfolio weights (Bertsimas, Pachamanova, and Sim 2004, Gotoh and Takeda 2011), and report the corresponding results in Appendix E.3. [7]

Methods of Portfolio Construction. We evaluate a series of portfolio construction methods that differ in their data source (real observed data or diffusion-generated data) and in their estimation techniques for the mean and covariance matrix. We first describe classical approaches that rely solely on observed data.

1. `EW`: A simple strategy with equal weights on all risky assets. DeMiguel, Garlappi, and Uppal (2009) have documented its surprisingly efficient and robust performance.

2. `VW`: A value-weighted strategy that assigns each asset a weight proportional to its market capitalization relative to the total market capitalization in the dataset.

3. `Real Emp+Real Emp`: A baseline that directly uses the sample mean $\boldsymbol{\mu}_{\text{Emp}}$ and sample covariance matrix $\boldsymbol{\Sigma}_{\text{Emp}}$ as inputs to (34) to solve the optimal portfolio weights.

As the empirical estimator is suboptimal and may be unstable in small-data regimes, we also include shrinkage estimators, which are well documented to improve upon empirical estimators in such settings (Jorion 1986, Ledoit and Wolf 2003, 2004).

4. `Real BS+Real Emp`: A robust portfolio proposed by Jorion (1986) that utilizes a Bayes-Stein shrinkage mean $\boldsymbol{\mu}_{\text{BS}}$:

$$\boldsymbol{\mu}_{\text{BS}} = (1 - \gamma_{\text{BS}}) \cdot \boldsymbol{\mu}_{\text{Emp}} + \gamma_{\text{BS}} \cdot \mu_{\text{gmv}} \mathbf{1}_d$$

to solve (34), where $\mu_{\text{gmv}} = \mathbf{1}_d^\top \boldsymbol{\Sigma}_{\text{Emp}}^{-1} \boldsymbol{\mu}_d / \mathbf{1}_d^\top \boldsymbol{\Sigma}_{\text{Emp}}^{-1} \mathbf{1}_d$ denotes the average excess return on the sample global minimum-variance portfolio, and γ_{BS} is the shrinkage weight estimated by Jorion (1986, Equation (17)). The covariance estimator is still the sample covariance $\boldsymbol{\Sigma}_{\text{Emp}}$.

5. `Real OLSE+Real Emp`: A robust portfolio proposed by Bodnar, Okhrin, and Parolya (2019) that uses an Optimal Linear Shrinkage Estimator (OLSE) for high-dimensional mean $\boldsymbol{\mu}_{\text{OLSE}}$:

$$\boldsymbol{\mu}_{\text{OLSE}} = \alpha_{\text{OLSE}} \cdot \boldsymbol{\mu}_{\text{Emp}} + \beta_{\text{OLSE}} \cdot \mathbf{1}_d$$

to solve (34), where α_{OLSE} and β_{OLSE} are the shrinkage weights estimated by Jorion (1986, Equations (6) and (7)). The covariance estimator is still the sample covariance $\boldsymbol{\Sigma}_{\text{Emp}}$.

6. `Real Emp+Real LW`: A robust portfolio proposed by Ledoit and Wolf (2003, 2004) that uses a shrinkage covariance matrix $\boldsymbol{\Sigma}_{\text{LW}}$:

$$\boldsymbol{\Sigma}_{\text{LW}} = (1 - \gamma_{\text{LW}}) \cdot \boldsymbol{\Sigma}_{\text{Emp}} + \gamma_{\text{LW}} \cdot u \mathbf{I}_d$$

to solve (34), where $u = \text{tr}(\boldsymbol{\Sigma}_{\text{Emp}})/d$, and γ_{LW} is the shrinkage parameter estimated by Ledoit and Wolf (2022, Equation (2.14)). The mean estimator is still the sample mean $\boldsymbol{\mu}_{\text{Emp}}$.

7. `Real BS+Real LW`: A robust portfolio that combines Bayes–Stein shrinkage mean $\boldsymbol{\mu}_{\text{BS}}$ with the shrinkage covariance $\boldsymbol{\Sigma}_{\text{LW}}$ to solve (34).

8. `Real OLSE+Real LW`: A robust portfolio that combines OLSE mean $\boldsymbol{\mu}_{\text{OLSE}}$ with the shrinkage covariance $\boldsymbol{\Sigma}_{\text{LW}}$ to solve (34).

We further consider six methods that rely on our diffusion-generated data.

9. `Diff Emp+Diff Emp`: It extends `Real Emp+Real Emp` by replacing the empirical mean and covariance estimates with those obtained from diffusion-generated data.

10. `Diff BS+Diff Emp`: It extends `Real BS+Real Emp` by replacing the Bayes-Stein mean and empirical covariance estimates with those obtained from diffusion-generated data.

11. `Diff OLSE+Diff Emp`: It extends `Real OLSE+Real Emp` by replacing the OLSE mean and empirical covariance estimates with those obtained from diffusion-generated data.

12. `Diff Emp+Diff LW`: It extends `Real Emp+Real LW` by replacing the empirical mean and Ledoit-Wolf covariance estimates with those obtained from diffusion-generated data.

13. `Diff BS+Diff LW`: It extends `Real BS+Real LW` by replacing the Bayes-Stein mean and Ledoit-Wolf covariance estimates with those obtained from diffusion-generated data.

14. `Diff OLSE+Diff LW`: It extends `Real OLSE+Real LW` by replacing the OLSE mean and Ledoit-Wolf covariance estimates with those obtained from diffusion-generated data.

Finally, we consider two additional hybrid methods.

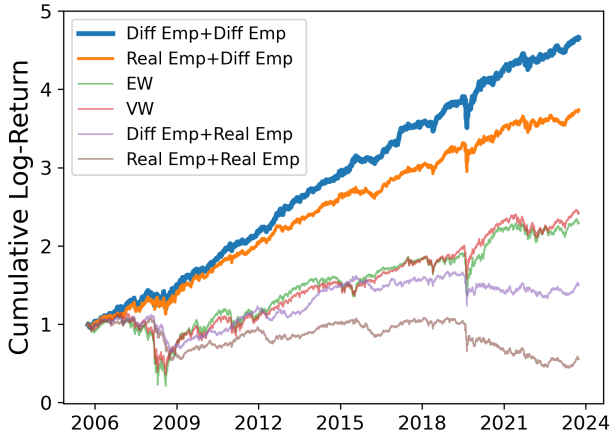
15. `Real Emp+Diff Emp`: It uses the empirical mean estimated from real data and the empirical covariance matrix estimated from diffusion-generated data to solve (34).

16. `Diff Emp+Real Emp`: It uses the empirical mean estimated from diffusion-generated data and the empirical covariance matrix estimated from real data to solve (34).

Methods 9–14 serve as diffusion-based counterparts to Methods 3–8 to evaluate the benefits of using diffusion-generated data in both mean and covariance estimation. Methods 15 and 16 are hybrid approaches designed to understand the contribution of diffusion-generated data in mean and covariance estimation, respectively.

Main Results. Target weights are updated quarterly and rebalanced daily. Following Kan and Zhou (2007), we set $\eta = 3$ and assume a transaction cost of 20 basis points. We also examine other values of η and the scenario without transaction costs, and find similar results; see Appendix E.2 for details. Table 3 reports out-of-sample portfolio performance under scenarios with transaction costs, including the average return (Mean), standard deviation (Std), Sharpe ratio (SR), certainty equivalent return (CER, i.e., the objective value in (34)), maximum drawdown (MDD), and turnover (TO).[8] Figure 2 further shows the cumulative returns of different portfolios in log scale with transaction costs.

Figure 2: Cumulative returns of different portfolios in log scale with transaction cost for $\eta = 3$.



First, using diffusion-generated data, `Diff Emp+Diff Emp` consistently outperforms by a large margin all alternative methods in Mean, SR, and CER, with transaction costs. Other diffusion-based methods also outperform their counterparts. In particular, `Diff Emp+Diff Emp` outperforms `EW` by a large margin, achieving approximately twice the Sharpe ratio. This is a highly nontrivial

Table 3: Performance of different portfolios with transaction costs for $\eta = 3$ (model updated quarterly)

Method	Mean	Std	SR	CER	MDD (%)	TO
Methods based on real observed data						
`EW`	0.100	0.206	0.486	0.037	53.128	**3.031**
`VW`	0.096	0.220	0.437	0.024	58.086	3.464
`Real Emp+Real Emp`	-0.017	0.128	-0.129	-0.041	45.011	46.722
`Real BS+Real Emp`	-0.021	0.126	-0.168	-0.045	45.864	45.612
`Real OLSE+Real Emp`	-0.039	0.127	-0.305	-0.063	55.596	45.952
`Real Emp+Real LW`	-0.003	0.123	-0.023	-0.025	38.852	38.827
`Real BS+Real LW`	-0.007	**0.121**	-0.059	-0.029	39.369	37.900
`Real OLSE+Real LW`	-0.024	0.122	-0.200	-0.047	45.864	38.543
Methods based on diffusion-generated data						
`Diff Emp+Diff Emp`	**0.216**	0.159	**1.361**	**0.178**	32.603	28.751
`Diff BS+Diff Emp`	0.213	0.157	1.357	0.176	32.610	27.978
`Diff OLSE+Diff Emp`	0.212	0.157	1.356	0.176	32.615	27.876
`Diff Emp+Diff LW`	0.180	0.152	1.186	0.145	32.781	26.353
`Diff BS+Diff LW`	0.178	0.150	1.184	0.144	32.862	25.773
`Diff OLSE+Diff LW`	0.178	0.150	1.184	0.144	32.882	25.697
Methods based on both real observed data and diffusion-generated data						
`Diff Emp+Real Emp`	0.037	0.134	0.275	0.010	37.423	29.323
`Real Emp+Diff Emp`	0.163	0.150	1.090	0.130	**30.760**	23.313

benchmark to beat, as shown by DeMiguel, Garlappi, and Uppal (2009), because all other methods without diffusion-generated data fail to beat `EW` in terms of risk-adjusted returns.

Second, `Diff Emp+Diff Emp` outperforms both hybrid methods in risk-adjusted returns. Between them, `Real Emp+Diff Emp` beats `Diff Emp+Real Emp`, and both significantly outperform sample-based methods including `Real Emp+Real Emp` and other classical shrinkage methods. This result reflects improvements in both the mean and covariance estimation from diffusion-generated data, but most of the improvements come from the improved covariance estimation, which is not surprising given the very design of our diffusion factor model. From a statistical perspective, again, the performance gains of `Diff Emp+Diff Emp` can be interpreted through a bias–variance trade-off: our diffusion factor model acts as a form of data-dependent regularization, introducing a modest modeling bias while substantially reducing finite-sample estimation variance, thereby yielding more robust moment estimators (Kotelnikov et al. 2023, Li et al. 2025).

Finally, with diffusion-generated data, shrinkage estimates of both the mean and covariance matrix are no longer necessary, as shown by the superior performance of `Diff Emp+Diff Emp` compared with other diffusion-based shrinkage methods. Although the shrinkage estimates have historically played an impactful role for robust portfolios, as shown by reviews in the literature (Avramov and Zhou 2010, Bodnar, Okhrin, and Parolya 2022, Ledoit and Wolf 2022), our results show that modern generative modeling techniques such as diffusion models may provide a simple yet effective and robust way to deal with data scarcity.

7.2 Factor Portfolio

To further demonstrate the benefits of diffusion-generated data, we apply existing statistical methods on top of our diffusion-generated data to obtain factors and evaluate the performance of the corresponding tangency portfolios.

Methods of Factor Estimation. We compare seven methods to estimate factors, where a projection matrix is first estimated from either observed data or diffusion-generated data, and then applied to test data to extract factors. Existing approaches that rely solely on observed data include:

1. `FF Method`: Firm characteristics-based factors that includes the Fama and French (2015a) five factors: market (Mkt-RF), size (SMB), value (HML), profitability (RMW), and investment (CMA), the momentum factor (MOM) of Carhart (1997), and the short-term and long-term reversal factors (ST-Rev and LT-Rev).[9]

2. `PCA Method`: Perform PCA on observed training data to obtain a projection matrix $\mathbf{W}_{\text{PCA}}$.

3. `POET Method`: Principal Orthogonal complEment Thresholding (POET) proposed by Fan, Liao, and Mincheva (2013), in which one computes a robust POET covariance estimator $\hat{\mathbf{\Sigma}}_{\text{POET}}$ and then apply SVD to obtain the projection matrix $\mathbf{W}_{\text{POET}}$.

4. `RPPCA Method`: Risk-premia PCA (RP-PCA) proposed by Lettau and Pelger (2020b), in which one performs PCA on $\frac{1}{n}\sum_{i=1}^{n}\mathbf{r}_i\mathbf{r}_i^\top+\gamma_{\text{RPPCA}}\bar{\mathbf{r}}\bar{\mathbf{r}}^\top$ to obtain a projection matrix $\mathbf{W}_{\text{RPPCA}}$, where $\{\mathbf{r}_i\}_{i=1}^n$ denotes samples of asset returns, $\bar{\mathbf{r}}=\frac{1}{n}\sum_{i=1}^{n}\mathbf{r}_i$ is the sample mean, and γ_{RPPCA} is a tuning parameter.

Methods based on our diffusion factor model are implemented by applying the same factor estimation procedures to diffusion-generated data, rather than to the observed training data:

5. `Diff+PCA Method`: It extends `PCA Method` by using diffusion-generated data.

6. `Diff+POET Method`: It extends `POET Method` by using diffusion-generated data.

7. `Diff+RPPCA Method`: It extends `RPPCA Method` by using diffusion-generated data.

Main Results. Next, we construct tangency portfolios that maximize the Sharpe ratio using the extracted factors by solving the following optimization problem:

$$\max_{\boldsymbol{\omega}} \frac{\boldsymbol{\omega}^\top\boldsymbol{\mu}_{\text{fac}}}{\sqrt{\boldsymbol{\omega}^\top\mathbf{\Sigma}_{\text{fac}}\boldsymbol{\omega}}}, \quad \text{subject to } \boldsymbol{\omega}^\top\mathbf{1}=1, \tag{35}$$

where $\boldsymbol{\omega}$ denotes the portfolio weights, and $\boldsymbol{\mu}_{\text{fac}}$ and $\mathbf{\Sigma}_{\text{fac}}$ are the mean and covariance matrix of the factors, respectively. Table 4 reports the Sharpe ratios of the tangency portfolios constructed across varying numbers of factors.[10]

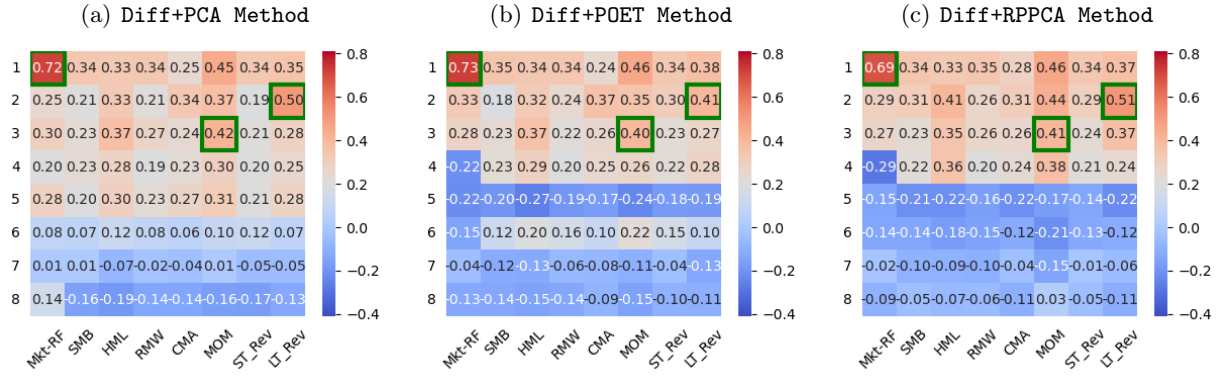
Table 4: Out-of-sample Sharpe ratios of factor tangency portfolios (model updated quarterly) . The number of factors is set to be 3, 5, 6, and 8, respectively.

# Factors	Diff+PCA	Diff+POET	Diff+RPPCA	FF	PCA	POET	RPPCA
3	1.204	1.167	**1.474**	0.431	0.480	0.856	0.857
5	2.280	2.264	**2.555**	0.474	0.519	0.963	0.928
6	2.698	2.701	**3.070**	0.552	0.630	1.375	1.433
8	3.536	3.471	**3.668**	0.810	0.735	1.811	1.861

Methods based on our diffusion factor model consistently outperform `FF Method` and their corresponding PCA counterparts. In particular, `Diff+PCA Method` exceeds both `FF Method` and `PCA Method` by a wide margin, achieving approximately three and five times their Sharpe ratios, respectively. Furthermore, applying the robust methods of factor estimation proposed by Fan, Liao, and Mincheva (2013), Lettau and Pelger (2020b) to diffusion-generated data yields additional improvements in portfolio performance. These results highlight the effectiveness of diffusion-generated factors in capturing systematic risk.

Finally, we assess whether diffusion-generated factors capture interpretable economic characteristics by analyzing their correlations with firm characteristics-based factors. For each method based on diffusion-generated data, Figure 3 reports the correlations between top eight factors estimated using diffusion-based methods and traditional factors in `FF Method`. Diffusion-generated factors exhibit notable correlations with traditional factors, with Mkt-RF, LT-REV, and MOM being the three leading factors for all three methods.

Figure 3: Correlation between the top 8 factors obtained using diffusion-based methods and those from the `FF Method`.



8 Conclusion and Future Work

We propose a diffusion factor model that embeds the latent factor structure into generative diffusion processes. To exploit the low-dimensional nature of asset returns, we introduce a time-varying score decomposition via orthogonal projections and design a score network with an encoder-decoder architecture. These modeling choices lead to a concise and structure-aware representation of the

score function.

On the theoretical front, we provide statistical guarantees for score approximation, score estimation, and distribution recovery. Our analysis introduces new techniques to address heterogeneous residual noise and time-varying subspaces, yielding error bounds that depend primarily on the intrinsic factor dimension k, with only mild dependence on the ambient dimension d. These results demonstrate that our framework effectively mitigates the curse of dimensionality in high-dimensional settings.

Simulation studies confirm that the proposed method achieves more accurate subspace recovery and smoother distribution estimation than classical baselines, particularly when the sample size is smaller than the asset dimension. The generated data reliably capture the true mean and covariance structure.

Finally, our empirical experiments on real data show that diffusion-generated data improves mean and covariance estimation, leading to superior mean-variance optimal portfolios. Our approach consistently outperforms traditional methods, achieving higher Sharpe ratios. Additionally, factors estimated from generated data exhibit interpretable economic characteristics, enabling tangency portfolios that better capture systematic risk. These findings highlight the substantial potential of our diffusion factor model for real-world financial applications.

With the widely used factor-model structure, our results open a path toward integrating financial data structures into generative AI architectures that admit statistical guarantees. Looking ahead, we plan to investigate several interesting and more sophisticated settings. These include understanding how the statistical guarantees change when the noise distribution is heavy-tailed; using diffusion models for missing-data imputation; establishing theoretical guarantees for how diffusion models improve decision-making; and developing methods for generating dynamic time-series data.

Notes

[1]The number of factors k varies from 1 to several dozen, balancing predictive power and economic interpretability (Harvey, Liu, and Zhu 2016, Giglio, Liao, and Xiu 2021).

[2]It is worth noting that complete independence from d is unattainable due to idiosyncratic noise spanning the full d-dimensional space. We achieve a mild polynomial dependence of the estimated score function on the ambient dimension d from the residual noise.

[3]The bounded-output condition $\sup_{\mathbf{r},t} \|\mathbf{g}_\zeta(\mathbf{r},t)\|_2 \le K$ is often enforced in practice by clipping the layer of ReLU networks (e.g., $g(a) = \mathrm{ReLU}(a-R) - \mathrm{ReLU}(a+R) - R$ clips to $[-R,R]$), which is a standard assumption for the score approximation in the score-based models (Vincent 2011, Bartlett, Foster, and Telgarsky 2017, Ho, Jain, and Abbeel 2020, Song et al. 2021). The sparsity constraint $\sum_{\ell=1}^{L}(\|\mathbf{b}_\ell\|_0 + \|\mathbf{W}_\ell\|_0) \le J$ directly controls the complexity of the neural network class and enters our covering-number bound in Lemma B.4. Empirically, sparsity in neural networks is typically induced by regularization; for example, explicit ℓ_1 regularization (akin to LASSO) on weight matrices induces sparsity (Srinivas, Subramanya, and Babu 2017, Louizos, Welling, and Kingma 2018) and large neural networks can be compressed by training a sparse sub-network without sacrificing performance (Han, Mao, and Dally 2016, Frankle and Carbin 2019, Hoefler et al. 2021).

[4]For practical implementation, we can use denoising diffusion probabilistic models (DDPM) discretization (Ho,

Jain, and Abbeel 2020). For $i = 1, 2, \ldots, m$,

$$\mathbf{R}_{i,t_{j-1}} = \frac{1}{\sqrt{\alpha_{t_j}}}(\mathbf{R}_{i,t_j} + (1-\alpha_t)\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_{i,t_j}, t_j)) + \frac{1-\alpha_{t_j}}{\alpha_{t_j}}\mathbf{z}_{t_j}, \text{ with } \mathbf{R}_{i,T} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d),$$

where $t_0 < t_1 < t_2 \cdots < t_\ell = T$ and $\{\mathbf{z}_t\}_{t=t_0}^{T}$ are i.i.d. following $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. In our analysis, the discretization error is a minor term that does not affect the main theoretical results. More specifically, previous work on score-based diffusion sampling (e.g., Chen et al. (2023b), Li et al. (2024b), Tang and Zhao (2025)) has shown that for a time step size Δt, the discretization error scales as $\mathcal{O}(\sqrt{\Delta t})$ and hence constitutes a negligible $o(1)$ when Δt is sufficiently small.

[5]The U.S. Securities and Exchange Commission (SEC) mandated the conversion to decimal pricing for all U.S. stock markets by April 9, 2001.

[6]With a five-year rolling window, each quarterly update replaces roughly 5% of the training data and reuses the remaining 95%, which balances information refresh against the computational cost of retraining the diffusion model. As a robustness check, an annually update frequency also leads to similar results; see Appendix E.4 for details.

[7]In the norm-constrained portfolio literature, various norms have been studied, including ℓ_1-, ℓ_2-, ℓ_∞-, and A-norm constraints (Jagannathan and Ma 2003, Bertsimas, Pachamanova, and Sim 2004, DeMiguel et al. 2009, Gotoh and Takeda 2011). It has been shown that these constraints have certain mathematical equivalence in terms of obtaining the portfolio weights (Bertsimas, Pachamanova, and Sim 2004, Gotoh and Takeda 2011). As a robustness check, we construct mean–variance portfolios under ℓ_1-norm constraints and evaluate their performance; see details in Appendix E.3.

[8]We rebalance daily to track the target weights, since daily return fluctuations cause realized weights to drift away from the targets and require small but frequent adjustments to maintain the target allocation. The reported turnover is annualized from daily trades, hence these adjustments can accumulate into a relatively high annual turnover, especially when the absolute values of the target weights are more extreme.

[9]These two factors are obtained from French's data library `https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html`.

[10]As noted by Kelly, Pruitt, and Su (2019), factor tangency portfolios may not be directly implementable, but they serve as important theoretical benchmarks for evaluating mean–variance efficiency. Compared to mean–variance portfolios, their generally higher Sharpe ratios may stem from two main sources. First, the relatively low dimensionality of the factor space compared to individual assets improves the stability of estimated means and covariances. Second, the exclusion of transaction costs can further enhance performance. Similar observations have been made in the literature; see Gu, Kelly, and Xiu (2020, 2021).

References

Acciaio, B., S. Eckstein, and S. Hou, 2024, Time-causal vae: Robust financial time series generator, *arXiv preprint arXiv:2411.02947* .

Acharya, V. V., R. Berner, R. Engle, H. Jung, J. Stroebel, X. Zeng, and Y. Zhao, 2023, Climate stress testing, *Annual Review of Financial Economics* 15, 291–326.

Adrian, T., E. Etula, and T. Muir, 2014, Financial intermediaries and the cross-section of asset returns, *The Journal of Finance* 69, 2557–2596.

Aït-Sahalia, Y., and D. Xiu, 2019, Principal component analysis of high-frequency data, *Journal of the American Statistical Association* 114, 287–303.

Albergo, M. S., N. M. Boffi, and E. Vanden-Eijnden, 2023, Stochastic interpolants: A unifying framework for flows and diffusions, *arXiv preprint arXiv:2303.08797* .

Alexander, C., 2005, The present and future of financial risk management, *Journal of Financial Econometrics* 3, 3–25.

Anderson, B. D., 1982, Reverse-time diffusion equation models, *Stochastic Processes and their Applications* 12, 313–326.

Anthony, M., and P. L. Bartlett, 2009, *Neural network learning: Theoretical foundations* (Cambridge University Press).

Avramov, D., and G. Zhou, 2010, Bayesian portfolio analysis, *Annual Review of Financial Economics* 2, 25–47.

Azangulov, I., G. Deligiannidis, and J. Rousseau, 2024, Convergence of diffusion models under the manifold hypothesis in high-dimensions, *arXiv preprint arXiv:2409.18804* .

Bagnara, M., 2024, Asset pricing and machine learning: a critical review, *Journal of Economic Surveys* 38, 27–56.

Bai, J., and S. Ng, 2002, Determining the number of factors in approximate factor models, *Econometrica* 70, 191–221.

Bai, J., and S. Ng, 2023, Approximate factor models with weaker loadings, *Journal of Econometrics* 235, 1893–1916.

Bakry, D., I. Gentil, M. Ledoux, et al., 2014, *Analysis and geometry of Markov diffusion operators*, volume 103 (Springer).

Baldi, P., and K. Hornik, 1989, Neural networks and principal component analysis: Learning from examples without local minima, *Neural Networks* 2, 53–58.

Barancikova, B., Z. Huang, and C. Salvi, 2025, Sigdiffusions: Score-based diffusion models for long time series via log-signature embeddings, in *International Conference on Learning Representations*.

Bartlett, P. L., D. J. Foster, and M. J. Telgarsky, 2017, Spectrally-normalized margin bounds for neural networks, *Advances in Neural Information Processing Systems* 30.

Bartlett, P. L., P. M. Long, G. Lugosi, and A. Tsigler, 2020, Benign overfitting in linear regression, *National Academy of Sciences* 117, 30063–30070.

Behn, M., R. Haselmann, and V. Vig, 2022, The limits of model-based regulation, *The Journal of Finance* 77, 1635–1684.

Benton, J., V. De Bortoli, A. Doucet, and G. Deligiannidis, 2024, Nearly d-linear convergence bounds for diffusion models via stochastic localization, in *International Conference on Learning Representations*.

Bertsimas, D., D. Pachamanova, and M. Sim, 2004, Robust linear optimization under general norms, *Operations Research Letters* 32, 510–516.

Bickel, P. J., and E. Levina, 2008, Regularized estimation of large covariance matrices, *The Annals of Statistics* 36, 199 – 227.

Bisias, D., M. Flood, A. W. Lo, and S. Valavanis, 2012, A survey of systemic risk analytics, *Annual Review of Finance Economics* 4, 255–296.

Bodnar, T., O. Okhrin, and N. Parolya, 2019, Optimal shrinkage estimator for high-dimensional mean vector, *Journal of Multivariate Analysis* 170, 63–79.

Bodnar, T., Y. Okhrin, and N. Parolya, 2022, Optimal shrinkage-based portfolio selection in high dimensions, *Journal of Business & Economic Statistics* 41, 140–156.

Borji, A., 2019, Pros and cons of GAN evaluation measures, *Computer Vision and Image Understanding* 179, 41–65.

Brophy, E., Z. Wang, Q. She, and T. Ward, 2023, Generative adversarial networks in time series: A systematic literature review, *ACM Computing Surveys* 55, 1–31.

Bryzgalova, S., V. DeMiguel, S. Li, and M. Pelger, 2023, Asset-pricing factors with economic targets, *SSRN Electronic Journal* Working paper.

Büchner, M., and B. Kelly, 2022, A factor model for option returns, *Journal of Financial Economics* 143, 1140–1161.

Cao, H., C. Tan, Z. Gao, Y. Xu, G. Chen, P.-A. Heng, and S. Z. Li, 2024, A survey on generative diffusion models, *IEEE Transactions on Knowledge and Data Engineering* .

Carhart, M. M., 1997, On persistence in mutual fund performance, *The Journal of Finance* 52, 57–82.

Cetingoz, A. R., and C.-A. Lehalle, 2025, Synthetic data for portfolios: A throw of the dice will never abolish chance, *arXiv preprint arXiv:2501.03993* .

Chamberlain, G., and M. Rothschild, 1983, Arbitrage, factor structure, and mean–variance analysis on large asset markets, *Econometrica* 51, 1281–1304.

Chazottes, J.-R., P. Collet, and F. Redig, 2021, Evolution of gaussian concentration bounds under diffusions, *Markov Processes and Related Fields* 27, 707–754.

Chen, L., M. Pelger, and J. Zhu, 2024, Deep learning in asset pricing, *Management Science* 70, 714–750.

Chen, M., Z. Xu, K. Q. Weinberger, and F. Sha, 2012, Marginalized denoising autoencoders for domain adaptation, in *International Conference on Machine Learning*.

Chen, M., K. Huang, T. Zhao, and M. Wang, 2023a, Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data, in *International Conference on Machine Learning*, 4672–4712, PMLR.

Chen, M., H. Jiang, W. Liao, and T. Zhao, 2022, Nonparametric regression on low-dimensional manifolds using deep relu networks: Function approximation and statistical recovery, *Information and Inference: A Journal of the IMA* 11, 1203–1253.

Chen, M., X. Li, and T. Zhao, 2020, On generalization bounds of a family of recurrent neural networks, in *International Conference on Artificial Intelligence and Statistics*, volume 108, 1233–1243 (PMLR).

Chen, M., W. Liao, H. Zha, and T. Zhao, 2020, Statistical guarantees of generative adversarial networks for distribution estimation, *arXiv preprint arXiv:2002.03938* 9.

Chen, M., S. Mei, J. Fan, and M. Wang, 2024, Opportunities and challenges of diffusion models for generative ai, *National Science Review* 11, nwae348.

Chen, N.-F., R. Roll, and S. A. Ross, 1986, Economic forces and the stock market, *Journal of Business* 383–403.

Chen, S., S. Chewi, J. Li, Y. Li, A. Salim, and A. Zhang, 2023b, Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions, in *International Conference on Learning Representations*.

Chen, S., G. Daras, and A. Dimakis, 2023, Restoration-degradation beyond linear diffusions: A non-asymptotic analysis for ddim-type samplers, in *International Conference on Machine Learning*, 4462–4484, PMLR.

Cole, F., and Y. Lu, 2024, Score-based generative models break the curse of dimensionality in learning a family of sub-gaussian probability distributions, in *International Conference on Learning Representations*.

Coletta, A., S. Gopalakrishnan, D. Borrajo, and S. Vyetrenko, 2024, On the constrained time-series generation problem, *Advances in Neural Information Processing Systems* 36.

Coletta, A., J. Jerome, R. Savani, and S. Vyetrenko, 2023, Conditional generators for limit order book environments: Explainability, challenges, and robustness, in *ACM International Conference on AI in Finance*, 27–35.

Connor, G., M. Hagmann, and O. Linton, 2012, Efficient semiparametric estimation of the fama–french model and extensions, *Econometrica* 80, 713–754.

Cont, R., 2001, Empirical properties of asset returns: Stylized facts and statistical issues, *Quantitative Finance* 1, 223–236.

Cont, R., M. Cucuringu, J. Kochems, and F. Prenzel, 2023, Limit order book simulation with generative adversarial networks, *SSRN Electronic Journal* Working paper.

Cont, R., M. Cucuringu, R. Xu, and C. Zhang, 2022, Tail-GAN: Learning to simulate tail risk scenarios, *arXiv preprint arXiv:2203.01664* .

Creswell, A., T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, 2018, Generative adversarial networks: An overview, *IEEE Signal Processing Magazine* 35, 53–65.

Dai, B., Z. Wang, and D. Wipf, 2020, The usual suspects? reassessing blame for vae posterior collapse, in *International Conference on Machine Learning*, 2313–2322 (PMLR).

Davis, C., and W. M. Kahan, 1970, The rotation of eigenvectors by a perturbation. iii, *SIAM Journal on Numerical Analysis* 7, 1–46.

De Bortoli, V., 2022, Convergence of denoising diffusion models under the manifold hypothesis, *arXiv preprint arXiv:2208.05314* .

De Bortoli, V., J. Thornton, J. Heng, and A. Doucet, 2021, Diffusion schrödinger bridge with applications to score-based generative modeling, *Advances in Neural Information Processing Systems* 34, 17695–17709.

DeMiguel, V., L. Garlappi, F. J. Nogales, and R. Uppal, 2009, A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms, *Management Science* 55, 798–812.

DeMiguel, V., L. Garlappi, and R. Uppal, 2009, Optimal versus naive diversification: How inefficient is the 1/n portfolio strategy?, *The Review of Financial Studies* 22, 1915–1953.

Dhariwal, P., and A. Nichol, 2021, Diffusion models beat gans on image synthesis, *Advances in Neural Information Processing Systems* 34, 8780–8794.

Dou, Z., S. Kotekal, Z. Xu, and H. H. Zhou, 2024, From optimal score matching to optimal sampling, *arXiv preprint arXiv:2409.07032* .

Eckerli, F., and J. Osterrieder, 2021, Generative adversarial networks in finance: an overview, *arXiv preprint arXiv:2106.06364* .

Elkamhi, R., C. Jo, and Y. Nozawa, 2024, A one-factor model of corporate bond premia, *Management Science* 70, 1875–1900.

Fabozzi, F. J., D. Huang, and G. Zhou, 2010, Robust portfolios: contributions from operations research and finance, *Annals of Operations Research* 176, 191–220.

Fama, E. F., and K. R. French, 1993, Common risk factors in the returns on stocks and bonds, *Journal of Financial Economics* 33, 3–56.

Fama, E. F., and K. R. French, 2004, The capital asset pricing model: Theory and evidence, *Journal of Economic Perspectives* 18, 25–46.

Fama, E. F., and K. R. French, 2015a, A five-factor asset pricing model, *Journal of Financial Economics* 116, 1–22.

Fama, E. F., and K. R. French, 2015b, A five-factor asset pricing model, *Journal of Financial Economics* 116, 1–22.

Fan, J., J. Guo, and S. Zheng, 2022, Estimating number of factors by adjusted eigenvalues thresholding, *Journal of the American Statistical Association* 117, 852–861.

Fan, J., Y. Liao, and H. Liu, 2016, An overview of the estimation of large covariance and precision matrices, *The Econometrics Journal* 19, C1–C32.

Fan, J., Y. Liao, and M. Mincheva, 2013, Large covariance estimation by thresholding principal orthogonal complements, *Journal of the Royal Statistical Society Series B: Statistical Methodology* 75, 603–680.

Fan, J., Y. Liao, and W. Wang, 2016, Projected principal component analysis in factor models, *Annals of Statistics* 44, 219.

Fan, J., and Q. Yao, 2003, *Nonlinear time series: nonparametric and parametric methods* (Springer).

Federal Reserve Board, 2023, 2023 stress test scenarios, https://www.federalreserve.gov/publications/2023-stress-test-scenarios.htm.

Feng, G., S. Giglio, and D. Xiu, 2020, Taming the factor zoo: A test of new factors, *The Journal of Finance* 75, 1327–1370.

Feng, G., J. He, N. G. Polson, and J. Xu, 2024, Deep learning in characteristics-sorted factor models, *Journal of Financial and Quantitative Analysis* 59, 3001–3036.

Feng, G., L. Jiang, J. Li, and Y. Song, 2023, Deep tangency portfolio, *Available at SSRN 3971274* .

Ferson, W. E., and C. R. Harvey, 1991, The variation of economic risk premiums, *Journal of Political Economy* 99, 385–415.

Frankle, J., and M. Carbin, 2019, The lottery ticket hypothesis: Finding sparse, trainable neural networks, in *International Conference on Learning Representations*.

Fu, H., Z. Yang, M. Wang, and M. Chen, 2024, Unveil conditional diffusion models with classifier-free guidance: A sharp statistical theory.

Gao, X., H. M. Nguyen, and L. Zhu, 2025, Wasserstein convergence guarantees for a general class of score-based generative models, *Journal of Machine Learning Research* 26, 1–54.

Gao, X., J. Zha, and X. Y. Zhou, 2024, Reward-directed score-based diffusion models via q-learning, *arXiv preprint arXiv:2409.04832* .

Giglio, S., B. Kelly, and D. Xiu, 2022, Factor models, machine learning, and asset pricing, *Annual Review of Financial Economics* 14, 337–368.

Giglio, S., Y. Liao, and D. Xiu, 2021, Thousands of alpha tests, *The Review of Financial Studies* 34, 3456–3496.

Giglio, S., and D. Xiu, 2021, Asset pricing with omitted factors, *Journal of Political Economy* 129, 1947–1990.

Giglio, S., D. Xiu, and D. Zhang, 2025, Test assets and weak factors, *The Journal of Finance* 80, 259–319.

Gotoh, J.-y., and A. Takeda, 2011, On the role of norm constraints in portfolio selection, *Computational Management Science* 8, 323–353.

Gouk, H., E. Frank, B. Pfahringer, and M. J. Cree, 2021, Regularisation of neural networks by enforcing lipschitz continuity, *Machine Learning* 110, 393–416.

Gu, S., B. Kelly, and D. Xiu, 2020, Empirical asset pricing via machine learning, *The Review of Financial Studies* 33, 2223–2273.

Gu, S., B. Kelly, and D. Xiu, 2021, Autoencoder asset pricing models, *Journal of Econometrics* 222, 429–450.

Gui, J., Z. Sun, Y. Wen, D. Tao, and J. Ye, 2021, A review on generative adversarial networks: Algorithms, theory, and applications, *IEEE Transactions on Knowledge and Data Engineering* 35, 3313–3332.

Guo, Z., J. Liu, Y. Wang, M. Chen, D. Wang, D. Xu, and J. Cheng, 2024, Diffusion models in bioinformatics and computational biology, *Nature Reviews Bioengineering* 2, 136–154.

Hambly, B., R. Xu, and H. Yang, 2023, Recent advances in reinforcement learning in finance, *Mathematical Finance* 33, 437–503.

Han, S., H. Mao, and W. J. Dally, 2016, Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, in *International Conference on Learning Representations*.

Han, Y., M. Razaviyayn, and R. Xu, 2024, Neural network-based score estimation in diffusion models: Optimization and generalization, in *International Conference on Learning Representations*.

Harvey, C. R., Y. Liu, and H. Zhu, 2016, ... and the cross-section of expected returns, *The Review of Financial Studies* 29, 5–68.

Haussmann, U. G., and E. Pardoux, 1986, Time reversal of diffusions, *The Annals of Probability* 1188–1205.

He, J., D. Spokoyny, G. Neubig, and T. Berg-Kirkpatrick, 2019, Lagging inference networks and posterior collapse in variational autoencoders, in *International Conference on Learning Representations*.

He, X. D., S. Kou, and X. Peng, 2022, Risk measures: robustness, elicitability, and backtesting, *Annual Review of Statistics and Its Application* 9, 141–166.

He, Z., B. Kelly, and A. Manela, 2017, Intermediary asset pricing: New evidence from many asset classes, *Journal of Financial Economics* 126, 1–35.

Ho, J., A. Jain, and P. Abbeel, 2020, Denoising diffusion probabilistic models, *Advances in Neural Information Processing Systems* 33, 6840–6851.

Hoefler, T., D. Alistarh, T. Ben-Nun, N. Dryden, and A. Peste, 2021, Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks, *Journal of Machine Learning Research* 22, 1–124.

Hoffman, M. D., and M. J. Johnson, 2016, Elbo surgery: yet another way to carve up the variational evidence lower bound, in *NIPS 2016 Workshop in Advances in Approximate Bayesian Inference*, volume 1.

Hou, K., C. Xue, and L. Zhang, 2015, Digesting anomalies: An investment approach, *The Review of Financial Studies* 28, 650–705.

Huang, D. Z., J. Huang, and Z. Lin, 2025, Convergence analysis of probability flow ODE for score-based generative models, *IEEE Transactions on Information Theory* 71, 4581–4601.

Hultin, H., H. Hult, A. Proutiere, S. Samama, and A. Tarighati, 2023, A generative model of a limit order book using recurrent neural networks, *Quantitative Finance* 23, 931–958.

Hyvärinen, A., and P. Dayan, 2005, Estimation of non-normalized statistical models by score matching., *Journal of Machine Learning Research* 6.

Jacquier, E., and N. Polson, 2011, Bayesian methods in finance, in *The Oxford Handbook of Bayesian Econometrics*, chapter 9, 439–512 (Oxford University Press).

Jagannathan, R., and T. Ma, 2003, Risk reduction in large portfolios: Why imposing the wrong constraints helps, *The Journal of Finance* 58, 1651–1683.

Jagannathan, R., and Z. Wang, 1996, The conditional capm and the cross-section of expected returns, *The Journal of Finance* 51, 3–53.

Jegadeesh, N., and S. Titman, 1993, Returns to buying winners and selling losers: Implications for stock market efficiency, *The Journal of Finance* 48, 65–91.

Jorion, P., 1986, Bayes-stein estimation for portfolio analysis, *Journal of Financial and Quantitative analysis* 21, 279–292.

Kan, R., and G. Zhou, 2007, Optimal portfolio choice with parameter uncertainty, *Journal of Financial and Quantitative Analysis* 42, 621–656.

Karatzas, I., and S. Shreve, 1991, *Brownian motion and stochastic calculus*, volume 113 (Springer Science & Business Media).

Karras, T., M. Aittala, T. Aila, and S. Laine, 2022, Elucidating the design space of diffusion-based generative models, *Advances in Neural Information Processing Systems* 35, 26565–26577.

Kelly, B., S. Malamud, and L. H. Pedersen, 2023, Principal portfolios, *The Journal of Finance* 78, 347–387.

Kelly, B., D. Palhares, and S. Pruitt, 2023, Modeling corporate bond returns, *The Journal of Finance* 78, 1967–2008.

Kelly, B., D. Xiu, et al., 2023, Financial machine learning, *Foundations and Trends® in Finance* 13, 205–363.

Kelly, B. T., S. Pruitt, and Y. Su, 2019, Characteristics are covariances: A unified model of risk and return, *Journal of Financial Economics* 134, 501–524.

Koehler, F., A. Heckett, and A. Risteski, 2023, Statistical efficiency of score matching: The view from isoperimetry, in *International Conference on Learning Representations*.

Kotelnikov, A., D. Baranchuk, I. Rubachev, and A. Babenko, 2023, Tabddpm: Modelling tabular data with diffusion models, in *International Conference on Machine Learning*, 17564–17579, PMLR.

Ledoit, O., and M. Wolf, 2003, Improved estimation of the covariance matrix of stock returns with an application to portfolio selection, *Journal of Empirical Finance* 10, 603–621.

Ledoit, O., and M. Wolf, 2004, A well-conditioned estimator for large-dimensional covariance matrices, *Journal of Multivariate Analysis* 88, 365–411.

Ledoit, O., and M. Wolf, 2022, The power of (non-) linear shrinking: A review and guide to covariance matrix estimation, *Journal of Financial Econometrics* 20, 187–218.

Lee, H., J. Lu, and Y. Tan, 2022, Convergence for score-based generative modeling with polynomial complexity, *Advances in Neural Information Processing Systems* 35, 22870–22882.

Lee, H., J. Lu, and Y. Tan, 2023, Convergence of score-based generative modeling for general data distributions, in *International Conference on Algorithmic Learning Theory*, 946–985, PMLR.

Lettau, M., and S. Ludvigson, 2001, Consumption, aggregate wealth, and expected stock returns, *The Journal of Finance* 56, 815–849.

Lettau, M., and M. Pelger, 2020a, Estimating latent asset-pricing factors, *Journal of Econometrics* 218, 1–31.

Lettau, M., and M. Pelger, 2020b, Factors that fit the time series and cross-section of stock returns, *The Review of Financial Studies* 33, 2274–2325.

Li, G., Y. Wei, Y. Chen, and Y. Chi, 2024a, Towards non-asymptotic convergence for diffusion-based generative models, in *International Conference on Learning Representations*.

Li, G., Y. Wei, Y. Chi, and Y. Chen, 2024b, A sharp convergence theory for the probability flow odes of diffusion models, *arXiv preprint arXiv:2408.02320* .

Li, R., Q. Di, and Q. Gu, 2025, Unified convergence analysis for score-based diffusion models with deterministic samplers, in *International Conference on Learning Representations*.

Li, X., Y. Dai, and Q. Qu, 2024, Understanding generalizability of diffusion models requires rethinking the hidden gaussian structure, *Advances in Neural Information Processing Systems* 37, 57499–57538.

Li, Z., Q. Huang, L. Yang, and M. van Leeuwen, 2025, Diffusion models for tabular data: Challenges, current progress, and future directions, *arXiv preprint arXiv:2502.17119* .

Liao, S., H. Ni, M. Sabate-Vidales, L. Szpruch, M. Wiese, and B. Xiao, 2024, Sig-wasserstein gans for conditional time series generation, *Mathematical Finance* 34, 622–670.

Liu, H., T. Zhu, N. Jia, J. He, and Z. Zheng, 2024, Learning to simulate from heavy-tailed distribution via diffusion model, *Available at SSRN 4975931* .

Liu, Y., A. Tsyvinski, and X. Wu, 2022, Common risk factors in cryptocurrency, *The Journal of Finance* 77, 1133–1177.

Locatello, F., S. Bauer, M. Lucic, G. Rätsch, S. Gelly, B. Schölkopf, and O. Bachem, 2019, Challenging common assumptions in the unsupervised learning of disentangled representations, in *International Conference on Machine Learning*, volume 97, 4114–4124.

Louizos, C., M. Welling, and D. P. Kingma, 2018, Learning sparse neural networks through l_0 regularization, in *International Conference on Learning Representations*.

Lyu, Z., X. Xu, C. Yang, D. Lin, and B. Dai, 2022, Accelerating diffusion models via early stop of the diffusion process, *arXiv preprint arXiv:2205.12524* .

Nagel, S., 2013, Empirical cross-sectional asset pricing, *Annual Review of Finance Economics* 5, 167–199.

Nichol, A. Q., and P. Dhariwal, 2021, Improved denoising diffusion probabilistic models, in *International Conference on Machine Learning*, 8162–8171, PMLR.

Oko, K., S. Akiyama, and T. Suzuki, 2023, Diffusion models are minimax optimal distribution estimators, in *International Conference on Machine Learning*, volume 202, 26517–26582 (PMLR).

Onatski, A., 2010, Determining the number of factors from empirical distribution of eigenvalues, *The Review of Economics and Statistics* 92, 1004–1016.

Pástor, L., and R. F. Stambaugh, 2003, Liquidity risk and expected stock returns, *Journal of Political Economy* 111, 642–685.

Raponi, V., C. Robotti, and P. Zaffaroni, 2020, Testing beta-pricing models using large cross-sections, *The Review of Financial Studies* 33, 2796–2842.

Reppen, A. M., and H. M. Soner, 2023, Deep empirical risk minimization in finance: Looking into the future, *Mathematical Finance* 33, 116–145.

Revuz, D., and M. Yor, 2013, *Continuous martingales and Brownian motion*, volume 293 (Springer Science & Business Media).

Ronneberger, O., P. Fischer, and T. Brox, 2015, U-net: Convolutional networks for biomedical image segmentation, in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241, Springer.

Ross, S. A., 2013, The arbitrage theory of capital asset pricing, in *Handbook of the fundamentals of financial decision making: Part I*, 11–30 (World Scientific).

Saatci, Y., and A. G. Wilson, 2017, Bayesian gan, *Advances in Neural Information Processing Systems* 30.

Schneider, T., P. E. Strahan, and J. Yang, 2023, Bank stress testing: Public interest or regulatory capture?, *Review of Finance* 27, 423–467.

Shapiro, J., and J. Zeng, 2024, Stress testing and bank lending, *The Review of Financial Studies* 37, 1265–1314.

Song, Y., and S. Ermon, 2019, Generative modeling by estimating gradients of the data distribution, *Advances in Neural Information Processing Systems* 32.

Song, Y., and S. Ermon, 2020, Improved techniques for training score-based generative models, *Advances in Neural Information Processing Systems* 33, 12438–12448.

Song, Y., S. Garg, J. Shi, and S. Ermon, 2020, Sliced score matching: A scalable approach to density and score estimation, in *Uncertainty in Artificial Intelligence*, 574–584, PMLR.

Song, Y., J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, 2021, Score-based generative modeling through stochastic differential equations, *International Conference on Learning Representations* .

Soudry, D., E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro, 2018, The implicit bias of gradient descent on separable data, *Journal of Machine Learning Research* 19, 1–57.

Srinivas, S., A. Subramanya, and R. V. Babu, 2017, Training sparse neural networks, in *Conference on Computer Vision and Pattern Recognition Workshops*, 455–462, IEEE.

Tang, R., and Y. Yang, 2024, Adaptivity of diffusion models to manifold structures, in *International Conference on Artificial Intelligence and Statistics*, 1648–1656, PMLR.

Tang, W., and H. Zhao, 2024, Contractive diffusion probabilistic models, *arXiv preprint arXiv:2401.13115* .

Tang, W., and H. Zhao, 2025, Score-based diffusion models via stochastic differential equations, *Statistics Surveys* 19, 28–64.

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

Thomas, M., and A. T. Joy, 2006, *Elements of information theory* (Wiley-Interscience).

Tsybakov, A. B., 2009, *Introduction to Nonparametric Estimation*, first edition (Springer).

Tu, J., and G. Zhou, 2010, Incorporating economic objectives into bayesian priors: Portfolio choice under parameter uncertainty, *Journal of Financial and Quantitative Analysis* 45, 959–986.

Tukey, J. W., 1962, The future of data analysis, in *Breakthroughs in Statistics: Methodology and Distribution*, 408–452 (Springer).

Vershynin, R., 2018, *High-dimensional probability: An introduction with applications in data science*, volume 47 (Cambridge university press).

Vincent, P., 2011, A connection between score matching and denoising autoencoders, *Neural Computation* 23, 1661–1674.

Vuletić, M., and R. Cont, 2025, VOLGAN: A generative model for arbitrage-free implied volatility surfaces, *Applied Mathematical Finance* 1–36.

Vuletić, M., F. Prenzel, and M. Cucuringu, 2024, Fin-GAN: Forecasting and classifying financial time series via generative adversarial networks, *Quantitative Finance* 24, 175–199.

Wainwright, M. J., 2019, *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48 (Cambridge university press).

Wang, P., H. Zhang, Z. Zhang, S. Chen, Y. Ma, and Q. Qu, 2024, Diffusion models learn low-dimensional distributions via subspace clustering, *arXiv preprint arXiv:2409.02426* .

Weitzner, D., M. Delbracio, P. Milanfar, and R. Giryes, 2025, The diffusion process as a correlation machine: Linear denoising insights, *Transactions on Machine Learning Research* .

Wibisono, A., Y. Wu, and K. Y. Yang, 2024, Optimal score estimation via empirical bayes smoothing, in *Conference on Learning Theory*, volume 247 (PMLR).

Xiao, Z., K. Kreis, and A. Vahdat, 2022, Tackling the generative learning trilemma with denoising diffusion gans, in *International Conference on Learning Representations*.

Yakovlev, K., and N. Puchkin, 2025, Generalization error bound for denoising score matching under relaxed manifold assumption, in *Conference on Learning Theory*, volume 291, 5824–5891 (PMLR).

Yang, K. Y., and A. Wibisono, 2022, Convergence in kl and rényi divergence of the unadjusted langevin algorithm using estimated score, in *NeurIPS 2022 Workshop on Score-Based Methods*.

Yang, L., Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M.-H. Yang, 2023, Diffusion models: A comprehensive survey of methods and applications, *ACM Computing Surveys* 56, 1–39.

Yarotsky, D., 2017, Error bounds for approximations with deep relu networks, *Neural Networks* 94, 103–114.

Yogo, M., 2006, A consumption-based explanation of expected stock returns, *The Journal of Finance* 61, 539–580.

Yoon, J., D. Jarrett, and M. Van der Schaar, 2019, Time-series generative adversarial networks, *Advances in Neural Information Processing Systems* 32.

Zhang, K., H. Yin, F. Liang, and J. Liu, 2024, Minimax optimality of score-based diffusion models: Beyond the density lower bound assumptions, in *International Conference on Machine Learning*, volume 235, 60134–60178 (PMLR).

Zhou, X. Y., and D. Li, 2000, Continuous-time mean-variance portfolio selection: A stochastic LQ framework, *Applied Mathematics and Optimization* 42, 19–33.

A Omitted Proof in Section 3

In this section, we provide the formal proof of Lemma 1.

Proof of Lemma 1. By definition, the marginal distribution of $\mathbf{R}_t$ is

$$p_t(\mathbf{r}) = \int \underbrace{\phi(\mathbf{r}; \alpha_t \mathbf{r}_0, h_t \mathbf{I}_d)}_{\text{Gaussian transition kernel}} p_{\text{data}}(\mathbf{r}_0) \mathrm{d}\mathbf{r}_0$$
$$\overset{(i)}{=} \int \phi(\mathbf{r}; \alpha_t(\boldsymbol{\beta}\mathbf{f} + \boldsymbol{\varepsilon}), h_t \mathbf{I}_d) p_{\text{fac}}(\mathbf{f}) \phi\big(\boldsymbol{\varepsilon}; \mathbf{0}, \mathrm{diag}\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_d^2\}\big) \mathrm{d}\mathbf{f} \mathrm{d}\boldsymbol{\varepsilon}.$$

Here, equality (i) invokes the factor model (8) to represent $\mathbf{r}_0$ and the independence between factor and noise.

Since $\boldsymbol{\varepsilon}$ is Gaussian with uncorrelated entries, we can simplify p_t as

$$\begin{aligned} p_t(\mathbf{r}) &= \int \frac{1}{(2\pi h_t)^{d/2}} \exp\left(-\frac{\|\mathbf{r} - \alpha_t(\boldsymbol{\beta}\mathbf{f} + \boldsymbol{\varepsilon})\|_2^2}{2h_t}\right) \prod_{i=1}^d \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{\varepsilon_i^2}{2\sigma_i^2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\varepsilon_i \mathrm{d}\mathbf{f} \\ &\overset{(i)}{=} \int \prod_{i=1}^d \frac{1}{\sqrt{2\pi(h_t + \sigma_i^2 \alpha_t^2)}} \exp\left(-\frac{([\mathbf{r} - \alpha_t \boldsymbol{\beta}\mathbf{f}]_i)^2}{2(h_t + \sigma_i^2 \alpha_t^2)}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f} \\ &= \int \frac{1}{\sqrt{(2\pi)^d \det(\boldsymbol{\Lambda}_t)}} \exp\left(-\frac{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta}\mathbf{f}\|_2^2}{2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f}, \end{aligned} \tag{A.1}$$

where (i) holds by completing the squares and integrating with respect to ε_i, and the last equality holds by applying the formula of $\boldsymbol{\Lambda}_t$ in (10).

Now we define orthogonal decomposition of the rescaled returns $\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}$ into the subspace spanned by $\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}$ and its complement:

$$\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r} = (\mathbf{I} - \mathbf{T}_t)\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r} + \mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r},$$

where $\boldsymbol{\Gamma}_t$ and $\mathbf{T}_t$ are defined in (11), respectively. Along with the fact that $\mathbf{T}_t(\mathbf{I} - \mathbf{T}_t) = \mathbf{0}$, we can rewrite $p_t(\mathbf{r})$ in (A.1) as

$$p_t(\mathbf{r}) = \frac{(2\pi)^{-\frac{d}{2}}}{\sqrt{\det(\boldsymbol{\Lambda}_t)}} \exp\left(-\frac{\|(\mathbf{I} - \mathbf{T}_t)\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}\|_2^2}{2}\right) \int \exp\left(-\frac{\|\mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2^2}{2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f}. \tag{A.2}$$

Take the take gradient of $\log p_t$ with respect to $\mathbf{r}$ using expression in (A.2), we obtain:

$$\nabla \log p_t(\mathbf{r}) = -\boldsymbol{\Lambda}_t^{-\frac{1}{2}} (\mathbf{I} - \mathbf{T}_t)^2 \boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}$$

$$
\begin{aligned}
&- \frac{\int \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{T}_t (\mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \mathbf{f}) \exp\left(-\frac{\|\mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \mathbf{f}\|_2^2}{2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f}}{\int \exp\left(-\frac{\|\mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \mathbf{f}\|_2^2}{2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f}} \\
&\overset{(i)}{=} -\boldsymbol{\Lambda}_t^{-\frac{1}{2}} (\mathbf{I} - \mathbf{T}_t) \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} \\
&\quad - \frac{\int \boldsymbol{\Lambda}_t^{-1} \boldsymbol{\beta} (\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{\frac{1}{2}} \mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \mathbf{f}) \exp\left(-\frac{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} (\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{\frac{1}{2}} \mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \mathbf{f})\|_2^2}{2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f}}{\int \exp\left(-\frac{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} (\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{\frac{1}{2}} \mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \mathbf{f})\|_2^2}{2}\right) p_{\text{fac}}(\mathbf{f}) \mathrm{d}\mathbf{f}} \\
&\overset{(ii)}{=} - \underbrace{\boldsymbol{\Lambda}_t^{-\frac{1}{2}} (\mathbf{I} - \mathbf{T}_t) \cdot \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r}}_{\mathbf{s}_{\text{comp}}(\mathbf{r},t)\text{: Complement score}} + \underbrace{\mathbf{T}_t \boldsymbol{\Lambda}_t^{\frac{1}{2}} \boldsymbol{\beta} \cdot \nabla \log p_t^{\text{fac}}(\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{\frac{1}{2}} \mathbf{T}_t \cdot \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r})}_{\mathbf{s}_{\text{sub}}(\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{\frac{1}{2}} \mathbf{T}_t \cdot \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r}, t)\text{: Subspace score}},
\end{aligned}
$$

where (i) holds due to the fact that $(\mathbf{I}-\mathbf{T}_t)^2 = \mathbf{I}-\mathbf{T}_t$ and the following straightforward calculation by invoking the formula $\mathbf{T}_t$ in (11) and $\boldsymbol{\beta}^\top \boldsymbol{\beta} = \mathbf{I}_k$:

$$
\mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \mathbf{f} = \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \boldsymbol{\Gamma}_t \boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \cdot \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \mathbf{f} = \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \boldsymbol{\beta} \left(\boldsymbol{\beta}^\top \boldsymbol{\Lambda}_t^{\frac{1}{2}} \mathbf{T}_t \boldsymbol{\Lambda}_t^{-\frac{1}{2}} \mathbf{r} - \alpha_t \mathbf{f}\right).
$$

In addition, (ii) follows the definition of p_t^{fac}. □

B Omitted Proofs in Section 4

In this section, we provide proofs of Theorem 1, Theorem 2 and the lemmas used in the proof.

B.1 Proof of Theorem 1

Proof. Given the neural network architecture defined in (15), our goal is to construct a diagonal matrix $\mathbf{D}_t = \operatorname{diag}\left\{1/(h_t + \alpha_t^2 c_1), \ldots, 1/(h_t + \alpha_t^2 c_d)\right\} \in \mathbb{R}^d$ induced by a vector $\mathbf{c} = (c_1, \ldots, c_d) \in \mathbb{R}^d$, a matrix $\mathbf{V} \in \mathbb{R}^{d \times k}$ with orthonormal columns, and a ReLU network $\mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{V}^\top \mathbf{D}_t \mathbf{r}, t) \in \mathcal{S}_{\text{ReLU}}$ so that $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{r}, t)$ serves as a good approximator to $\nabla \log p_t(\mathbf{r})$. Thanks to the score decomposition in (16), we choose $\mathbf{D}_t(\sigma_1, \ldots, \sigma_d) = \operatorname{diag}\{1/(h_t + \sigma_1^2 \alpha_t^2), \ldots, 1/(h_t + \sigma_d^2 \alpha_t^2)\}$ and $\mathbf{V} = \boldsymbol{\beta}$. It remains to choose neural network hyper-parameters to guarantee the desired approximation power.

Step 1: Approximation on $\mathcal{C} \times [0, T]$. Define $\mathcal{C} = \left\{\mathbf{z} \in \mathbb{R}^k \middle| \|\mathbf{z}\|_2 \leq S\right\}$ as a k-dimensional ball of radius $S > 0$, with the choice of

$$
S = \mathcal{O}(\sqrt{(1 + \sigma_{\max}^2)(k + \log(1/\epsilon))}). \tag{B.1}
$$

On $\mathcal{C} \times [0, T]$, we approximate the coordinate ξ_i separately for each $i = 1, \ldots, d$. To ease the

analysis, we define the following linear transformation:

$$\boldsymbol{\xi}'(\mathbf{y}', t') := \boldsymbol{\xi}(\mathbf{z}, t) \text{ with } \mathbf{y}' := (\mathbf{z} + S\mathbf{1})/(2S) \text{ and } t' := t/T \tag{B.2}$$

such that the domain of $\mathcal{C} \times [0, T]$ is transformed to be contained within $[0, 1]^k \times [0, 1]$. Therefore, we can equivalently approximate ξ_i' for each $i = 1, \ldots, d$ on the new domain $[0, 1]^k \times [0, 1]$.

Recall that the subspace score $\mathbf{s}_{\text{sub}}(\mathbf{z}, t)$ is L_s-Lipschitz in $\mathbf{z}$ by Assumption 3. Then, by the definition of $\mathbf{s}_{\text{sub}}$ and $\boldsymbol{\xi}$ in (13) and (17), we derive that $\boldsymbol{\xi}(\mathbf{z}, t)$ is $2(1 + L_s)(1 + \sigma_{\max}^4)$-Lipschitz in $\mathbf{z}$. Immediately, we obtain that $\boldsymbol{\xi}'(\mathbf{y}', t')$ is $4S(1+L_s)(1+\sigma_{\max}^4)$-Lipschitz in $\mathbf{y}'$; so is each coordinate ξ_i. Denote $L_z = 2(1 + L_s)(1 + \sigma_{\max}^4)$.

Next, define

$$\tau(S) := \sup_{t \in [0,T]} \sup_{\mathbf{z} \in \mathcal{C}} \left\| \frac{\partial}{\partial t} \boldsymbol{\xi}(\mathbf{z}, t) \right\|_2 . \tag{B.3}$$

Then for any $\mathbf{y}' \in [0, 1]^k$, the Lipschitz constant of $\boldsymbol{\xi}'(\mathbf{y}', t')$ with respect to t' is bounded by $T\tau(S)$. Substituting the order of S in (B.1) into the upper bound of $\tau(S)$ in (B.32) in Lemma B.1, we have

$$\tau(S) = \mathcal{O}(L_s(1 + \sigma_{\max}^7) \operatorname{poly} k^{3/2} \log^{3/2}(1/\epsilon)). \tag{B.4}$$

For notation simplicity, we abbreviate $\tau(S)$ as τ when there is no confusion.

Now we construct a partition of the product space $[0, 1]^k \times [0, 1]$. For the hypercube $[0, 1]^k$, we partition it uniformly into smaller, non-overlapping hypercubes, each with an edge length of e_1. Similarly, we partition the interval $[0, 1]$ into non-overlapping subintervals, each of length e_2. Here, we take

$$e_1 = \mathcal{O}\left(\frac{\epsilon}{SL_z}\right) \quad \text{and} \quad e_2 = \mathcal{O}\left(\frac{\epsilon}{T\tau}\right).$$

In addition, we denote $N_1 = \left\lceil \frac{1}{e_1} \right\rceil, N_2 = \left\lceil \frac{1}{e_2} \right\rceil$.

Let $\mathbf{m} = [m_1, \ldots, m_k]^\top \in \{0, \ldots, N_1 - 1\}^k$ be a multi-index. We define a function $\bar{g}' : \mathbb{R}^{k+1} \mapsto \mathbb{R}^k$, with the i-th component g_i' being

$$\bar{g}_i'(\mathbf{y}', t') = \sum_{\mathbf{m}, j = 0, \ldots, N_2 - 1} \xi_i'\left(\frac{\mathbf{m}}{N_1}, \frac{j}{N_2}\right) \Psi_{\mathbf{m},j}(\mathbf{y}', t') . \tag{B.5}$$

Here $\Psi_{\mathbf{m},j}(\mathbf{y}', t')$ is a partition of unity function. Specifically, we choose $\Psi_{\mathbf{m},j}$ as a product of coordinate-wise trapezoid functions:

$$\Psi_{\mathbf{m},j}(\mathbf{y}', t') = \psi\left(3N_2\left(t' - \frac{j}{N_2}\right)\right) \prod_{i=1}^{d} \psi\left(3N_1\left(y_i' - \frac{m_i}{N_1}\right)\right),$$

where ψ is a one-demensional trapezoid function with the specific formula:

$$\psi(a) = \begin{cases} 1, & |a| < 1 \\ 2 - |a|, & |a| \in [1, 2] \\ 0, & |a| > 2 \end{cases}.$$

For any $1 \leq i \leq k$, we claim that

1. $\bar{g}'_i$ defined in (B.5) can approximate ξ_i arbitrarily well as long as N_1 and N_2 are sufficiently large;

2. $\bar{g}'_i$ can be well approximated by a ReLU neural network $\bar{g}'_{\zeta,i}$ with a controllable error.

The above two claims can be verified using Lemma 10 in Chen et al. (2020), in which we substitute the Lipschitz coefficients $4S(1 + L_s)(1 + \sigma_{\max}^4)$ and $T\tau$ of $\boldsymbol{\xi}'$ into the error analysis. Specifically, for any $1 \leq i \leq k$, we consider the ReLU neural network $\bar{g}'_{\zeta,i}$ that satisfies the following Lipschitz property:

$$\begin{aligned} &\left\|\bar{g}'_{\zeta,i}\left(\mathbf{y}'_1, t'\right) - \bar{g}'_{\zeta,i}\left(\mathbf{y}'_2, t'\right)\right\|_\infty \leq 10kSL_z \left\|\mathbf{y}'_1 - \mathbf{y}'_1\right\|_2, \quad \forall\, \mathbf{y}'_1, \mathbf{y}'_1 \in [0,1]^k, t' \in [0,1], \text{ and} \\ &\left\|\bar{g}'_{\zeta,i}\left(\mathbf{y}', t'_1\right) - \bar{g}'_{\zeta,i}\left(\mathbf{y}', t'_2\right)\right\|_\infty \leq 10T\tau \left\|t_1 - t_2\right\|_2, \forall\, t'_1, t'_2 \in [0,1], \mathbf{y}' \in [0,1]^k. \end{aligned}$$

By concatenating $\bar{g}'_{\zeta,i}$'s together, we construct $\overline{\mathbf{g}}_\zeta = [\bar{g}_{\zeta,1}, \ldots, \bar{g}_{\zeta,k}]^\top$. For a given error level $\epsilon > 0$, with a neural network configuration

$$\begin{aligned} M &= \mathcal{O}\left(T\tau(L_s+1)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon}+k\right)^{\frac{k}{2}}\right), \gamma_1 = 20k(1+L_s)(1+\sigma_{\max}^4), \\ L &= \mathcal{O}\left(\log\frac{1}{\epsilon}+k\right),\ J = \mathcal{O}\left(T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon}+k\right)^{\frac{k+2}{2}}\right), \gamma_2 = 10\tau, \\ K &= \mathcal{O}\left((1+L_s)(1+\sigma_{\max}^4)\left(\log\frac{1}{\epsilon}+k\right)^{\frac{1}{2}}\right), \kappa = \max\left\{(1+L_s)(1+\sigma_{\max}^4)\left(\log\frac{1}{\epsilon}+k\right)^{\frac{1}{2}}, T\tau\right\}, \end{aligned}$$

we have

$$\sup_{(\mathbf{y}',t')\in[0,1]^k\times[0,1]} \left\|\overline{\mathbf{g}}'_\zeta(\mathbf{y}', t') - \boldsymbol{\xi}'(\mathbf{y}', t')\right\|_\infty \leq \epsilon.$$

To transform the function $\overline{\mathbf{g}}'_\zeta$ back to domain $\mathcal{C} \times (0, T]$, we define

$$\overline{\mathbf{g}}_\zeta(\mathbf{z}, t) := \overline{\mathbf{g}}'_\zeta(\mathbf{y}', t')\mathbb{1}\{\|\mathbf{z}\|_2 \leq S\}. \tag{B.6}$$

By the definition of $\boldsymbol{\xi}'$ in (B.2), we deduce that

$$\sup_{(\mathbf{z},t)\in\mathcal{C}\times[0,T]} \left\|\overline{\mathbf{g}}_\zeta(\mathbf{z}, t) - \boldsymbol{\xi}(\mathbf{z}, t)\right\|_\infty \leq \epsilon. \tag{B.7}$$

Also by the variable transformation in (B.2), we obtain that $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$ is Lipschitz continuous in $\mathbf{z}$ and t. Specifically, for any $\mathbf{z}_1, \mathbf{z}_2 \in \mathcal{C}$ and $t \in [0, T]$, it holds that

$$\left\|\overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{z}_1, t\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{z}_2, t\right)\right\|_{\infty} \leq 10kL_z \left\|\mathbf{z}_1 - \mathbf{z}_2\right\|_2 .$$

In addition, for any $t_1, t_2 \in [0, T]$ and $\mathbf{z} \in \mathcal{C}$, it holds that

$$\left\|\overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{z}, t_1\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{z}, t_2\right)\right\|_{\infty} \leq 10\tau|t_1 - t_2|.$$

By definition of $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$ in (B.6), we have $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{z}, t\right) = \mathbf{0}$ for $\|\mathbf{z}\|_2 > S$. Therefore, the Lipschitz continuity property in $\mathbf{z}$ can be extended to $\mathbb{R}^k$.

Step 2: Bounding L^2 Approximation Error. Denote $\mathbf{Z} = \boldsymbol{\beta}^{\top}\boldsymbol{\Lambda}_t^{-1}\mathbf{R}_t$ with the distribution P_t^{fac}. The L^2 approximation error of $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$ can be decomposed into two terms

$$\begin{aligned}\left\|\boldsymbol{\xi}\left(\mathbf{Z}, t\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{Z}, t\right)\right\|_{L^2\left(P_t^{\mathrm{fac}}\right)} = &\left\|\left(\boldsymbol{\xi}\left(\mathbf{Z}, t\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{Z}, t\right)\right) \mathbb{1}\{\|\mathbf{Z}\|_2 < S\}\right\|_{L^2(P_t^{\mathrm{fac}})} \\ &+ \left\|\boldsymbol{\xi}\left(\mathbf{Z}, t\right) \mathbb{1}\{\|\mathbf{Z}\|_2 > S\}\right\|_{L^2\left(P_t^{\mathrm{fac}}\right)} .\end{aligned} \tag{B.8}$$

By applying the L^{∞} approximation error bound in (B.7), the first term in (B.8) is bounded by

$$\left\|\left(\boldsymbol{\xi}\left(\mathbf{Z}, t\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{Z}, t\right)\right) \mathbb{1}\{\|\mathbf{Z}\|_2 < S\}\right\|_{L^2(P_t^{\mathrm{fac}})} \leq \sqrt{k} \sup_{(\mathbf{z},t)\in\mathcal{C}\times[0,T]} \left\|\left(\boldsymbol{\xi}\left(\mathbf{z}, t\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{z}, t\right)\right)\right\|_{\infty} \leq \sqrt{k}\epsilon. \tag{B.9}$$

The second term on the right-hand side of (B.8) is controlled by the upper bound (B.38) in Lemma B.2. Specifically, by choosing $S = \mathcal{O}(\sqrt{(1+\sigma_{\max}^2)(k + \log(1/\epsilon))})$, we have

$$\left\|\boldsymbol{\xi}\left(\mathbf{Z}, t\right) \mathbb{1}\{\|\mathbf{Z}\|_2 > S\}\right\|_{L^2\left(P_t^{\mathrm{fac}}\right)} \leq \epsilon. \tag{B.10}$$

Combining (B.9) and (B.10), we deduce that

$$\left\|\boldsymbol{\xi}\left(\mathbf{Z}, t\right) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}\left(\mathbf{Z}, t\right)\right\|_{L^2\left(P_t^{\mathrm{fac}}\right)} \leq (\sqrt{k} + 1)\epsilon. \tag{B.11}$$

Furthermore, by involving $\overline{\mathbf{g}}_{\boldsymbol{\zeta}}$, we construct the following approximator $\overline{\mathbf{s}}_{\boldsymbol{\theta}}$ for $\nabla \log p_t(\mathbf{r})$

$$\overline{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{r}, t) := \alpha_t \boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\overline{\mathbf{g}}_{\boldsymbol{\zeta}}(\boldsymbol{\beta}^{\top}\boldsymbol{\Lambda}_t^{-1}\mathbf{r}, t) - \boldsymbol{\Lambda}_t^{-1}\mathbf{r}. \tag{B.12}$$

Then, by applying the formula of $\nabla \log p_t$ and $\overline{\mathbf{s}}_{\boldsymbol{\theta}}$ in (16) and (B.12) respectively, we obtain that

$$\|\nabla \log p_t(\cdot) - \overline{\mathbf{s}}_{\boldsymbol{\zeta}}(\cdot, t)\|_{L^2(P_t)} = \left\|\alpha_t \boldsymbol{\Lambda}_t^{-1/2}\boldsymbol{\beta}(\boldsymbol{\xi}(\mathbf{Z}, t) - \overline{\mathbf{g}}_{\boldsymbol{\zeta}}(\mathbf{Z}, t))\right\|_{L^2\left(P_t^{\mathrm{fac}}\right)} \leq \frac{(\sqrt{k}+1)\epsilon}{\min\{\sigma_d^2, 1\}},$$

where the inequality invokes $\|\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\|_{\mathrm{op}} \leq 1/(h_t + \sigma_d^2\alpha_t^2) \leq 1/\min\{\sigma_d^2, 1\}$ and the error bound (B.11). □

B.2 Proof of Theorem 2

Proof.

Step 1: Error Decomposition. The proof is based on the following bias-variance decomposition on $\mathcal{L}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}})$. For any $a \in (0,1)$, we decompose $\mathcal{L}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}})$ as

$$\begin{aligned}\mathcal{L}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}) &= \mathcal{L}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}) - (1+a)\widehat{\mathcal{L}}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}) + (1+a)\widehat{\mathcal{L}}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}) \\ &\le \underbrace{\mathcal{L}^{\text{trunc}}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}) - (1+a)\widehat{\mathcal{L}}^{\text{trunc}}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}})}_{(A)} + \underbrace{\mathcal{L}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}) - \mathcal{L}^{\text{trunc}}(\widehat{\mathbf{s}}_{\boldsymbol{\theta}})}_{(B)} + (1+a)\underbrace{\inf_{\mathbf{s}_{\boldsymbol{\theta}} \in \mathcal{S}_{\text{NN}}} \widehat{\mathcal{L}}(\mathbf{s}_{\boldsymbol{\theta}})}_{(C)},\end{aligned}$$

where $\mathcal{L}^{\text{trunc}}$ is defined as

$$\mathcal{L}^{\text{trunc}}(\mathbf{s}_{\boldsymbol{\theta}}) := \int \ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) p_t(\mathbf{r}) \mathrm{d}\mathbf{r} \quad \text{with} \quad \ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) := \ell(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) \mathbb{1}\{\|\mathbf{r}\|_2 \le \rho\} \tag{B.13}$$

subject to some truncation radius ρ to be determined. In the sequel, we bound (A) – (C) separately. The term (A) is the statistical error due to finite samples, term (B) is the truncation error, term (C) reflects the approximation error of $\mathcal{S}_{\text{NN}}$.

Note that the introduction of the hyper-parameter $a > 0$ (to be determined) is to handle the bias by applying Lemma 15 of Chen et al. (2023a), which is a standard Bernstein-type concentration inequality for empirical processes over function classes (see, for example, Theorem 3.27, Theorem 4.10, and Chapter 5 in Wainwright (2019)). Conversely, setting $a = 0$ results in a convergence rate at $\mathcal{O}(n^{-1/2})$, as derived using only Hoeffding's concentration inequality.

Step 2: Bounding Term (A). We denote $\mathcal{G} := \left\{\ell^{\text{trunc}}(\cdot; \mathbf{s}_{\boldsymbol{\theta}}) : \mathbf{s}_{\boldsymbol{\theta}} \in \mathcal{S}_{\text{NN}}\right\}$ as the class of loss functions induced by the score network $\mathcal{S}_{\text{NN}}$. We first determine an upper bound on all functions in $\mathcal{G}$ by bounding $\sup_{\mathbf{s}_{\boldsymbol{\theta}} \in \mathcal{S}_{\text{NN}}} \sup_{\mathbf{r} \in \mathbb{R}^d} |\ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}})|$.

To start, we consider

$$\begin{aligned}\left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{r}', t) + \frac{(\mathbf{r}' - \alpha_t \mathbf{r})}{h_t}\right\|_2 &\le \|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{r}', t) + \mathbf{D}_t \mathbf{r}'\|_2 + \left\|\frac{(\mathbf{I} - h_t \mathbf{D}_t)\mathbf{r}'}{h_t}\right\|_2 + \left\|\frac{\alpha_t \mathbf{r}}{h_t}\right\|_2 \\ &\overset{(i)}{=} \alpha_t \|\mathbf{D}_t \mathbf{V} \mathbf{g}_{\boldsymbol{\theta}}(\mathbf{V}^\top \mathbf{D}_t \mathbf{r}', t)\|_2 + \frac{\|\mathbf{I} - h_t \mathbf{D}_t\|_2 \|\mathbf{r}'\|_2}{h_t} + \frac{\alpha_t \|\mathbf{r}\|_2}{h_t} \\ &= \mathcal{O}\left(\frac{K + \|\mathbf{r}'\|_2 + \|\mathbf{r}\|_2}{h_t}\right),\end{aligned} \tag{B.14}$$

where (i) holds by applying the formula of $\mathbf{s}_{\boldsymbol{\theta}}$ in (18) and (ii) follows from the facts that $\alpha_t^2 \le 1$, $\|\mathbf{D}_t\|_{\text{op}} \le 1/h_t$, $\|\mathbf{V}\|_{\text{op}} = 1$, and $\|\mathbf{g}_{\boldsymbol{\theta}}\|_2 \le K$.

By the definition of ℓ^{trunc} in (B.13), for any $\mathbf{s}_{\boldsymbol{\theta}} \in \mathcal{S}_{\text{NN}}$ we have $\ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) = 0$ if $\|\mathbf{r}\|_2 > \rho$. For any $\|\mathbf{r}\|_2 \le \rho$, we have

$$\ell^{\text{trunc}}(\mathbf{r}; \mathbf{s}_{\boldsymbol{\theta}}) = \frac{1}{T - t_0} \int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t | \mathbf{R}_0 = \mathbf{r}} \left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{R}_t, t) + \frac{\mathbf{R}_t - \alpha_t \mathbf{r}}{h_t}\right\|_2^2 \cdot \mathbb{1}\{\|\mathbf{r}\|_2 \le \rho\} \, \mathrm{d}t$$

$$
\begin{aligned}
&\overset{(i)}{=} \mathcal{O}\bigg(\frac{1}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\bigg(\frac{K^2+\|\mathbf{R}_t\|_2^2+\|\mathbf{r}\|_2^2}{h_t^2}\bigg)\cdot\mathbb{1}\left\{\|\mathbf{r}\|_2\le\rho\right\}\mathrm{d}t\bigg)\\
&\overset{(ii)}{=} \mathcal{O}\bigg(\frac{1}{T-t_0}\int_{t_0}^{T}\bigg(\frac{2\rho^2+K^2}{h_t^2}+\frac{d}{h_t}\bigg)\mathrm{d}t\bigg)\\
&= \mathcal{O}\bigg(\frac{\rho^2+K^2}{t_0\,(T-t_0)}+\frac{d}{T-t_0}\log\frac{T}{t_0}\bigg),
\end{aligned}
$$

where (i) holds by the uniform upper bound (B.14); (ii) holds by applying the facts that $(\mathbf{R}_t|\mathbf{R}_0=\mathbf{r})\sim\mathcal{N}(\alpha_t\mathbf{r},h_t\mathbf{I}_d)$ and $\|\mathbf{r}\|_2\le\rho$ and $\alpha_t^2\le 1$.

To bound term (A), it is essential to consider the covering number of $\mathcal{S}_{\mathrm{NN}}$, as it measures the approximation power of the neural network class. Take $\mathbf{s}_{\boldsymbol{\theta}_1}$ and $\mathbf{s}_{\boldsymbol{\theta}_2}$ such that

$$
\sup_{\|\mathbf{r}'\|_2\le 3\rho+\sqrt{d\log d},t\in[t_0,T]}\left\|\mathbf{s}_{\boldsymbol{\theta}_1}(\mathbf{r}',t)-\mathbf{s}_{\boldsymbol{\theta}_2}(\mathbf{r}',t)\right\|_2\le\iota,
$$

we then have

$$
\begin{aligned}
&\left\|\ell^{\mathrm{trunc}}\left(\cdot;\mathbf{s}_{\boldsymbol{\theta}_1}\right)-\ell^{\mathrm{trunc}}\left(\cdot;\mathbf{s}_{\boldsymbol{\theta}_2}\right)\right\|_\infty\\
&=\sup_{\|\mathbf{r}\|_2\le\rho}\frac{1}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\bigg[\left\|\mathbf{s}_{\boldsymbol{\theta}_1}\left(\mathbf{R}_t,t\right)-\mathbf{s}_{\boldsymbol{\theta}_2}\left(\mathbf{R}_t,t\right)\right\|_2\cdot\left\|\mathbf{s}_{\boldsymbol{\theta}_1}\left(\mathbf{R}_t,t\right)+\mathbf{s}_{\boldsymbol{\theta}_2}\left(\mathbf{R}_t,t\right)+\frac{2(\mathbf{R}_t-\alpha_t\mathbf{r})}{h_t}\right\|_2\bigg]\mathrm{d}t\\
&\overset{(i)}{\le}\sup_{\|\mathbf{r}\|_2\le\rho}\frac{1}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\bigg[\frac{2}{h_t}\left(K+\|\mathbf{R}_t\|_2+\|\mathbf{r}\|_2\right)\left\|\mathbf{s}_{\boldsymbol{\theta}_1}\left(\mathbf{R}_t,t\right)-\mathbf{s}_{\boldsymbol{\theta}_2}\left(\mathbf{R}_t,t\right)\right\|_2\\
&\qquad\cdot\mathbb{1}\left\{\|\mathbf{R}_t\|_2\le 3\rho+\sqrt{d\log d}\right\}\bigg]\,\mathrm{d}t\\
&\quad+\sup_{\|\mathbf{r}\|_2\le\rho}\frac{1}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\bigg[\frac{2}{h_t}\left(K+\|\mathbf{R}_t\|_2+\|\mathbf{r}\|_2\right)\left\|\mathbf{s}_{\boldsymbol{\theta}_1}\left(\mathbf{R}_t,t\right)-\mathbf{s}_{\boldsymbol{\theta}_2}\left(\mathbf{R}_t,t\right)\right\|_2\\
&\qquad\cdot\mathbb{1}\left\{\|\mathbf{R}_t\|_2> 3\rho+\sqrt{d\log d}\right\}\bigg]\,\mathrm{d}t\\
&\overset{(ii)}{\le}\sup_{\|\mathbf{r}\|_2\le\rho}\frac{2\iota}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\bigg[\frac{1}{h_t}\left(K+\|\mathbf{R}_t\|_2+\|\mathbf{r}\|_2\right)\cdot\mathbb{1}\left\{\|\mathbf{R}_t\|_2\le 3\rho+\sqrt{d\log d}\right\}\bigg]\,\mathrm{d}t\\
&\quad+\sup_{\|\mathbf{r}\|_2\le\rho}\frac{1}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\bigg[\frac{2}{h_t}\left(K+\|\mathbf{R}_t\|_2+\|\mathbf{r}\|_2\right)\left\|\mathbf{s}_{\boldsymbol{\theta}_1}\left(\mathbf{R}_t,t\right)-\mathbf{s}_{\boldsymbol{\theta}_2}\left(\mathbf{R}_t,t\right)\right\|_2\\
&\qquad\cdot\mathbb{1}\left\{\|\mathbf{R}_t\|_2> 3\rho+\sqrt{d\log d}\right\}\bigg]\,\mathrm{d}t,
\end{aligned}
$$

where (i) follows from applying the upper bound in (B.14) and decomposing the error into two parts: within the compact domain of radius $3\rho+\sqrt{d\log d}$, and outside this domain; (ii) holds since $\|\mathbf{s}_{\boldsymbol{\theta}_1}(\mathbf{r}',t)-\mathbf{s}_{\boldsymbol{\theta}_2}(\mathbf{r}',t)\|_2\le\iota$ in the compact domain $\|\mathbf{r}'\|\le 3\rho+\sqrt{d\log d}$. Then, we deduce that

$$
\begin{aligned}
&\|\ell^{\mathrm{trunc}}(\cdot;\mathbf{s}_{\boldsymbol{\theta}_1})-\ell^{\mathrm{trunc}}(\cdot;\mathbf{s}_{\boldsymbol{\theta}_2})\|_\infty\\
&\overset{(i)}{=}\mathcal{O}\bigg(\frac{2\iota}{T-t_0}\int_{t_0}^{T}\frac{K+\sqrt{h_t d}+2\rho}{h_t}\mathrm{d}t+\frac{2}{T-t_0}\int_{t_0}^{T}\frac{1}{h_t}\bigg(\rho K^2 h_t^{-\frac{d+4}{2}}\left(\frac{\rho}{d}\right)^d\exp\bigg(-\frac{\rho^2}{h_t}\bigg)\bigg)\mathrm{d}t\bigg)
\end{aligned}
$$

$$\overset{(ii)}{=} \mathcal{O}\bigg(\iota \cdot \frac{(\rho+K)\log(T/t_0)+\sqrt{d}(\sqrt{T}-\sqrt{t_0})}{T-t_0} + \rho K^2 \left(\frac{\rho}{d}\right)^{\frac{d}{2}} \exp\left(-\frac{\rho^2}{2h_T}\right)\bigg),$$

where (i) holds by applying $(\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}) \sim \mathcal{N}(\alpha_t\mathbf{r}, h_t\mathbf{I}_d)$, $\|\mathbf{r}\|_2 \le \rho$ and the upper bound (B.43) in Lemma B.3; (ii) follows from the facts that $h_t = \mathcal{O}(t)$ as $t \to 0$ and the second term in (i) has a dominating exponential decay rate $\exp(-\rho^2/h_t)$. For notational simplicity, we denote

$$\eta := \rho K^2 (\rho/d)^{d/2} \exp(-\rho^2/(2h_T)). \tag{B.15}$$

Denote the τ-covering number of a class of functions $\mathcal{H}$ under a metric $\Psi(\cdot)$ by

$$\mathfrak{N}(\tau, \mathcal{H}, \Psi(\cdot)) = \inf\{|\mathcal{H}_1| : \mathcal{H}_1 \subseteq \mathcal{H},\ \forall h \in \mathcal{H},\ \exists h_1 \in \mathcal{H},\ \text{s.t. } \Psi(h, h_1) \le \tau\}, \tag{B.16}$$

where $|\mathcal{H}_1|$ represents the number of functions in the class $\mathcal{H}_1$. Immediately, we can deduce that an ι-covering of $\mathcal{S}_{\text{NN}}$ induces a covering of $\mathcal{G}$ with an accuracy $\iota \cdot \frac{(\rho+K)\log(T/t_0)+\sqrt{d}(\sqrt{T}-\sqrt{t_0})}{T-t_0} + \eta$. To apply Bernstein-type concentration inequality (Chen et al. 2023a, Lemma 15) for $\ell^{\text{trunc}}(\cdot; \mathbf{s}_{\boldsymbol{\theta}})$, let us take $B = \mathcal{O}(\frac{\rho^2+K^2+t_0 d\log(T/t_0)}{t_0(T-t_0)})$, $\tau = \iota$ and the corresponding covering number of $\mathcal{S}_{\text{NN}}$ as

$$\mathfrak{N}\left(\frac{(\iota-\eta)(T-t_0)}{(\rho+K)\log(T/t_0)+\sqrt{d}(\sqrt{T}-\sqrt{t_0})}, \mathcal{S}_{\text{NN}}, \|\cdot\|_2\right)$$

Then by Lemma 15 of Chen et al. (2023a), with probability $1-\delta$, it holds that

$$(A) = \mathcal{O}\left(\frac{(1+\frac{3}{a})\big(\frac{\rho^2+K^2+t_0 d\log(\frac{T}{t_0})}{t_0(T-t_0)}\big)}{n} \log \frac{\mathfrak{N}\Big(\frac{(\iota-\eta)(T-t_0)}{(\rho+K)\log(\frac{T}{t_0})+\sqrt{d}(\sqrt{T}-\sqrt{t_0})}, \mathcal{S}_{\text{NN}}, \|\cdot\|_2\Big)}{\delta} + (2+a)\iota\right). \tag{B.17}$$

Step 3: Bounding Term (B). By applying the formulas of ℓ and ℓ^{trunc} in (7) and (B.13), we have

$$\begin{aligned}(B) &= \frac{1}{T-t_0}\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_0\sim P_{\text{data}}}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0}\left[\left\|\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t,t) + \frac{(\mathbf{R}_t-\alpha_t\mathbf{R}_0)}{h_t}\right\|_2^2 \mathbb{1}\{\|\mathbf{R}_0\|_2 > \rho\}\right]\mathrm{d}t \\ &\le \frac{2}{T-t_0}\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_0\sim P_{\text{data}}}\left[\frac{h_t d + K^2 + 2\|\mathbf{R}_0\|_2^2}{h_t^2}\mathbb{1}\{\|\mathbf{R}_0\|_2 > \rho\}\right]\mathrm{d}t,\end{aligned}$$

where the inequality follows from applying the upper bound (B.14) and $(\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}) \sim \mathcal{N}(\alpha_t\mathbf{r}, h_t\mathbf{I}_d)$. Notice that the density of $\mathbf{R}_0 = \boldsymbol{\beta}\mathbf{F} + \boldsymbol{\varepsilon}$ can be bounded by

$$\begin{aligned}p_{\text{data}}(\mathbf{r}) &= \prod_{i=1}^{d}\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{\varepsilon_i^2}{2\sigma_i^2}\right) p_{\text{fac}}(\mathbf{f}) \\ &\overset{(i)}{\le} \frac{(2\pi)^{-(d+k)/2}C_1}{\prod_{i=1}^{d}\sigma_i}\exp\left(-\frac{\sigma_{\max}^{-2}\|\boldsymbol{\epsilon}\|_2^2 + C_2\|\boldsymbol{\beta}\mathbf{f}\|_2^2}{2}\right)\end{aligned}$$

$$\leq \frac{C_1(2\pi)^{-(d+k)/2}}{\prod_{i=1}^d \sigma_i} \exp\left(-\frac{\|\mathbf{r}\|_2^2}{2(\sigma_{\max}^2 + 1/C_2)}\right), \tag{B.18}$$

where (i) follows from the sub-Gaussian tail (19) in Assumption 2 and the fact that $\boldsymbol{\beta}$ is a norm-preserving transformation satisfying $\boldsymbol{\beta}^\top \boldsymbol{\beta} = \mathbf{I}$ in Assumption 1. Therefore, by applying the upper bound of p_{data} in (B.18), we obtain

$$\begin{aligned}
(B) &\leq \frac{2}{T-t_0}\int_{t_0}^T \left[\left(\frac{h_t d + K^2 + 2\|\mathbf{r}\|^2}{h_t^2}\right)\frac{C_1(2\pi)^{-\frac{d+k}{2}}}{\prod_{i=1}^d \sigma_i}\exp\left(-\frac{\|\mathbf{r}\|_2^2}{2(\sigma_{\max}^2+1/C_2)}\right)\mathbb{1}\{\|\mathbf{r}\|_2 > \rho\}\right]\mathrm{d}t \\
&\overset{(i)}{\leq} \frac{C_1 d(\sigma_{\max}^2 + 1/C_2)2^{-(d+k)/2}}{(\prod_{i=1}^d \sigma_i)(T-t_0)\Gamma(d/2+1)}\exp\left(-\frac{\rho^2}{2(\sigma_{\max}^2+1/C_2)}\right)\int_{t_0}^T \frac{\rho^{d-1}(h_t d + K^2 + \rho^2)}{h_t^2}\mathrm{d}t \\
&= \mathcal{O}\left(\frac{d\rho^{d-1}2^{-(d+k)/2}(\sigma_{\max}^2+1/C_2)}{(\prod_{i=1}^d \sigma_i)(T-t_0)\Gamma(d/2+1)}\left(\frac{\rho^2+K^2}{t_0} + d\log\frac{T}{t_0}\right)\exp\left(-\frac{\rho^2}{2(\sigma_{\max}^2+1/C_2)}\right)\right),
\end{aligned} \tag{B.19}$$

where (i) follows from the tail estimation in Proposition 2.6.6 of Vershynin (2018).

Step 4: Bounding Term (C). Recall that $\bar{\mathbf{s}}_{\boldsymbol{\theta}}$ is the constructed network approximator in Theorem 1. For any $\epsilon > 0$, we have

$$(C) \leq \underbrace{\widehat{\mathcal{L}}(\bar{\mathbf{s}}_{\boldsymbol{\theta}}) - (1+a)\mathcal{L}^{\text{trunc}}(\bar{\mathbf{s}}_{\boldsymbol{\theta}})}_{(\spadesuit)} + (1+a)\underbrace{\mathcal{L}^{\text{trunc}}(\bar{\mathbf{s}}_{\boldsymbol{\theta}})}_{(\clubsuit)},$$

where $(\spadesuit)$ is the statistical error and $(\clubsuit)$ is the approximation error.

First, we can bound $(\spadesuit)$ with high probability using the fact that $\mathbf{R}_0$ has a sub-Gaussian tail. Specifically, applying Proposition 2.6.6 of Vershynin (2018) to $\mathbf{R}_0$ with density bound (B.18), we obtain

$$\mathbb{P}\left(\|\mathbf{R}_0\|_2 > \rho\right) \leq \frac{C_1 d(\sigma_{\max}^2+1/C_2)2^{-(d+k)/2}}{(T-t_0)\Gamma(d/2+1)\left(\prod_{i=1}^d \sigma_i\right)}\rho^{d-1}\exp\left(-\frac{\rho^2}{2(\sigma_{\max}^2+1/C_2)}\right) := q. \tag{B.20}$$

Applying union bound for n i.i.d. data samples $\{\mathbf{R}_0^i\}_{i=1}^n$ from P_{data} leads to

$$\mathbb{P}\left(\left\|\mathbf{R}_0^i\right\|_2 \leq \rho \text{ for all } i = 1, \ldots, n\right) \geq 1 - nq.$$

Immediately, we obtain that, with probability $1 - nq$, it holds

$$(\spadesuit) = \widehat{\mathcal{L}}^{\text{trunc}}(\bar{\mathbf{s}}_{\boldsymbol{\theta}}) - (1+a)\mathcal{L}^{\text{trunc}}(\bar{\mathbf{s}}_{\boldsymbol{\theta}}).$$

Meanwhile, Lemma 15 of Chen et al. (2023a) implies that with probability $1 - \delta$, it holds that

$$(\spadesuit) = \mathcal{O}\left(\frac{(1+6/a)}{n}\left(\frac{\rho^2+K^2}{t_0(T-t_0)} + \frac{d}{T-t_0}\log\frac{T}{t_0}\right)\log\frac{1}{\delta}\right). \tag{B.21}$$

Here, we take δ defined in (B.17). For (♣), we have

$$(\clubsuit) \le \mathcal{L}(\bar{\mathbf{s}}_{\boldsymbol{\theta}}) = \frac{1}{T-t_0}\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t\sim P_t} \|\bar{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t,t) - \nabla \log p_t(\mathbf{R}_t)\|_2^2 \,\mathrm{d}t \\ + \underbrace{\mathcal{L}\left(\bar{\mathbf{s}}_{\boldsymbol{\theta}}\right) - \frac{1}{T-t_0}\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t\sim P_t} \|\bar{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t,t) - \nabla \log p_t(\mathbf{R}_t)\|_2^2 \,\mathrm{d}t}_{(E)}. \tag{B.22}$$

Then, by Theorem 1, we have

$$(\clubsuit) = \mathcal{O}\left(\frac{k\epsilon^2}{t_0(T-t_0)}\right) + (E). \tag{B.23}$$

Note that the two terms in (E) are equivalent to the score-matching objectives (5) and (6), hence (E) is on a constant order.

Combining two error bounds for (♠) and (♣) in (B.21) and (B.23), we deduce that, with probability $(1-nq)(1-\delta)$, it holds

$$(C) = \mathcal{O}\left(\frac{(1+6/a)}{n}\left(\frac{\rho^2+K^2}{t_0\,(T-t_0)} + \frac{d}{T-t_0}\log\frac{T}{t_0}\right)\log\frac{1}{\delta} + \frac{(1+a)k\epsilon^2}{t_0(T-t_0)}\right) + (1+a)\cdot(E). \tag{B.24}$$

Step 5: Choosing ρ and putting together (A), (B) and (C). Under a fixed $\delta > 0$ in (B.17), we choose ρ and τ as the following to balance terms (A), (B), and (C),

$$\rho = \mathcal{O}\left(\sqrt{\sigma_1^2\left(d + \log K + \log(n/\delta)\right)}\right) \text{ and } \tau = \mathcal{O}\left(\frac{1}{nt_0(T-t_0)}\right). \tag{B.25}$$

By direct calculation, our choice of ρ implies that $q \le \delta/n$, where q is defined in (B.20). Next, we derive the error bound for terms (A)-(C) under our choice of the hyper-parameters.

1. For term (A), we first give an upper bound for η defined in (B.15). Substituting the order of ρ in (B.25) into (B.15), we deduce that

$$\eta = \mathcal{O}\left(\frac{1}{nt_0(T-t_0)}\right). \tag{B.26}$$

Then, substituting the order of K in (22) in Theorem 1 and the hyperparameters in (B.25)

and (B.26) into (B.17), we obtain that with probability $1-\delta$, it holds that

$$
\begin{aligned}
(A) = \mathcal{O}\Bigg(& \frac{(1+\sigma_{\max}^8)(1+3/a)\left((1+L_s)^2\left(\log\frac{1}{\epsilon}+k\right)^2+d+\log\frac{n}{\delta}\right)}{nt_0(T-t_0)} \\
& \cdot \log\frac{\mathfrak{N}\left(\frac{1}{nt_0\left(\rho+K+\sqrt{d}(\sqrt{T}-\sqrt{t_0})\right)}, \mathcal{S}_{\mathrm{NN}}, \|\cdot\|_2\right)}{\delta} + \frac{1}{nt_0(T-t_0)}\Bigg) \\
\overset{(i)}{=} \mathcal{O}\Bigg(& \frac{(1+\sigma_{\max}^8)(1+3/a)\left((L_s+1)^2(\log\frac{1}{\epsilon}+k)^2+d+\log\frac{n}{\delta}\right)}{nt_0(T-t_0)} \\
& \cdot \left(dk + T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon}+k\right)^{\frac{k+4}{2}}\right)\log\frac{T\tau dk}{t_0\iota\epsilon} + \frac{1}{nt_0(T-t_0)}\Bigg),
\end{aligned}
\tag{B.27}
$$

where (i) follows from applying the upper bound (B.46) for the covering number of $\mathcal{S}_{\mathrm{NN}}$ in Lemma B.4.

2. For term (B), by plugging the order of ρ and K, defined in (B.25) and (22), into (B.19) and by straightforward calculations, we have

$$
(B) = \mathcal{O}\left(\frac{1}{nt_0(T-t_0)}\right). \tag{B.28}
$$

3. For term (C), applying the fact that $q \le \frac{\delta}{n}$ and the order of ρ and K in (B.25) and (22) to (B.24), with probability $1-2\delta$, it holds

$$
\begin{aligned}
(C) = \mathcal{O}\Bigg(& \frac{(1+\sigma_{\max}^8)(1+6/a)\left((1+L_s)^2\left(\log\frac{1}{\epsilon}+k\right)^2+d+\log\frac{n}{\delta}\right)}{nt_0(T-t_0)}\log\frac{1}{\delta} \\
& + \frac{1}{nt_0(T-t_0)} + \frac{k\epsilon^2}{\min\{\sigma_d^4, 1\}}\Bigg) + (1+a)\cdot(E).
\end{aligned}
\tag{B.29}
$$

Summing up the error terms in (B.27)-(B.29), we derive that with probability $1-3\delta$, it holds that

$$
\begin{aligned}
\mathcal{L}\left(\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\right) &\le (A) + (B) + (1+a)\cdot(C) \\
&= \mathcal{O}\Bigg(\frac{(1+\sigma_{\max}^8)(1+6/a)\left((1+L_s)^2(\log\frac{1}{\epsilon}+k)^2+d+\log\frac{n}{\delta}\right)}{nt_0(T-t_0)} \\
&\quad \cdot \left(dk + T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon}+k\right)^{\frac{k+4}{2}}\right)\log\frac{T\tau dk}{t_0\iota\epsilon}\Bigg) \\
&\quad + \mathcal{O}\left(\frac{1}{nt_0(T-t_0)} + \frac{k\epsilon^2}{t_0(T-t_0)}\right) + (1+a)^2\cdot(E).
\end{aligned}
$$

By the definition of (E) in (B.22) and setting $a = \epsilon^2$, with probability $1 - 3\delta$, it holds that

$$\begin{aligned}
&\frac{1}{T-t_0}\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t\sim P_t} \|\bar{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t,t) - \nabla \log p_t(\mathbf{R}_t)\|_2^2 \,\mathrm{d}t \\
&= \mathcal{O}\Bigg(\frac{(1+\sigma_{\max}^8)\left((1+L_s)^2(\log\frac{1}{\epsilon}+k)^2 + d + \log\frac{n}{\delta}\right)}{\epsilon^2 n t_0 (T-t_0)} \\
&\quad \cdot \left(dk + T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon}+k\right)^{\frac{k+4}{2}}\right)\log\frac{T\tau dk}{t_0\iota\epsilon}\Bigg) \\
&\quad + \mathcal{O}\left(\frac{1}{nt_0(T-t_0)} + \frac{k\epsilon^2}{t_0(T-t_0)}\right) \\
&=: (E_1) + (E_2).
\end{aligned}$$

Step 6: Balancing Error Terms. Take $\delta = 1/(3n)$ such that with probability $1 - 1/n$, it holds that

$$\begin{aligned}
(E_1) &= \mathcal{O}\left(\frac{(1+\sigma_{\max}^{k+8})(1+L_s)^k(d^2\log d)(k^{\frac{k+7}{2}}\log k)(\tau\log\tau)T\epsilon^{-(k+3)}\log^{\frac{k+7}{2}}(\frac{1}{\epsilon})\log^3 n}{nt_0}\right) \\
&\overset{(i)}{=} \mathcal{O}\left(\frac{(1+\sigma_{\max}^{2k})(1+L_s)^k(d^{\frac{7}{2}}\log d)(k^{\frac{k+10}{2}}\log^{\frac{5}{2}} k)\epsilon^{-(k+3)}\log^{\frac{k+10}{2}}(\frac{1}{\epsilon})\log^{\frac{9}{2}} n}{nt_0}\right) \\
&\overset{(ii)}{=} \tilde{\mathcal{O}}\left(\frac{1}{n}\epsilon^{-(k+3)}\log^{\frac{k+10}{2}}(\frac{1}{\epsilon})\right)
\end{aligned}$$

and

$$(E_2) = \tilde{\mathcal{O}}\left(\frac{1}{n} + \epsilon^2\right). \tag{B.30}$$

Here (i) follows from invoking the upper bound of $\tau(S)$ in (B.4) and $\tilde{\mathcal{O}}(\cdot)$ in (i) holds by keeping terms only on the sample size n and the error term ϵ.

To balance two error terms (E_1) and (E_2), we choose ϵ as the following

$$\epsilon = n^{-\frac{1-\delta(n)}{k+5}} \text{ with } \delta(n) = \frac{(k+10)\log\log n}{2\log n}. \tag{B.31}$$

Consequently, we obtain

$$\begin{aligned}
\frac{1}{n}\epsilon^{-(k+3)}\log^{\frac{k+10}{2}}(1/\epsilon) &= n^{-1+\frac{(k+3)(1-\delta(n))}{k+5}}(1/\epsilon)^{\frac{(k+10)\log\log(1/\epsilon)}{2\log(1/\epsilon)}} \\
&= n^{-\frac{2-2\delta(n)}{k+5}} \cdot n^{-1+\left(1+\frac{(k+10)\log\log(1/\epsilon)}{2(k+5)\log(1/\epsilon)}\right)\left(1-\delta(n)\right)} \\
&\overset{(i)}{=} \mathcal{O}\left(n^{-\frac{2-2\delta(n)}{k+5}}\right) = \epsilon^2,
\end{aligned}$$

where (i) holds by the formula of $\delta(n)$ in (B.31). By straightforward calculations, we deduce that,

with probability $1-\frac{1}{n}$, it holds

$$\begin{aligned}
&\frac{1}{T-t_0}\int_{t_0}^{T}\mathbb{E}_{\mathbf{R}_t\sim P_t}\|\bar{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t,t)-\nabla\log p_t(\mathbf{R}_t)\|_2^2\,\mathrm{d}t\\
&=\mathcal{O}\bigg(\frac{1}{t_0}(1+\sigma_{\max}^{2k})(1+L_s)^k d^2 k^{\frac{k+10}{2}}\left(\sqrt{d}n^{-\frac{2-2\delta(n)}{k+5}}+n^{-\frac{k+3+2\delta(n)}{k+5}}\right)\log d\log^4 n\bigg)\\
&=\tilde{\mathcal{O}}\bigg(\frac{1}{t_0}(1+\sigma_{\max}^{2k})d^{\frac{5}{2}}k^{\frac{k+10}{2}}n^{-\frac{2-2\delta(n)}{k+5}}\log^4 n\bigg),
\end{aligned}$$

where the last equality follows from omitting terms associated with L_s and polynomial terms in $\log t_0$, $\log d$, and $\log k$. □

B.3 Supporting Lemmas and Proofs

Lemma B.1. *Under the same assumptions as in Theorem 1, it holds that*

$$\tau(S)=\mathcal{O}\bigg(L_s\,\mathrm{poly}(1+\sigma_{\max}^2)\,\mathrm{poly}(\sqrt{k}S)\bigg),\tag{B.32}$$

where $\tau(S)$ is defined in (B.3) and $\mathrm{poly}(\cdot)$ represents a cubic polynomial.

Proof of Lemma B.1.

Recall that $\tau(S)$ is associated with $\boldsymbol{\xi}(\mathbf{z},t)$, which is defined in (17). By direct calculation, we have

$$\begin{aligned}
\frac{\partial\boldsymbol{\xi}}{\partial t}&=-\frac{1}{2}\frac{\int\mathbf{f}\frac{\partial\|\boldsymbol{\Gamma}_t^{-\frac{1}{2}}(\mathbf{z}-\alpha_t\mathbf{f})\|_2^2}{\partial t}\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}}{\int\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}}+\frac{1}{2}\boldsymbol{\xi}\frac{\int\frac{\partial\|\boldsymbol{\Gamma}_t^{-\frac{1}{2}}(\mathbf{z}-\alpha_t\mathbf{f})\|_2^2}{\partial t}\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}}{\int\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\,\mathrm{d}\mathbf{f}}\\
&\overset{(i)}{=}\frac{\alpha_t^2}{2}\mathbb{E}[\mathbf{F}\mathbf{F}^\top\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-2}\boldsymbol{\beta}\mathbf{F}|\mathbf{Z}=\mathbf{z}]+\frac{\alpha_t}{2}\mathrm{Cov}[\mathbf{F}|\mathbf{Z}=\mathbf{z}]\mathbf{C}_t\mathbf{z}+\frac{\alpha_t^2}{2}\mathbb{E}[\mathbf{F}|\mathbf{Z}=\mathbf{z}]\mathbb{E}[\mathbf{F}^\top\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-2}\boldsymbol{\beta}\mathbf{F}|\mathbf{Z}=\mathbf{z}],
\end{aligned}\tag{B.33}$$

where (i) follows from plugging in

$$\frac{\partial\|\boldsymbol{\Gamma}_t^{-\frac{1}{2}}(\mathbf{z}-\alpha_t\mathbf{f})\|_2^2}{\partial t}=-\alpha_t^2\mathbf{f}^\top\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-2}\boldsymbol{\beta}\mathbf{f}+\alpha_t\mathbf{f}^\top\mathbf{C}_t\mathbf{z}+\mathbf{z}^\top\boldsymbol{\beta}^\top(\boldsymbol{\Lambda}_t^{-1}-\boldsymbol{\Lambda}_t^{-2})\boldsymbol{\beta}\mathbf{z}$$

with $\mathbf{C}_t=\boldsymbol{\beta}^\top(2\boldsymbol{\Lambda}_t^{-2}-\boldsymbol{\Lambda}_t^{-1})\boldsymbol{\beta}$ and re-arranging terms. To bound $\|\partial\boldsymbol{\xi}/\partial t\|_2$, we provide the following two upper bounds.

Conditional Third Moment Bound. By Cauchy-Schwarz inequality, we have

$$\begin{aligned}
\Big\|\mathbb{E}[\mathbf{F}\mathbf{F}^\top\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-2}\boldsymbol{\beta}\mathbf{F}|\mathbf{Z}=\mathbf{z}]\Big\|_2&\le\sqrt{\mathbb{E}[\|\mathbf{F}\|_2^2|\mathbf{Z}=\mathbf{z}]\cdot\mathbb{E}[\|\mathbf{F}^\top\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-2}\boldsymbol{\beta}\mathbf{F}\|_2^2|\mathbf{Z}=\mathbf{z}]}\\
&\le\frac{1}{h_t+\sigma_d^2\alpha_t^2}\sqrt{\mathbb{E}[\|\mathbf{F}\|_2^2|\mathbf{Z}=\mathbf{z}]\cdot\mathbb{E}[\|\mathbf{F}\|_2^4|\mathbf{Z}=\mathbf{z}]},
\end{aligned}\tag{B.34}$$

where the second inequality holds due to $\boldsymbol{\beta}^\top\boldsymbol{\beta} = \mathbf{I}_k$ and $\|\boldsymbol{\Lambda}_t^{-2}\|_{\mathrm{op}} \le 1/(h_t + \sigma_d^2\alpha_t^2)$.

Conditional Covariance Bound. Recall $\mathbf{s}_{\mathrm{sub}}$ defined in (13). Taking the derivative of $\mathbf{s}_{\mathrm{sub}}$ with respect to $\mathbf{z}$, we have

$$\begin{aligned}\frac{\partial \mathbf{s}_{\mathrm{sub}}(\mathbf{z},t)}{\partial \mathbf{z}} &= -\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta} + \alpha_t^2\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\frac{\int \mathbf{f}\mathbf{f}^\top\boldsymbol{\Gamma}_t^{-1}\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}}{\int \phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}} \\ &\quad - \alpha_t^2\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\frac{\int \mathbf{f}\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}}{\int \phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}}\frac{\int \mathbf{f}^\top\boldsymbol{\Gamma}_t^{-1}\phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}}{\int \phi(\mathbf{z};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\mathrm{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}} \\ &= \alpha_t^2\boldsymbol{\Lambda}_t^{-1}\boldsymbol{\beta}\left[\mathrm{Cov}\left(\mathbf{F}|\mathbf{Z}=\mathbf{z}\right)\boldsymbol{\Gamma}_t^{-1} - \frac{1}{\alpha_t^2}\mathbf{I}_k\right]. \end{aligned} \tag{B.35}$$

Since $\mathbf{s}_{\mathrm{sub}}$ is L_s-Lipschitz by Assumption 3, we deduce from (B.35) that for any $t \in (0,T]$, it holds

$$\|\mathrm{Cov}\left(\mathbf{F}|\mathbf{Z}=\mathbf{z}\right)\|_{\mathrm{op}} \le \frac{(h_t+\sigma_{\max}^2\alpha_t^2)(1+L_s(h_t+\sigma_{\max}^2\alpha_t^2))}{\alpha_t^2} \le (1+\sigma_{\max}^4)(1+L_s),$$

where the second inequality follows from taking $t=0$.

Furthermore, as

$$\|\mathbf{C}_t\|_{\mathrm{op}} = \|\boldsymbol{\beta}^\top(2\boldsymbol{\Lambda}_t^{-2} - \boldsymbol{\Lambda}_t^{-1})\boldsymbol{\beta}\|_{\mathrm{op}} \le \|2\boldsymbol{\Lambda}_t^{-2} - \boldsymbol{\Lambda}_t^{-1}\|_{\mathrm{op}} \le \frac{3}{(h_t+\sigma_d^2\alpha_t^2)^2},$$

it holds that

$$\|\mathrm{Cov}[\mathbf{F}|\mathbf{Z}=\mathbf{z}]\mathbf{C}_t\mathbf{z}\|_2 \le \frac{3}{(h_t+\sigma_d^2\alpha_t^2)^2}\|\mathrm{Cov}[\mathbf{F}|\mathbf{Z}=\mathbf{z}]\|_{\mathrm{op}}\|\mathbf{z}\|_2 \tag{B.36}$$

$$\le \frac{3(1+\sigma_{\max}^4)(1+L_s)}{(h_t+\sigma_d^2\alpha_t^2)^2}\|\mathbf{z}\|_2. \tag{B.37}$$

By substituting the conditional third moment bound in (B.34) and covariance bound in (B.36) into (B.33), and using the fact that $\mathbf{F},\mathbf{Z}$ have the sub-Gaussian tails in the compact domain $\mathcal{S}$, we conclude that

$$\tau(S) = \mathcal{O}\Big(L_s(1+\sigma_{\max}^4)\,\mathrm{poly}(\sqrt{k}S)\Big).$$

where $\mathrm{poly}(\cdot)$ represents a cubic polynomial. □

Lemma B.2. *Suppose Assumption 2 holds. Let $\boldsymbol{\xi}$ be defined in (17) and $\mathbf{Z} = \boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{R}$ with distribution P_t^{fac}. Given $\epsilon > 0$, with $S = c\left(\sqrt{(1+\sigma_{\max}^2)(k+\log(1/\epsilon))}\right)$ for some constant $c > 0$, it holds that*

$$\|\boldsymbol{\xi}(\mathbf{Z},t)\,\mathbb{1}\{\|\mathbf{Z}\|_2 > S\}\|_{L^2(P_t^{\mathrm{fac}})} \le \epsilon, \quad \forall t \in (0,T]. \tag{B.38}$$

Proof of Lemma B.2.

Plugging in the expression of $\boldsymbol{\xi}$ in (17), we obtain that

$$
\begin{aligned}
&\int \left\| \int \frac{\mathbf{f}\phi(\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}}{\int\phi(\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}} \right\|_2^2 \mathbb{1}\{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r}\|_2 > S\}p_t(\mathbf{r})\mathrm{d}\mathbf{r} \\
&\overset{(i)}{\leq} \int_{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r}\|_2>S} \|\mathbf{f}\|_2^2 \frac{\phi(\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}}{\int\phi(\boldsymbol{\Gamma}_t\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r};\alpha_t\mathbf{f},\boldsymbol{\Gamma}_t)p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}} p_t(\mathbf{r})\mathrm{d}\mathbf{r} \\
&\overset{(ii)}{\leq} \int\int_{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r}\|_2>S} \|\mathbf{f}\|_2^2\phi(\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r};\alpha_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f},\mathbf{I})\phi((\mathbf{I}-\mathbf{T}_t)\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r};\mathbf{0},\mathbf{I})p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{r}\mathrm{d}\mathbf{f} \\
&= \underbrace{\int_{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r}\|_2>S}\int_{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2\leq\frac{1}{2}\|\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}\|_2} \|\mathbf{f}\|_2^2\phi(\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r};\alpha_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f},\mathbf{I})p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}\mathrm{d}(\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r})}_{(A)} \\
&\quad + \underbrace{\int_{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{r}\|_2>S}\int_{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2>\frac{1}{2}\|\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}\|_2} \|\mathbf{f}\|_2^2\phi(\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r};\alpha_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f},\mathbf{I})p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}\mathrm{d}(\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r})}_{(B)},
\end{aligned}
$$

where (i) holds due to the Cauchy-Schwarz inequality, (ii) invokes the expression of $p_t(\mathbf{r})$ in (A.2) and re-arranging terms, and the last equality holds by straightforward calculations.

Bounding Term (A)**.** We define the change of variable $\mathbf{X} := \mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{R}_t$ and denote by $\mathbf{x}$ a realization of $\mathbf{X}$. By the Cauchy-Schwarz inequality and $\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2 \leq \frac{1}{2}\|\mathbf{T}_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{r}\|_2$, we have

$$
\|\mathbf{x}-\alpha_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2^2 \geq \frac{1}{2}\|\mathbf{x}\|_2^2 - \alpha_t^2\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2^2 \geq \frac{1}{4}\|\mathbf{x}\|_2^2.
$$

As a result, we can deduce that

$$
\begin{aligned}
(A) &\leq \int_{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{x}\|_2>S}\int_{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2\leq\frac{1}{2}\|\mathbf{x}\|_2} \|\mathbf{f}\|_2^2(2\pi)^{-\frac{k}{2}}\exp\left(-\frac{\|\mathbf{x}\|_2^2}{8}\right)p_{\text{fac}}(\mathbf{f})\mathrm{d}\mathbf{f}\mathrm{d}\mathbf{x} \\
&\leq \mathbb{E}\left[\|\mathbf{f}\|_2^2\right]\cdot\int_{\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\mathbf{x}\|_2>S}(2\pi)^{-\frac{k}{2}}\exp\left(-\frac{\|\mathbf{x}\|_2^2}{8}\right)\mathrm{d}\mathbf{x} \\
&\overset{(i)}{\leq} \mathbb{E}\left[\|\mathbf{f}\|_2^2\right]\cdot\int_{\|\mathbf{x}\|_2>(h_t+\sigma_d^2\alpha_t^2)^{\frac{1}{2}}S}(2\pi)^{-\frac{k}{2}}\exp\left(-\frac{\|\mathbf{x}\|_2^2}{8}\right)\mathrm{d}\mathbf{x} \\
&\overset{(ii)}{\leq} \mathbb{E}\left[\|\mathbf{f}\|_2^2\right]\cdot\frac{2^{-\frac{k}{2}+2}kS^{k-2}(h_t+\sigma_d^2\alpha_t^2)^{(k-2)/2}}{(\frac{1}{2}-\eta)\Gamma(\frac{k}{2}+1)}\exp\left(-\frac{(h_t+\sigma_d^2\alpha_t^2)S^2}{8}\right).
\end{aligned} \tag{B.39}
$$

where (i) holds due to $\|\boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\|_{\text{op}} \leq (h_t+\sigma_d^2\alpha_t^2)^{-\frac{1}{2}}$, and (ii) follows from the sub-Gaussian tail in Proposition 2.6.6 of Vershynin (2018).

Bounding Term (B)**.** We define the change of variable $\mathbf{Y} := \boldsymbol{\beta}^\top\boldsymbol{\Lambda}_t^{-1}\mathbf{R}_t$ and denote by $\mathbf{y}$ the realization of $\mathbf{Y}$. Given $S > \min\{B/2, 1\}$, applying the sub-Gaussian tail of $p_{\text{fac}}(\mathbf{f})$ in (19), we

obtain that

$$
\begin{aligned}
(B) \leq & \int_{\|\mathbf{y}\|_2>S} \int_{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2>\frac{1}{2}\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\Gamma}_t\mathbf{y}\|_2} \phi(\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\Gamma}_t\mathbf{y}; \alpha_t\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}, \mathbf{I}) \cdot \frac{C_1}{(2\pi)^{\frac{k}{2}}}\|\mathbf{f}\|_2^2 \exp\left(-\frac{C_2\|\mathbf{f}\|_2^2}{2}\right) \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{y} \\
\overset{(i)}{\leq} & \frac{C_1}{(2\pi)^k} \int_{\|\mathbf{y}\|_2>S} \int_{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2>\frac{1}{2}\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\Gamma}_t\mathbf{y}\|_2} \exp\left(-\frac{C_2\left\|(\alpha_t^2\mathbf{I}_k + C_2\boldsymbol{\Gamma}_t)^{-\frac{1}{2}}\boldsymbol{\Gamma}_t\mathbf{y}\right\|_2^2}{2}\right) \\
& \cdot \|\mathbf{f}\|_2^2 \exp\left(-\frac{\left\|(\alpha_t^2\boldsymbol{\Gamma}_t^{-1} + C_2\mathbf{I}_k)^{\frac{1}{2}}\left(\mathbf{f} - \alpha_t(\alpha_t^2\boldsymbol{\Gamma}_t^{-1} + C_2\mathbf{I}_k)^{-1}\mathbf{y}\right)\right\|_2^2}{2}\right) \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{y} \\
\overset{(ii)}{\leq} & \frac{C_1}{(2\pi)^k} \int_{\|\mathbf{y}\|_2>S} \int_{\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\mathbf{f}\|_2>\frac{1}{2}\|\boldsymbol{\Lambda}_t^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\Gamma}_t\mathbf{y}\|_2} \exp\left(-\frac{C_2\left\|(\alpha_t^2\mathbf{I}_k + C_2\boldsymbol{\Gamma}_t)^{-\frac{1}{2}}\boldsymbol{\Gamma}_t\mathbf{y}\right\|_2^2}{2}\right) \\
& \cdot \|\mathbf{f}\|_2^2 \exp\left(-\frac{C_2\left\|\mathbf{f} - \alpha_t(\alpha_t^2\boldsymbol{\Gamma}_t^{-1} + C_2\mathbf{I}_k)^{-1}\mathbf{y}\right\|_2^2}{2}\right) \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{y}, \qquad \text{(B.40)}
\end{aligned}
$$

where (i) invokes the formula of $\phi(\mathbf{y}; \alpha_t\mathbf{f}, \boldsymbol{\Gamma}_t)$ in (12) and completing the square for $\mathbf{f}$, (ii) follows from $\|\alpha_t^2\boldsymbol{\Gamma}_t^{-1} + C_2\mathbf{I}_k\|_{\mathrm{op}} \geq C_2$.

Furthermore, applying $\mathbb{E}[\|\mathbf{f}\|_2^2] \leq \alpha_t^2\|(\alpha_t^2\boldsymbol{\Gamma}_t^{-1} + C_2\mathbf{I}_k)^{-\frac{1}{2}}\mathbf{y}\|_2^2 + k$ to (B.40), we deduce that

$$
\begin{aligned}
(B) &\leq \frac{C_1}{C_2^{\frac{k}{2}}(2\pi)^k} \int_{\|\mathbf{y}\|_2>S} \left[\alpha_t^2\|(\alpha_t^2\boldsymbol{\Gamma}_t^{-1} + C_2\mathbf{I}_k)^{-1}\mathbf{y}\|_2^2 + k\right] \cdot \exp\left(-\frac{C_2\left\|(\alpha_t^2\mathbf{I}_k + C_2\boldsymbol{\Gamma}_t)^{-\frac{1}{2}}\boldsymbol{\Gamma}_t\mathbf{y}\right\|_2^2}{2}\right) \mathrm{d}\mathbf{y} \\
&\leq \frac{C_1 2^{-\frac{k}{2}+2} k S^k}{C_2\Gamma(\frac{k}{2}+1)(\alpha_t^2 + C_2(h_t + \sigma_{\max}^2\alpha_t^2))} \exp\left(-\frac{(\alpha_t^2 + C_2(h_t + \sigma_{\max}^2\alpha_t^2))C_2S^2}{2}\right), \qquad \text{(B.41)}
\end{aligned}
$$

where the last inequality is due to $\|\boldsymbol{\Gamma}_t\|_{\mathrm{op}} \geq h_t + \sigma_d^2\alpha_t^2$, $\|(\alpha_t^2\mathbf{I}_k + C_2\boldsymbol{\Gamma}_t)^{-\frac{1}{2}}\|_{\mathrm{op}} \geq \sqrt{\alpha_t^2 + C_2(h_t + \sigma_{\max}^2\alpha_t^2)}$ and the sub-Gaussian tail in Proposition 2.6.6 of Vershynin (2018) and similar operator norm bounds in (ii).

Combining the error bounds (B.39) and (B.41) for (A) and (B), we conclude that

$$
\|\boldsymbol{\xi}(\mathbf{Z}, t)\,\mathbb{1}\{\|\mathbf{Z}\|_2 > S\}\|_{L^2(P_t^{\mathrm{fac}})} \leq c' \frac{2^{-\frac{k}{2}+3} k (h_t + \sigma_d^2\alpha_t^2)^{\frac{k}{2}} S^k}{\Gamma(\frac{k}{2}+1)(\alpha_t^2 + C_2(h_t + \sigma_{\max}^2\alpha_t^2))} \exp\left(-\frac{(h_t + \sigma_d^2\alpha_t^2)S^2}{8}\right) \qquad \text{(B.42)}
$$

for some constant $c' > 0$. Given any $\epsilon > 0$, by the upper bound of truncation error in (B.42), we can choose

$$
S = c\left(\sqrt{\left(1 + \sigma_{\max}^2\right)\left(k + \log\frac{1}{\epsilon}\right)}\right),
$$

such that $\|\boldsymbol{\xi}(\mathbf{Z}, t)\,\mathbb{1}\{\|\mathbf{Z}\|_2 > S\}\|_{L^2(P_t^{\mathrm{fac}})} \leq \epsilon$. Here, c is an absolute constant. □

Lemma B.3. *Suppose Assumption 2 holds. For any $\mathbf{s}_{\boldsymbol{\theta}_1}(\cdot, t)$ and $\mathbf{s}_{\boldsymbol{\theta}_2}(\cdot, t)$, when ρ is sufficiently*

large, it holds that

$$\sup_{\|\mathbf{r}\|_2\le\rho}\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\Big[\left(K+\|\mathbf{R}_t\|_2+\|\mathbf{r}\|_2\right)\|\mathbf{s}_{\boldsymbol{\theta}_1}(\mathbf{R}_t,t)-\mathbf{s}_{\boldsymbol{\theta}_2}(\mathbf{R}_t,t)\|_2\cdot\mathbb{1}\left\{\|\mathbf{R}_t\|_2>3\rho+\sqrt{d\log d}\right\}\Big]$$
$$=\mathcal{O}\bigg(\rho K^2 h_t^{-2-\frac{d}{2}}\left(\frac{\rho}{d}\right)^d\exp\left(-\frac{\rho^2}{h_t}\right)\bigg). \tag{B.43}$$

Proof of Lemma B.3. Denote $\mathbf{D}_{ti}=\mathrm{diag}\{1/(h_t+c_{i1}\alpha_t^2),\ldots,1/(h_t+c_{i1}\alpha_t^2)\}$ for $i=1,2$. Applying the formula of $\mathbf{s}_{\boldsymbol{\theta}_1}$ and $\mathbf{s}_{\boldsymbol{\theta}_2}$ in (18), we calculate

$$\begin{aligned}
&\mathbb{E}_{\mathbf{R}_t|\mathbf{R}_0=\mathbf{r}}\Big[\left(K+\|\mathbf{R}_t\|_2+\|\mathbf{r}\|_2\right)\|\mathbf{s}_{\boldsymbol{\theta}_1}(\mathbf{R}_t,t)-\mathbf{s}_{\boldsymbol{\theta}_2}(\mathbf{R}_t,t)\|_2\cdot\mathbb{1}\left\{\|\mathbf{R}_t\|_2>3\rho+\sqrt{d\log d}\right\}\Big]\\
&\overset{(i)}{\le}\int\bigg(K+\left\|\mathbf{r}'\right\|_2+\|\mathbf{r}\|_2\bigg)\bigg(\|(\mathbf{D}_{t1}-\mathbf{D}_{t2})\mathbf{r}'\|_2+\|\alpha_t(\mathbf{D}_{t1}\mathbf{V}_1-\mathbf{D}_{t2}\mathbf{V}_2)\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top\mathbf{D}_{t1}\mathbf{r}',t)\|_2\\
&\quad+\|\alpha_t\mathbf{D}_{t2}\mathbf{V}_2(\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top\mathbf{D}_{t1}\mathbf{r}',t)-\mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_2^\top\mathbf{D}_{t2}\mathbf{r}',t))\|_2\bigg)\cdot\mathbb{1}\left\{\left\|\mathbf{r}'\right\|_2>3\rho+\sqrt{d\log d}\right\}\phi(\mathbf{r}';\alpha_t\mathbf{r},h_t\mathbf{I})\mathrm{d}\mathbf{r}'\\
&\overset{(ii)}{=}\mathcal{O}\bigg(\int_{\|\mathbf{r}'\|_2>3\rho+\sqrt{d\log d}}\frac{(K+\|\mathbf{r}'\|_2+\|\mathbf{r}\|_2)(K+\|\mathbf{r}'\|_2)}{h_t^2(2\pi h_t)^{\frac{d}{2}}}\exp\left(-\frac{1}{2h_t}\left(\frac{1}{2}\left\|\mathbf{r}'\right\|_2^2-\|\mathbf{r}\|_2^2\right)\right)\mathrm{d}\mathbf{r}'\bigg),
\end{aligned}\tag{B.44}$$

where (i) is due to Cauchy-Schwarz inequality; (ii) follows from the upper bounds (15) and (18) of $\{\mathbf{g}_{\boldsymbol{\theta}_i},\mathbf{V}_i,\mathbf{D}_{ti}\}_{i=1,2}$, $\alpha_t^2\le 1$ and

$$\phi(\mathbf{r}';\alpha_t\mathbf{r},h_t\mathbf{I})\le(2\pi h_t)^{-\frac{d}{2}}\exp\left(-\frac{1}{2h_t}\left(\frac{1}{2}\left\|\mathbf{r}'\right\|_2^2-\|\mathbf{r}\|_2^2\right)\right).$$

Then, substituting the upper bound for the tail estimation in Proposition 2.6.6 of Vershynin (2018) into (B.44), we deduce that

$$\begin{aligned}
(\text{B.44})&=\mathcal{O}\bigg(\frac{(K^2+K\|\mathbf{r}\|_2)(2h_t)^{-2-\frac{d}{2}}(3\rho+\sqrt{d\log d})^{d-2}}{\Gamma(\frac{d}{2}+1)}\exp\left(-\frac{(3\rho+\sqrt{d\log d})^2}{4h_t}+\frac{\|\mathbf{r}\|_2^2}{2h_t}\right)\bigg)\\
&\quad+\frac{(2K+\|\mathbf{r}\|_2)(2h_t)^{-2-\frac{d}{2}}(3\rho+\sqrt{d\log d})^{d-1}}{\Gamma(\frac{d}{2}+1)}\exp\left(-\frac{(3\rho+\sqrt{d\log d})^2}{4h_t}+\frac{\|\mathbf{r}\|_2^2}{2h_t}\right)\\
&\quad+\frac{(2h_t)^{-2-\frac{d}{2}}(3\rho+\sqrt{d\log d})^{d}}{\Gamma(\frac{d}{2}+1)}\exp\left(-\frac{(3\rho+\sqrt{d\log d})^2}{4h_t}+\frac{\|\mathbf{r}\|_2^2}{2h_t}\right)\\
&=\mathcal{O}\left(K^2\|\mathbf{r}\|_2 h_t^{-2-\frac{d}{2}}\left(\frac{\rho}{d}\right)^d\exp\left(-\frac{9\rho^2-2\|\mathbf{r}\|_2^2}{4h_t}\right)\right).
\end{aligned}\tag{B.45}$$

Here, the last inequality holds since $\Gamma(\frac{d}{2}+1)=\mathcal{O}(\prod_{j=1}^{\frac{d}{2}}j)$ and

$$\frac{2^{-\frac{d}{2}}(3\rho+\sqrt{d\log d})^d\exp\left(-\frac{6\rho\sqrt{d\log d}+d\log d}{4h_t}\right)}{\Gamma(\frac{d}{2}+1)}=\mathcal{O}\bigg(\left(\frac{\rho}{d}\right)^d\bigg)$$

for a sufficiently large $\rho > \max\{B, d\}$.

Immediately, substituting $\|\mathbf{r}\|_2 \leq \rho$ into (B.45), we obtain the desired result. □

Lemma B.4. *For any given $\epsilon > 0$, $\delta > 0$, and $\rho = \mathcal{O}\left(\sqrt{\sigma_{\max}^2 (d + \log K + \log(n/\delta))}\right)$ defined in (B.25), the ν-covering number of $\mathcal{S}_{\mathrm{NN}}$ in (18) is*

$$\log \mathfrak{N}(\nu, \mathcal{S}_{NN}, \|\cdot\|_2) = \left(\left(dk + T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon}+k\right)^{\frac{k+4}{2}}\right)\log\frac{T\tau dk}{t_0\nu\epsilon}\right). \tag{B.46}$$

Proof of Lemma B.4. $\mathcal{S}_{\mathrm{NN}}$ consists of three components:

1. A vector $\mathbf{c} = (c_1, c_2, \ldots, c_d) \in [0, \sigma_{\max}]^d$ and its induced matrix
$$\mathbf{D}_t = \mathrm{diag}\{1/(h_t + \alpha_t^2 c_1), 1/(h_t + \alpha_t^2 c_2), \ldots, 1/(h_t + \alpha_t^2 c_d)\}.$$
2. A matrix $\mathbf{V}$ with orthonormal columns.
3. A ReLU network $\mathbf{g}_{\boldsymbol{\zeta}}$.

Denote $\mathbf{D}_{ti} = \mathrm{diag}\{1/(h_t + \alpha_t^2 c_{i1}), 1/(h_t + \alpha_t^2 c_{i2}), \ldots, 1/(h_t + \alpha_t^2 c_{id})\}$ for $i = 1, 2$. Directly incorporating the sub-additive property of L^2 norm and $\alpha_t^2 \leq 1$, we have

$$\begin{aligned}
&\|\mathbf{s}_{\boldsymbol{\theta}_1}(\mathbf{r}, t) - \mathbf{s}_{\boldsymbol{\theta}_2}(\mathbf{r}, t)\|_2 \\
\leq\ & \|(\mathbf{D}_{t1}\mathbf{V}_1 - \mathbf{D}_{t2}\mathbf{V}_1)\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t)\|_2 + \|(\mathbf{D}_{t2}\mathbf{V}_1 - \mathbf{D}_{t2}\mathbf{V}_2)\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t)\|_2 \\
&+ \|\mathbf{D}_{t2}\mathbf{V}_2(\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t) - \mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t))\|_2 + \|\mathbf{D}_{t2}\mathbf{V}_2(\mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t) - \mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_2^\top \mathbf{D}_{t1}\mathbf{r}, t))\|_2 \\
&+ \|\mathbf{D}_{t2}\mathbf{V}_2(\mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_2^\top \mathbf{D}_{t1}\mathbf{r}, t) - \mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_2^\top \mathbf{D}_{t2}\mathbf{r}, t))\|_2 + \|(\mathbf{D}_{t1} - \mathbf{D}_{t2})\mathbf{r}\|_2 \\
\leq\ & \|\mathbf{D}_{t1} - \mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t)\|_2 + \|\mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{V}_1 - \mathbf{V}_2\|_{\mathrm{op}}\|\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t)\|_2 \\
&+ \|\mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t) - \mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{V}_1^\top \mathbf{D}_{t1}\mathbf{r}, t)\|_2 + \gamma\|\mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{V}_1 - \mathbf{V}_2^\top\|_{\mathrm{op}}\|\mathbf{D}_{t1}\|_{\mathrm{op}}\|\|\mathbf{r}\|_2 \\
&+ \gamma\|\mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{D}_{t1} - \mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{r}\|_2 + \|\mathbf{D}_{t1} - \mathbf{D}_{t2}\|_{\mathrm{op}}\|\mathbf{r}\|_2,
\end{aligned} \tag{B.47}$$

where the last inequality follows from the fact that $\{\mathbf{V}_i\}_{i=1,2}$ are orthogonal and $\{\mathbf{g}_{\boldsymbol{\zeta}_i}\}_{i=1,2}$ is γ-Lipschitz.

To analyze the covering number of $\mathcal{S}_{\mathrm{NN}}$, we consider

$$\|\mathbf{c}_1 - \mathbf{c}_2\|_\infty \leq \delta_c, \|\mathbf{V}_1 - \mathbf{V}_2\|_{\mathrm{op}} \leq \delta_V, \text{ and } \sup_{\|\mathbf{r}\|_2 \leq 3\rho + \sqrt{d\log d}, t\in[t_0,T]} \|\mathbf{g}_{\boldsymbol{\zeta}_1}(\mathbf{r}, t) - \mathbf{g}_{\boldsymbol{\zeta}_2}(\mathbf{r}, t)\|_2 \leq \delta_f. \tag{B.48}$$

Immediately, we can deduce that

$$\sup_{t\in[t_0,T]} \|\mathbf{D}_{t1} - \mathbf{D}_{t2}\|_{\mathrm{op}} \leq \frac{\delta_c}{t_0^2}. \tag{B.49}$$

Then, on the domain with radius $\|\mathbf{r}\|_2 \le 3\rho + \sqrt{d\log d}$ and $t \in [t_0, T]$, by substituting the upper bounds (B.48) and (B.49) into (B.47), we obtain

$$\begin{aligned}
&\sup_{\|\mathbf{r}\|_2 \le 3\rho+\sqrt{d\log d}, t\in[t_0,T]} \|\mathbf{s}_{\boldsymbol{\theta}_1}(\mathbf{r},t) - \mathbf{s}_{\boldsymbol{\theta}_2}(\mathbf{r},t)\|_2 \\
&\le \frac{\delta_c K}{t_0^2} + \frac{\delta_V K}{t_0} + \frac{\delta_f}{t_0} + \frac{\gamma\delta_V(3\rho+\sqrt{d\log d})}{t_0^2} + \frac{\gamma\delta_c(3\rho+\sqrt{d\log d})}{t_0^3} + \frac{\delta_c(3\rho+\sqrt{d\log d})}{t_0^2} \\
&\overset{(i)}{=} \frac{\delta_c(\gamma(3\rho+\sqrt{d\log d}) + t_0 K + t_0(3\rho+\sqrt{d\log d}))}{t_0^3} + \frac{\delta_V(\gamma(3\rho+\sqrt{d\log d}) + t_0 K)}{t_0^2} + \frac{\delta_f}{t_0} \\
&\overset{(ii)}{=} \mathcal{O}\left(\frac{\delta_c\gamma(3\rho+\sqrt{d\log d}) + t_0\delta_V\gamma(3\rho+\sqrt{d\log d}) + t_0^2\delta_f}{t_0^3}\right),
\end{aligned}$$

where (i) follows from rearranging terms, and (ii) holds by omitting higher-order terms on δ_c, δ_V, and δ_f. For a hypercube $[0, \sigma_{\max}]^d$, the δ_c-covering number is bounded by $\left(\frac{\sigma_{\max}}{\delta_c}\right)^d$. For a set of matrices $\{\mathbf{V} \in \mathbb{R}^{d\times k} : \|\mathbf{V}\|_{\text{op}} \le 1\}$, its δ_V-covering number is bounded by $\left(1 + \frac{2\sqrt{k}}{\delta_V}\right)^{dk}$ (a standard volume-ratio bound for matrices with a bounded operator norm; see Lemma 5.7 and Example 5.8 of Wainwright (2019) and Lemma 8 in Chen, Li, and Zhao (2020)). Following Lemma 5.3 in Chen et al. (2022), which provides δ_f-covering number bounds for ReLU network classes (see also Chapters 14 and 16 of Anthony and Bartlett (2009)), we take the upper bound $\left(\frac{2L^2M(3\rho+\sqrt{d\log d})\kappa^L M^{L+1}}{\delta_f}\right)^J$ for the δ_f-covering number of the function class (15). Therefore, with $\rho = \mathcal{O}\big(\sqrt{\sigma_{\max}^2(d + \log K + \log(n/\delta))}\big)$, we have

$$\begin{aligned}
\log \mathfrak{N}(\nu, \mathcal{S}_{\text{NN}}, \|\cdot\|_2) &\le \mathcal{O}\left(d\log\left(\frac{\sigma_{\max}\gamma(3\rho+\sqrt{d\log d})}{t_0^3\nu}\right) + dk\log\left(1 + \frac{2\sqrt{k}\gamma(3\rho+\sqrt{d\log d})}{t_0^2\nu}\right)\right. \\
&\qquad \left. + J\log\left(\frac{2L^2M(3\rho+\sqrt{d\log d})\kappa^L M^{L+1}}{t_0\nu}\right)\right) \\
&\overset{(i)}{=} \mathcal{O}\left(\left(dk + T\tau(1+L_s)^k(1+\sigma_{\max}^k)\epsilon^{-(k+1)}\left(\log\frac{1}{\epsilon} + k\right)^{\frac{k+4}{2}}\right)\log\frac{T\tau dk}{t_0\nu\epsilon}\right),
\end{aligned}$$

where (i) follows from invoking the order of network parameters in (22) in Theorem 1 and omitting higher orders terms such as $\log\log d$ and $\log\log k$. □

C Omitted Proofs in Section 5

In this section, we provide the proof of Theorem 3 and the lemmas used in the proof.

C.1 Proof of Theorem 3

Proof. The proof contains two parts: the distribution estimation and the latent subspace recovery. For notational simplicity, let us denote

$$\epsilon := \frac{1}{t_0}(1+\sigma_{\max}^{2k}) d^{\frac{5}{2}} k^{\frac{k+10}{2}} n^{-\frac{2-2\delta(n)}{k+5}} \log^4 n. \tag{C.1}$$

Part 1: Return Distribution Estimation. First, we can decompose $\mathrm{TV}(P_{\mathrm{data}}, \widehat{P}_{t_0})$ into

$$\mathrm{TV}(P_{\mathrm{data}}, \widehat{P}_{t_0}) \le \mathrm{TV}(P_{\mathrm{data}}, P_{t_0}) + \mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0}) + \mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0}),$$

where P_{data} is the initial distribution of $\mathbf{R}$ in (8), $\widehat{P}_{t_0}$ and $\widetilde{P}_{t_0}$ are the marginal distribution of the estimated backward process $\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}$ in (4) initialized with $\widehat{\mathbf{R}}^{\leftarrow}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ and $\widehat{\mathbf{R}}^{\leftarrow}_0 \sim P_T$, respectively. Here, $\mathrm{TV}(P_{\mathrm{data}}, P_{t_0})$ is the early-stopping error, $\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0})$ captures the approximation error of the score estimation, and $\mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0})$ reflects the mixing error. We bound each term in Lemma C.1 and the error bound (C.28) is given by

$$\begin{aligned}
\mathrm{TV}\left(P_{\mathrm{data}}, \widehat{P}_{t_0}\right) &= \mathcal{O}\left(dt_0 L_s(1+\sigma_{\max}^2) + \sqrt{\epsilon(T-t_0)} + \sqrt{\mathrm{KL}\left(P_{\mathrm{data}} \| \mathcal{N}\left(\mathbf{0}, \mathbf{I}_d\right)\right)} \exp(-T)\right) \\
&= \tilde{\mathcal{O}}\left((1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{2(k+5)}} \log^{\frac{5}{2}} n\right)
\end{aligned}$$

where the last equality follows from invoking the order of ϵ in (C.1), $t_0 = n^{-\frac{1-\delta(n)}{k+5}}$, and $T = \mathcal{O}(\log n)$ and omitting the lower-order terms in dt_0. Hence the distribution estimation result in Theorem 3 is completed.

Part 2: Latent Subspace Recovery. First, we generate $m = \mathcal{O}\left(\lambda_{\max}^{-2}(\mathbf{\Sigma}_0) dn^{\frac{2(1-\delta(n))}{k+5}} \log n\right)$ new samples via Algorithm 1. By the error bound (26) in Lemma 2, we obtain that, with probability $1 - 1/n$, it holds

$$\left\| \widehat{\mathbf{\Sigma}}_0 - \mathbf{\Sigma}_0 \right\|_{\mathrm{op}} = \tilde{\mathcal{O}}\left(\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\right).$$

Therefore, applying Weyl's theorem to $\widehat{\mathbf{\Sigma}}_0$ and $\mathbf{\Sigma}_0$, we deduce that for any $i = 1, 2, \ldots, d$, it holds that

$$\left|\lambda_i(\widehat{\mathbf{\Sigma}}_0) - \lambda_i(\mathbf{\Sigma}_0)\right| = \tilde{\mathcal{O}}\left(\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\right).$$

In addition, for any $i = 1, 2, \ldots, k$, it holds that

$$\left|\frac{\lambda_i(\widehat{\mathbf{\Sigma}}_0)}{\lambda_i(\mathbf{\Sigma}_0)} - 1\right| = \tilde{\mathcal{O}}\left(\frac{\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}}}{\lambda_i(\mathbf{\Sigma}_0)} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\right).$$

Next, we analyze the SVD of $\widehat{\mathbf{\Sigma}}_0$. Recall that the top k-dimensional eigenspace of $\mathbf{\Sigma}_0$ and $\widehat{\mathbf{\Sigma}}_0$ are denoted as $\mathbf{U}$ and $\widehat{\mathbf{U}}$, respectively. For any $j = 1, 2, \ldots, k$, define

$$\cos\angle_j(\widehat{\mathbf{U}}, \mathbf{U}) := \max_{\hat{\mathbf{u}}\in\mathrm{Col}(\widehat{\mathbf{U}}), \mathbf{u}\in\mathrm{Col}(\mathbf{U})} \frac{|\hat{\mathbf{u}}^\top \mathbf{u}|}{\|\hat{\mathbf{u}}\|_2\|\mathbf{u}\|_2} := |\hat{\mathbf{u}}_j^\top \mathbf{u}_j|,$$
$$\text{subject to } \hat{\mathbf{u}}_j^\top \hat{\mathbf{u}}_\ell = 0 \text{ and } \mathbf{u}_j^\top \mathbf{u}_\ell = 0, \text{ for any } \ell = 1, 2, \ldots, j-1,$$

where $\mathrm{Col}(\cdot)$ represents the column space, $\hat{\mathbf{u}}_0 := \mathbf{0}$, and $\mathbf{u}_0 := \mathbf{0}$. Applying Davis-Kahan-$\sin(\theta)$ Theorem of Davis and Kahan (1970) to $\widehat{\mathbf{U}}$ and $\mathbf{U}$, we have

$$\|\sin\angle(\widehat{\mathbf{U}}, \mathbf{U})\|_{\mathrm{F}} \le \frac{\|\widehat{\mathbf{\Sigma}}_0 - \mathbf{\Sigma}_0\|_{\mathrm{F}}}{\lambda_k(\mathbf{\Sigma}_0) - \lambda_{k+1}(\mathbf{\Sigma}_0)} = \mathcal{O}\bigg(\frac{\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}}}{\texttt{Eigen-gap}(k)} \cdot n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\bigg), \tag{C.2}$$

where

$$\sin\angle(\widehat{\mathbf{U}}, \mathbf{U}) := \Big(1 - \cos^2\angle_1(\widehat{\mathbf{U}}, \mathbf{U}), 1 - \cos^2\angle_2(\widehat{\mathbf{U}}, \mathbf{U}), \ldots, 1 - \cos^2\angle_k(\widehat{\mathbf{U}}, \mathbf{U})\Big). \tag{C.3}$$

The inequality in (C.2) follows from the fact that $\|\mathbf{A}\|_{\mathrm{F}} \le \sqrt{k}\|\mathbf{A}\|_{\mathrm{op}}$ for any $\mathbf{A} \in \mathbb{R}^{d\times k}$. By the property of SVD, we can find two orthogonal matrices $\mathbf{O}_1, \mathbf{O}_2 \in \mathbb{R}^{k\times k}$ such that

$$\widehat{\mathbf{U}}^\top \mathbf{U} = \mathbf{O}_1^\top \,\mathrm{diag}\Big\{\cos\angle_1(\widehat{\mathbf{U}}, \mathbf{U}), \cos\angle_2(\widehat{\mathbf{U}}, \mathbf{U}), \ldots, \cos\angle_k(\widehat{\mathbf{U}}, \mathbf{U})\Big\}\, \mathbf{O}_2.$$

Immediately, it holds that

$$\widehat{\mathbf{U}}^\top \mathbf{U}\mathbf{U}^\top \widehat{\mathbf{U}} = \mathbf{O}_1^\top \,\mathrm{diag}\left\{\cos^2(\angle_1), \ldots, \cos^2(\angle_k)\right\} \mathbf{O}_1. \tag{C.4}$$

Therefore, we deduce that

$$\begin{aligned}
\|\widehat{\mathbf{U}}\widehat{\mathbf{U}}^\top - \mathbf{U}\mathbf{U}^\top\|_{\mathrm{F}}^2 &= \mathrm{tr}(\widehat{\mathbf{U}}\widehat{\mathbf{U}}^\top + \mathbf{U}\mathbf{U}^\top - 2\widehat{\mathbf{U}}\widehat{\mathbf{U}}^\top\mathbf{U}\mathbf{U}^\top) \\
&\overset{(i)}{=} \mathrm{tr}(\widehat{\mathbf{U}}^\top\widehat{\mathbf{U}}) + \mathrm{tr}(\mathbf{U}^\top\mathbf{U}) - 2\,\mathrm{tr}(\mathbf{O}_1^\top \,\mathrm{diag}\left\{\cos^2(\angle_1), \ldots, \cos^2(\angle_k)\right\} \mathbf{O}_1) \\
&\overset{(ii)}{=} 2k - \sum_{i=1}^k \cos^2(\angle_i) = 2\|\sin\angle(\widehat{\mathbf{U}}, \mathbf{U})\|_{\mathrm{F}}^2,
\end{aligned} \tag{C.5}$$

where (i) invokes (C.4) and (ii) holds due to the fact that $\widehat{\mathbf{U}}$, $\mathbf{U}$, and $\mathbf{O}_1$ have orthogonal columns, and the last equality holds by the definition of $\sin\angle$ defined in (C.3). Therefore, substituting the error bound of $\|\sin\angle(\widehat{\mathbf{U}}, \mathbf{U})\|_{\mathrm{F}}^2$ in (C.2) into (C.5), we obtain

$$\|\widehat{\mathbf{U}}\widehat{\mathbf{U}}^\top - \mathbf{U}\mathbf{U}^\top\|_{\mathrm{F}} = \tilde{\mathcal{O}}\bigg(\frac{\lambda_{\max}(\mathbf{\Sigma}_0)(1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+12}{4}}}{\texttt{Eigen-gap}(k)} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n\bigg). \qquad \square$$

C.2 Supporting Lemmas for Theorem 3

Recall that Lemma 2 and Lemma Lemma 3 are key results of our development. We provide their proofs in Appendices C.2.1 and C.2.2 respectively. Some additional lemmas that support the proof of Theorem 3 are stated and proved in Appendix C.2.3.

C.2.1 Proof of Lemma 2

Proof.

For notational simplicity, let us define

$$\begin{aligned}
\boldsymbol{\Sigma}_{t_0} &:= \mathbb{E}_{\mathbf{R}_{t_0}\sim P_{t_0}}\left[\mathbf{R}_{t_0}\mathbf{R}_{t_0}^\top\right] - \mathbb{E}_{\mathbf{R}_{t_0}\sim P_{t_0}}[\mathbf{R}_{t_0}]\mathbb{E}_{\mathbf{R}_{t_0}\sim P_{t_0}}[\mathbf{R}_{t_0}]^\top, \\
\widetilde{\boldsymbol{\Sigma}}_{t_0} &:= \mathbb{E}_{\mathbf{R}_{t_0}\sim \widetilde{P}_{t_0}}\left[\mathbf{R}_{t_0}\mathbf{R}_{t_0}^\top\right] - \mathbb{E}_{\mathbf{R}_{t_0}\sim \widetilde{P}_{t_0}}[\mathbf{R}_{t_0}]\mathbb{E}_{\mathbf{R}_{t_0}\sim \widetilde{P}_{t_0}}[\mathbf{R}_{t_0}]^\top, \text{ and} \\
\check{\boldsymbol{\Sigma}}_{t_0} &= \mathbb{E}_{\mathbf{R}_{t_0}\sim \widehat{P}_{t_0}}\left[\mathbf{R}_{t_0}\mathbf{R}_{t_0}^\top\right] - \mathbb{E}_{\mathbf{R}_{t_0}\sim \widehat{P}_{t_0}}[\mathbf{R}_{t_0}]\mathbb{E}_{\mathbf{R}_{t_0}\sim \widehat{P}_{t_0}}[\mathbf{R}_{t_0}]^\top.
\end{aligned} \tag{C.6}$$

The proof is based on the following error decomposition.

Error Decomposition. We decompose the target operator norm as

$$\left\|\widehat{\boldsymbol{\Sigma}}_0 - \boldsymbol{\Sigma}_0\right\|_{\text{op}} \le \underbrace{\left\|\boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_{t_0}\right\|_{\text{op}}}_{(A)} + \underbrace{\left\|\boldsymbol{\Sigma}_{t_0} - \widetilde{\boldsymbol{\Sigma}}_{t_0}\right\|_{\text{op}}}_{(B)} + \underbrace{\left\|\widetilde{\boldsymbol{\Sigma}}_{t_0} - \check{\boldsymbol{\Sigma}}_{t_0}\right\|_{\text{op}}}_{(C)} + \underbrace{\left\|\widehat{\boldsymbol{\Sigma}}_0 - \check{\boldsymbol{\Sigma}}_{t_0}\right\|_{\text{op}}}_{(D)}, \tag{C.7}$$

where term (A) is the early-stopping error, term (B) is the approximation error of $\mathcal{S}_{\text{NN}}$ term (C) is the mixing error of forward process (1), term (D) is the finite-sample error.

Bounding Term (A). Using the fact that $\mathbf{R}_{t_0} = e^{-t_0/2}\mathbf{R}_0 + \mathbf{B}_{1-e^{-t_0}}$, we have

$$\boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_{t_0} = \boldsymbol{\Sigma}_0 - e^{-t_0}\boldsymbol{\Sigma}_0 - (1-e^{-t_0})\mathbf{I}_d = (1-e^{-t_0})(\boldsymbol{\Sigma}_0 - \mathbf{I}_d).$$

Therefore, by the definition of (A) in (C.7) and $t_0 = n^{-\frac{1-\delta(n)}{k+5}}$, we obtain

$$(A) = \mathcal{O}\Big(\lambda_{\max}(\boldsymbol{\Sigma}_0)\cdot n^{-\frac{1-\delta(n)}{k+5}}\Big). \tag{C.8}$$

Bounding Term (B). Under the coupled SDE system (28), we have

$$\begin{aligned}
(B) &\le \left\|\mathbb{E}_{\mathbf{R}_{t_0}\sim P_{t_0}}\left[\mathbf{R}_{t_0}\mathbf{R}_{t_0}^\top\right] - \mathbb{E}_{\mathbf{R}_{t_0}\sim \widehat{P}_{t_0}}\left[\mathbf{R}_{t_0}\mathbf{R}_{t_0}^\top\right]\right\|_{\text{op}} \\
&\quad + \left\|\mathbb{E}_{\mathbf{R}_{t_0}\sim P_{t_0}}[\mathbf{R}_{t_0}]\mathbb{E}_{\mathbf{R}_{t_0}\sim P_{t_0}}[\mathbf{R}_{t_0}]^\top - \mathbb{E}_{\mathbf{R}_{t_0}\sim \widehat{P}_{t_0}}[\mathbf{R}_{t_0}]\mathbb{E}_{\mathbf{R}_{t_0}\sim \widehat{P}_{t_0}}[\mathbf{R}_{t_0}]^\top\right\|_{\text{op}} \\
&= \left\|\mathbb{E}\left[(\mathbf{R}^{\leftarrow}_{T-t_0})(\mathbf{R}^{\leftarrow}_{T-t_0})^\top\right] - \mathbb{E}\left[(\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0})(\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0})^\top\right]\right\|_{\text{op}}
\end{aligned} \tag{C.9}$$

$$+ \left\| \mathbb{E}\left[\mathbf{R}^{\leftarrow}_{T-t_0}\right] \mathbb{E}\left[\mathbf{R}^{\leftarrow}_{T-t_0}\right]^\top - \mathbb{E}\left[\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\right] \mathbb{E}\left[\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\right]^\top \right\|_{\mathrm{op}}, \tag{C.10}$$

where the last equality invokes $\mathbf{R}^{\leftarrow}_t$ and $\widehat{\mathbf{R}}^{\leftarrow}_t$ defined in (28). For term (C.9), we have

$$\begin{aligned}(\text{C.9}) &\le \left\| \mathbb{E}\left[(\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0})(\mathbf{R}^{\leftarrow}_{T-t_0})^\top\right] \right\|_{\mathrm{op}} + \left\| \mathbb{E}\left[(\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0})(\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0})^\top\right] \right\|_{\mathrm{op}} \\ &\le \sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \cdot \left(\sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2} + \sqrt{\mathbb{E}\|\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \right), \end{aligned} \tag{C.11}$$

where (C.11) follows from the Cauchy-Schwarz inequality and rearranging terms. Similarly, for term (C.10), using the Cauchy-Schwarz inequality, we have

$$\begin{aligned}(\text{C.10}) &\le \left\| (\mathbb{E}\left[\mathbf{R}^{\leftarrow}_{T-t_0}\right] - \mathbb{E}[\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}]) \mathbb{E}\left[\mathbf{R}^{\leftarrow}_{T-t_0}\right]^\top \right\|_{\mathrm{op}} + \left\| \mathbb{E}[\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}] (\mathbb{E}\left[\mathbf{R}^{\leftarrow}_{T-t_0}\right]^\top - \mathbb{E}[\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}]^\top) \right\|_{\mathrm{op}} \\ &\le \sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \cdot \left(\sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2} + \sqrt{\mathbb{E}\|\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \right), \end{aligned} \tag{C.12}$$

Then, substituting (C.11) and (C.12) into (C.9) and (C.10), we deduce that

$$\begin{aligned}(B) &\le 2\sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \cdot \left(\sqrt{\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2} + \sqrt{\mathbb{E}\|\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2} \right) \\ &= \mathcal{O}\left((1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n \right), \end{aligned} \tag{C.13}$$

where the last equality follows from applying the upper bound (29) in Lemma 3 and using the fact that

$$\begin{aligned}&\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2 = \mathbb{E}\|e^{-t_0/2}\mathbf{R}_0 + \mathbf{B}_{1-e^{-t_0}}\|_2^2 \le e^{-t_0}\mathbb{E}\|\mathbf{R}_0\|_2^2 + 1 - e^{-t_0} = \mathcal{O}(1) \quad \text{and} \\ &\mathbb{E}\|\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2 \le 2\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2 + 2\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2 = \mathcal{O}(1). \end{aligned}$$

Bounding Term (C)**.** Applying Lemma 3 to the estimated backward process starting from P_T and $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$, respectively, we obtain

$$\left\| \widetilde{\mathbf{\Sigma}}_{t_0} - \check{\mathbf{\Sigma}}_{t_0} \right\|_{\mathrm{op}} = \mathcal{O}(2\mathbb{E}\|\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0} - \mathbf{R}^{\leftarrow}_{T-t_0}\|_2^2) = \mathcal{O}\left((1+\sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n \right). \tag{C.14}$$

Bounding Term (D)**.** By introducing the estimation error between $\bar{\mathbf{R}}_0$ and $\mathbb{E}[\mathbf{R}_i]$, we have

$$\begin{aligned}(D) &\le \left\| \frac{1}{m-1} \sum_{i=1}^m \left((\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])(\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])^\top \right) - \check{\mathbf{\Sigma}}_{t_0} \right\|_{\mathrm{op}} \\ &\quad + \left\| \bar{\mathbf{R}}_0 (\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i])^\top \right\|_{\mathrm{op}} + \left\| (\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i]) \mathbb{E}[\mathbf{R}_i]^\top \right\|_{\mathrm{op}} \\ &\overset{(i)}{\le} \left\| \frac{1}{m-1} \sum_{i=1}^m \left((\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])(\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])^\top \right) - \check{\mathbf{\Sigma}}_{t_0} \right\|_{\mathrm{op}} \end{aligned}$$

$$
\begin{aligned}
&+ \|\bar{\mathbf{R}}_0\|_2 \|\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i]\|_2 + \|\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i]\|_2 \|\mathbb{E}[\mathbf{R}_i]\|_2 \\
&\overset{(ii)}{\leq} \left\| \frac{1}{m-1} \sum_{i=1}^{m} \left((\check{\boldsymbol{\Sigma}}_{t_0})^{-1/2} (\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])(\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])^\top (\check{\boldsymbol{\Sigma}}_{t_0})^{-1/2} - \mathbf{I}_d \right) \right\|_{\mathrm{op}} \left\| \check{\boldsymbol{\Sigma}}_{t_0} \right\|_{\mathrm{op}} \\
&\quad + \|(\check{\boldsymbol{\Sigma}}_{t_0})^{-\frac{1}{2}} \bar{\mathbf{R}}_0\|_2 \|(\check{\boldsymbol{\Sigma}}_{t_0})^{-\frac{1}{2}} (\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i])\|_2 \|\check{\boldsymbol{\Sigma}}_{t_0}\|_{\mathrm{op}} \\
&\qquad + \|(\check{\boldsymbol{\Sigma}}_{t_0})^{-\frac{1}{2}} (\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i])\|_2 \|(\check{\boldsymbol{\Sigma}}_{t_0})^{-\frac{1}{2}} \mathbb{E}[\mathbf{R}_i]\|_2 \|\check{\boldsymbol{\Sigma}}_{t_0}\|_{\mathrm{op}},
\end{aligned}
$$

where (i) follows from the Hölder inequality and (ii) holds due to the covariance normalization using $(\check{\boldsymbol{\Sigma}}_{t_0})^{-\frac{1}{2}}$. Applying Theorem 3.1.1 and Theorem 4.6.1 of Vershynin (2018) to

$$
(\check{\boldsymbol{\Sigma}}_{t_0})^{-1/2} (\bar{\mathbf{R}}_0 - \mathbb{E}[\mathbf{R}_i]) \quad \text{and} \quad \frac{1}{m-1} \sum_{i=1}^{m} (\check{\boldsymbol{\Sigma}}_{t_0})^{-1/2} (\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])(\mathbf{R}_i - \mathbb{E}[\mathbf{R}_i])^\top (\check{\boldsymbol{\Sigma}}_{t_0})^{-1/2},
$$

respectively, we obtain that with probability $1 - \delta$, it holds

$$
\begin{aligned}
(D) &= \mathcal{O}\left(\max\left\{ \frac{\sqrt{d} + \sqrt{\log(2/\delta)}}{\sqrt{m}}, \left(\frac{\sqrt{d} + \sqrt{\log(2/\delta)}}{\sqrt{m}} \right)^2 \right\} \cdot \left\| \check{\boldsymbol{\Sigma}}_{t_0} \right\|_{\mathrm{op}} \right) \\
&= \mathcal{O}\left(\lambda_{\max}(\boldsymbol{\Sigma}_0)(1 + \sigma_{\max}^k) d^{\frac{5}{4}} k^{\frac{k+10}{4}} n^{-\frac{1-\delta(n)}{k+5}} \log^{\frac{5}{2}} n \right),
\end{aligned} \tag{C.15}
$$

where the last inequality the last inequality invokes the order of m in (27) and the fact that

$$
\left\| \check{\boldsymbol{\Sigma}}_{t_0} \right\|_{\mathrm{op}} \leq \left\| \check{\boldsymbol{\Sigma}}_{t_0} - \widetilde{\boldsymbol{\Sigma}}_{t_0} \right\|_{\mathrm{op}} + \left\| \widetilde{\boldsymbol{\Sigma}}_{t_0} - \boldsymbol{\Sigma}_{t_0} \right\|_{\mathrm{op}} + \left\| \boldsymbol{\Sigma}_{t_0} - \boldsymbol{\Sigma}_0 \right\|_{\mathrm{op}} + \left\| \boldsymbol{\Sigma}_0 \right\|_{\mathrm{op}}.
$$

Summing up the upper bound of (A)–(D) in (C.8) and (C.13)–(C.15), we obtain the desired result. □

C.2.2 Proof of Lemma 3

Proof. For notational simplicity, we denote $\widehat{\mathbf{s}}_{T-t}(\cdot) := \widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\cdot, T-t)$ and let $\widehat{\sigma}_{\min}^2$ and $\widehat{\sigma}_{\max}^2$ be the minimal and maximal elements of $\mathbf{c}$ in $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$, respectively. First, by direct calculation, we obtain

$$
\begin{aligned}
\frac{\mathrm{d}\mathbb{E}\|\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow}\|_2^2}{\mathrm{d}t} &= 2\mathbb{E}\left[\left(\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow} \right)^\top \left(\frac{1}{2}\mathbf{R}_t^{\leftarrow} - \frac{1}{2}\widehat{\mathbf{R}}_t^{\leftarrow} + \nabla \log p_{T-t}\left(\mathbf{R}_t^{\leftarrow}\right) - \widehat{\mathbf{s}}_{T-t}(\widehat{\mathbf{R}}_t^{\leftarrow}) \right) \right] \\
&= \mathbb{E}\|\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow}\|_2^2 + 2 \underbrace{\mathbb{E}\left[(\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow})^\top (\nabla \log p_{T-t}\left(\mathbf{R}_t^{\leftarrow}\right) - \widehat{\mathbf{s}}_{T-t}(\widehat{\mathbf{R}}_t^{\leftarrow})) \right]}_{(*)}.
\end{aligned}
$$

Consider $\tilde{\mathbf{g}}_{\boldsymbol{\zeta}} : \mathbb{R}^k \times [0, T] \to \mathbb{R}^k$, equivalent to $\mathbf{g}_{\boldsymbol{\zeta}}$ defined via transformation, defined as

$$
\tilde{\mathbf{g}}_{\boldsymbol{\zeta}}(\mathbf{z}, t) := \mathbf{g}_{\boldsymbol{\zeta}}(\mathbf{V}^\top \mathbf{D}_t \mathbf{V} \mathbf{z}, t), \tag{C.16}
$$

where $\mathbf{V}$, $\mathbf{D}_t$, and $\mathbf{g}_{\boldsymbol{\zeta}}$ are components of $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}$ defined in (18). Note that the Lipschitz constant of the ReLU network $\tilde{\mathbf{g}}_{\boldsymbol{\zeta}}$ with respect to $\mathbf{z}$ is also on the order of γ defined in (22). Then, for term $(*)$,

we have

$$
\begin{aligned}
(*) &= \mathbb{E}\left[(\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow})^\top (\nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow}) - \widehat{\mathbf{s}}_{T-t}(\mathbf{R}_t^{\leftarrow}))\right] \\
&\quad + \mathbb{E}\left[(\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow})^\top (\widehat{\mathbf{s}}_{T-t}(\mathbf{R}_t^{\leftarrow}) - \widehat{\mathbf{s}}_{T-t}(\widehat{\mathbf{R}}_t^{\leftarrow}))\right] \\
&\overset{(i)}{\leq} \frac{\mathbb{E}\|\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow}\|_2^2}{4} + \mathbb{E}\left\|\widehat{\mathbf{s}}_{T-t}(\mathbf{R}_t^{\leftarrow}) - \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right\|_2^2 \\
&\quad + \mathbb{E}\left[(\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow})^\top \mathbf{D}_{T-t}^{1/2} (\alpha_{T-t}\gamma_1 \mathbf{D}_{T-t}^{1/2} \mathbf{V}(\mathbf{V}^\top \mathbf{D}_{T-t} \mathbf{V})^{-1} \mathbf{V}^\top \mathbf{D}_{T-t}^{1/2} - \mathbf{I}) \mathbf{D}_{T-t}^{1/2} (\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow})\right] \\
&\overset{(ii)}{\leq} \left(\frac{1}{4} + \frac{(\alpha_{T-t}\gamma_1 - 1)\mathbb{1}\{\alpha_{T-t}\gamma_1 > 1\}}{h_{T-t} + \widehat{\sigma}_{\min}^2 \alpha_{T-t}^2} + \frac{(\alpha_{T-t}\gamma_1 - 1)\mathbb{1}\{\alpha_{T-t}\gamma_1 \leq 1\}}{h_{T-t} + \widehat{\sigma}_{\max}^2 \alpha_{T-t}^2}\right) \mathbb{E}\|\mathbf{R}_t^{\leftarrow} - \widehat{\mathbf{R}}_t^{\leftarrow}\|_2^2 \\
&\quad + \mathbb{E}\left\|\widehat{\mathbf{s}}_{T-t}(\mathbf{R}_t^{\leftarrow}) - \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right\|_2^2,
\end{aligned}
$$

where (i) holds due to the Cauchy-Schwarz inequality and the fact that $\widehat{\mathbf{s}}_{T-t}(\cdot)$ is γ_1-Lipschitz; (ii) follows from

$$
\lambda_{\max}(\alpha_{T-t}\gamma_1 \mathbf{V}(\mathbf{V}^\top \mathbf{D}_{T-t}\mathbf{V})^{-1}\mathbf{V}^\top \mathbf{D}_{T-t} - \mathbf{I}) = \alpha_{T-t}\gamma_1 - 1,
$$

and

$$
\frac{1}{h_{T-t} + \widehat{\sigma}_{\max}^2 \alpha_{T-t}^2} \leq \|\widehat{\mathbf{D}}_{T-t}\|_{\mathrm{op}} \leq \frac{1}{h_{T-t} + \widehat{\sigma}_{\min}^2 \alpha_{T-t}^2}.
$$

Notice that $\alpha_{T-t}\gamma_1 \leq 1$ is equivalent to $t \leq T - 2\log\gamma_1$. By Grönwall's inequality, we obtain

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{R}_{T-t_0}^{\leftarrow} - \widehat{\mathbf{R}}_{T-t_0}^{\leftarrow}\|_2^2 \\
&\leq \left(\mathbb{E}\|\mathbf{R}_0^{\leftarrow} - \widehat{\mathbf{R}}_0^{\leftarrow}\|_2^2 + \int_0^{T-t_0} 2\mathbb{E}\left\|\widehat{\mathbf{s}}_{T-t}(\mathbf{R}_t^{\leftarrow}) - \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right\|_2^2 \mathrm{d}t\right) \\
&\quad \cdot \exp\left(\int_0^{T-2\log\gamma_1} \left(\frac{3}{2} + \frac{2(\alpha_{T-t}\gamma_1 - 1)}{h_{T-t} + \widehat{\sigma}_{\max}^2 \alpha_{T-t}^2}\right) \mathrm{d}t + \int_{T-2\log\gamma_1}^{T-t_0} \left(\frac{3}{2} + \frac{2(\alpha_{T-t}\gamma_1 - 1)}{h_{T-t} + \widehat{\sigma}_{\min}^2 \alpha_{T-t}^2}\right) \mathrm{d}t\right) \\
&= \left(\mathbb{E}\|\mathbf{R}_0^{\leftarrow} - \widehat{\mathbf{R}}_0^{\leftarrow}\|_2^2 + \int_0^{T-t_0} 2\mathbb{E}\left\|\widehat{\mathbf{s}}_{T-t}(\mathbf{R}_t^{\leftarrow}) - \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right\|_2^2 \mathrm{d}t\right) \qquad \text{(C.17)} \\
&\quad \cdot \exp\left(\frac{3}{2}(T - t_0) + \int_{t_0}^{2\log\gamma_1} \frac{2(\alpha_w\gamma_1 - 1)}{h_w + \widehat{\sigma}_{\min}^2 \alpha_w^2} \mathrm{d}w + \int_{2\log\gamma_1}^{T} \frac{2(\alpha_w\gamma_1 - 1)}{h_w + \widehat{\sigma}_{\max}^2 \alpha_w^2} \mathrm{d}w\right), \qquad \text{(C.18)}
\end{aligned}
$$

where the last equality follows from rearranging the terms and a change of variable $T - t = w$.

Now we claim that

$$
\int_{t_0}^{2\log\gamma_1} \frac{2(\alpha_w\gamma_1 - 1)}{h_w + \widehat{\sigma}_{\min}^2 \alpha_w^2} \mathrm{d}w + \int_{2\log\gamma_1}^{T} \frac{2(\alpha_w\gamma_1 - 1)}{h_w + \widehat{\sigma}_{\max}^2 \alpha_w^2} \mathrm{d}w \leq 4\gamma_1\left(1 - \log(\widehat{\sigma}_{\min}^2 + t_0)\right) - 2(T - t_0). \qquad \text{(C.19)}
$$

To verify this, consider the integral

$$\int \frac{\alpha_w\gamma_1 - 1}{h_w + c\alpha_w^2}\mathrm{d}w = \begin{cases} C - \dfrac{2\gamma_1 \arctan(\sqrt{c-1}e^{-w/2})}{\sqrt{c-1}} - \log(e^w + c - 1), & \forall\ c > 1 \quad (C.20) \\ C - 2\gamma_1 e^{-w/2} - w, & c = 1 \quad (C.21) \\ C - \dfrac{\gamma_1 \log\left(\frac{1+e^{-w/2}\sqrt{1-c}}{1-e^{-w/2}\sqrt{1-c}}\right)}{\sqrt{1-c}} - \log(e^w + c - 1), & \forall\ 0 < c < 1 \quad (C.22) \end{cases}$$

1. For the case $c > 1$, note that

$$\begin{aligned} &-\log\left(\frac{e^T + c - 1}{e^{t_0} + c - 1}\right) \le -(T - t_0) + \log(1 + (c-1)e^{-t_0}) \le \log(c - (c-1)t_0) - (T - t_0), \\ &-\frac{\arctan(\sqrt{c-1}e^{-T/2}) - \arctan(\sqrt{c-1}e^{-t_0/2})}{\sqrt{c-1}} \le e^{-t_0/2} - e^{-T/2} \le 1. \end{aligned} \quad (C.23)$$

   Then, by substituting (C.23) into the integral (C.20), we obtain

$$\int_{t_0}^{T} \frac{\alpha_w\gamma_1 - 1}{h_w + c\alpha_w^2}\mathrm{d}w \le 2\gamma_1 + \log(c - (c-1)t_0) - (T - t_0). \quad (C.24)$$

2. For the case $c = 1$, applying $e^{-w/2} \le 1$, we obtain that the integral in (C.21) satisfies

$$\int_{t_0}^{T} \frac{\alpha_w\gamma_1 - 1}{h_w + c\alpha_w^2}\mathrm{d}w \le -2\gamma_1(e^{-T} - e^{-t_0}) - (T - t_0) \le 2\gamma_1 - (T - t_0). \quad (C.25)$$

3. For the case $0 < c < 1$, due to the continuity of the integral (C.22) with respect to c and the bound in (C.25), we only need to focus on the case $c \ll 1$. Without loss of generality, we consider $c < 1/2$. By direct calculation, we have

$$\begin{aligned} &-\frac{1}{\sqrt{1-c}}\left(\log\left(\frac{1 + e^{-T/2}\sqrt{1-c}}{1 - e^{-T/2}\sqrt{1-c}}\right) - \log\left(\frac{1 + e^{-t_0/2}\sqrt{1-c}}{1 - e^{-t_0/2}\sqrt{1-c}}\right)\right) \\ &\le \frac{1}{\sqrt{1-c}}\log\left(\frac{1 + e^{-t_0/2}\sqrt{1-c}}{1 - e^{-t_0/2}\sqrt{1-c}}\right) \\ &\le \sqrt{2}(\log 4 - \log(c + t_0)), \end{aligned} \quad (C.26)$$

   where the last inequality follows from the fact that $e^{-x} \le 1/(1+x)$, $\sqrt{1-c} \le 1 - c/2$, and $\log((1+x)/(1-x))$ is increasing in x; and rearranging terms. Then, by substituting (C.23) and (C.26) into (C.22), we obtain

$$\int_{t_0}^{T} \frac{\alpha_w\gamma_1 - 1}{h_w + c\alpha_w^2}\mathrm{d}w \le 2\gamma_1(1 - \log(c + t_0)) - (T - t_0). \quad (C.27)$$

Combining the results in (C.24), (C.25), and (C.27), we verified the claim in (C.19). Finally,

applying the upper bound of score estimation in (29) and substituting (C.19) into (C.17) and (C.18), we deduce that

$$\begin{aligned}
&\mathbb{E}\|\mathbf{R}^{\leftarrow}_{T-t_0} - \widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}\|_2^2 \\
&\leq \left(\mathbb{E}\|\mathbf{R}^{\leftarrow}_{0} - \widehat{\mathbf{R}}^{\leftarrow}_{0}\|_2^2 + 2\epsilon(T-t_0)\right) \cdot \exp\left(\frac{3}{2}(T-t_0) + 4\gamma_1(1-\log(\widehat{\sigma}^2_{\min}+t_0)) - 2(T-t_0)\right) \\
&= \mathcal{O}\Big((1+\sigma^k_{\max})d^{\frac{5}{4}}k^{\frac{k+10}{4}}n^{-\frac{1-\delta(n)}{k+5}}\log^{\frac{5}{2}} n\Big),
\end{aligned}$$

where the last equality follows from invoking $\mathbb{E}\|\mathbf{R}^{\leftarrow}_{0} - \widehat{\mathbf{R}}^{\leftarrow}_{0}\|_2^2 = \mathcal{O}(e^{-T})$ and rearranging terms. □

C.2.3 Other Supporting Lemmas for Theorem 3

Lemma C.1. *Suppose that P_{data} is sub-Gaussian, and both $\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{r}, t)$ and $\nabla \log p_t(\mathbf{r})$ are Lipschitz with respect to both $\mathbf{r}$ and t. Consider the score estimation error satisfying*

$$\int_{t_0}^{T} \mathbb{E}_{\mathbf{R}_t \sim P_t} \left\|\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t, t) - \nabla \log p_t\left(\mathbf{R}_t\right)\right\|_2^2 \mathrm{d}t = \mathcal{O}(\epsilon(T-t_0)).$$

Then, the total variation distance is bounded by

$$\mathrm{TV}(P_{\text{data}}, \widehat{P}_{t_0}) = \mathcal{O}\left(dt_0 L_s(1+\sigma^2_{\max}) + \sqrt{\epsilon(T-t_0)} + \sqrt{\mathrm{KL}\left(P_{\text{data}} \| \mathcal{N}\left(\mathbf{0}, \mathbf{I}_d\right)\right)}\exp(-T)\right), \quad \text{(C.28)}$$

where P_{data} is the initial distribution of $\mathbf{R}$ in (8) and $\widehat{P}_{t_0}$ is the marginal distribution of the backward process $\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}$ in (4) starting from $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$.

Proof of Lemma C.1. To estimate $\mathrm{TV}(P_{\text{data}}, \widehat{P}_{t_0})$, we leverage the error decomposition in (25). Recall that $\widetilde{P}_{t_0}$ is the marginal distribution of $\widehat{\mathbf{R}}^{\leftarrow}_{T-t_0}$ in (4) initialized with $\widehat{\mathbf{R}}^{\leftarrow}_{0} \sim P_T$. In the decomposition (25), $\mathrm{TV}(P_{\text{data}}, P_{t_0})$ is the early stopping error, $\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0})$ is the statistical error arising from the score estimation, and $\mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0})$ is the mixing error of the forward process (1).

1. For term $\mathrm{TV}(P_{\text{data}}, P_{t_0})$, applying the upper bound (C.34) with $t = t_0$ in Lemma C.4, we obtain

$$\mathrm{TV}\left(P_{\text{data}}, P_{t_0}\right) = \mathcal{O}(dt_0). \quad \text{(C.29)}$$

2. For term $\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0})$, by Pinsker's inequality (Tsybakov 2009, Lemma 2.5) and the upper bound of KL-divergence (C.33) in Lemma C.3, we have

$$\mathrm{TV}(P_{t_0}, \widetilde{P}_{t_0}) \leq \mathrm{KL}(P_{t_0} || \widetilde{P}_{t_0}) = \mathcal{O}\big(\sqrt{\epsilon\left(T-t_0\right)}\big). \quad \text{(C.30)}$$

3. For term $\mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0})$, by Pinsker's inequality (Tsybakov 2009, Lemma 2.5) and Data pro-

cessing inequality (Thomas and Joy 2006, Theorem 2.8.1) , we deduce that

$$\mathrm{TV}(\widetilde{P}_{t_0}, \widehat{P}_{t_0}) \leq \sqrt{\mathrm{KL}(\widetilde{P}_{t_0} \| \widehat{P}_{t_0})} \leq \sqrt{\mathrm{KL}\left(P_T \| \mathcal{N}\left(0, \mathbf{I}_d\right)\right)} = \mathcal{O}\big(\sqrt{\mathrm{KL}\left(P_{\mathrm{data}} \| \mathcal{N}\left(0, \mathbf{I}_d\right)\right)} \exp(-T)\big), \tag{C.31}$$

where in the last inequality, we use the exponential mixing property of the O-U process.

Substituting the upper bounds (C.29), (C.30), and (C.31) into (25), we obtain the desired result. □

Lemma C.2 (Novikov's condition)**.** *Under the assumptions in Lemma C.1, it holds*

$$\mathbb{E}_{(\mathbf{R}_t^{\leftarrow})_{t\in[0,T-t_0]}}\left[\exp\left(\frac{1}{2}\int_0^{T-t_0} \|\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t^{\leftarrow}, t) - \nabla \log p_{T-t}\left(\mathbf{R}_t^{\leftarrow}\right)\|_2^2 \,\mathrm{d}t\right)\right] < \infty, \tag{C.32}$$

where the expectation is taken over the backward diffusion process $(\mathbf{R}_t^{\leftarrow})_{t\in[0,T-t_0]}$ *in* (3)*.*

Proof of Lemma C.2. The result follows from a straightforward calculation using the same techniques as in (Chen et al. 2023a, Lemma 11). □

Lemma C.3. *Suppose that the assumptions in Lemma C.1 hold. When both the ground-truth and the learned backward processes start with* $\mathbf{R}_0^{\leftarrow} \overset{d}{=} \widehat{\mathbf{R}}_0^{\leftarrow} \sim P_T$*, the KL-divergence between the laws of the terminal distributions of the processes* $\mathbf{R}_{T-t_0}^{\leftarrow}$ *and* $\widehat{\mathbf{R}}_{T-t_0}^{\leftarrow}$ *can be bounded by*

$$\mathrm{KL}(P_{t_0}||\widetilde{P}_{t_0}) \leq \mathbb{E}\left(\frac{1}{2}\int_0^{T-t_0} \|\widehat{\mathbf{s}}_{\boldsymbol{\theta}}\left(\mathbf{R}_t^{\leftarrow}, T-t\right) - \nabla \log p_{T-t}\left(\mathbf{R}_t^{\leftarrow}\right)\|_2^2 \,\mathrm{d}t\right) = \mathcal{O}\Big(\epsilon(T-t_0)\Big). \tag{C.33}$$

Proof of Lemma C.3. By Lemma C.2, the Novikov's condition holds. By the data processing inequality of the KL divergence (Thomas and Joy 2006, Theorem 2.8.1), we have

$$\begin{aligned}\mathrm{KL}\left(P_{t_0}\middle\|\widetilde{P}_{t_0}\right) &\leq \mathrm{KL}\left((\mathbf{R}_t^{\leftarrow})_{0\leq t\leq T-t_0}||(\widehat{\mathbf{R}}_t^{\leftarrow})_{0\leq t\leq T-t_0}\right) \\ &= \frac{1}{2}\mathbb{E}\left[\int_0^{T-t_0} \left\|\widehat{\mathbf{s}}_{\boldsymbol{\theta}}(\mathbf{R}_t^{\leftarrow}, T-t) - \nabla \log p_{T-t}(\mathbf{R}_t^{\leftarrow})\right\|_2^2 \mathrm{d}t\right].\end{aligned}$$

Immediately, we obtain the results directly using Girsanov's Theorem (Chen et al. 2023b, Theorem 9). □

Lemma C.4. *Suppose that the assumptions in Lemma C.1 hold. Then, for any* $t < 1/d$*, we have*

$$\mathrm{TV}(P_{data}, P_t) = \mathcal{O}(dt). \tag{C.34}$$

Proof of Lemma C.4. Given $\mathbf{R}_0$, $\mathbf{R}_t$ can be represented as

$$\mathbf{R}_t = e^{-t/2}\mathbf{R}_0 + \int_0^t e^{-(t-s)/2}\mathrm{d}\mathbf{W}_s,$$

where $\mathbf{R}_0$ and $\int_0^t e^{-(t-s)/2}\mathrm{d}\mathbf{W}_s$ are independent. Then, the density of $\mathbf{R}_t$ is given by

$$p_t(\mathbf{r}) = \int p_{\text{data}}(\mathbf{y})\phi(\mathbf{r}; \alpha_t\mathbf{y}, h_t)\mathrm{d}\mathbf{y}.$$

Define

$$S(d,t) := \mathcal{O}(\sqrt{d + \log(1/t)}) \tag{C.35}$$

as a truncation radius and we have

$$\begin{aligned}
\text{TV}\left(P_{\text{data}}, P_t\right) &= \frac{1}{2}\int |p_t(\mathbf{r}) - p_{\text{data}}(\mathbf{r})|\,\mathrm{d}\mathbf{r} \\
&\le \frac{1}{2}\int_{\|\mathbf{r}\|_2 > S(d,t)} \Big(p_t(\mathbf{r}) + p_{\text{data}}(\mathbf{r})\Big)\mathrm{d}\mathbf{r} \qquad \text{(C.36)} \\
&\quad + \frac{1}{2}\int_{\|\mathbf{r}\|_2 \le S(d,t)} \left| \int (p_{\text{data}}(\mathbf{y})\phi(\mathbf{r}; \alpha_t\mathbf{y}, h_t) - p_{\text{data}}(\mathbf{r}))\mathrm{d}\mathbf{y}\right|\mathrm{d}\mathbf{r}. \qquad \text{(C.37)}
\end{aligned}$$

By the density upper bound in (B.18) and Theorem 3.1 of Chazottes, Collet, and Redig (2021), it holds that p_{data} and $p_t(\mathbf{r})$ are sub-Gaussian and there exists a constant $A_1 > 0$ such that $(p_t(\mathbf{r}) + p_{\text{data}}(\mathbf{r})) \le \exp(-A_1\|\mathbf{r}\|_2^2/2)$.

For term (C.36), using the sub-Gaussian tail in Proposition 2.6.6 of Vershynin (2018) and invoking the order of $S(d,t)$ in (C.35), we obtain that

$$\text{(C.36)} = \mathcal{O}\left(\frac{2^{-\frac{d}{2}} d S(d,t)^{d-2}}{A_1\Gamma\left(\frac{d}{2}+1\right)} \exp\left(-\frac{A_1 S(d,t)^2}{2}\right)\right) = \mathcal{O}\big(t\exp\left(-A_1 d\right)\big). \tag{C.38}$$

For term (C.37), by taking a change of variable $\mathbf{z} := (\mathbf{r} - \alpha_t\mathbf{y})/\sqrt{h_t}$, we deduce that

$$\begin{aligned}
\text{(C.37)} &= \int_{\|\mathbf{r}\|_2 \le S(d,t)} \left| \int \left(p_{\text{data}}(\alpha_t^{-1}(\mathbf{r} - \sqrt{h_t}\mathbf{z}))\phi(\mathbf{z}; \mathbf{0}, \mathbf{I}) - p_{\text{data}}(\mathbf{r})\right)\mathrm{d}\mathbf{z}\right|\mathrm{d}\mathbf{r} \\
&\overset{(i)}{=} \mathcal{O}\bigg(\int_{\|\mathbf{r}\|_2 \le S(d,t)} \left| \int \left(\nabla p_{\text{data}}(\mathbf{r})\left(\frac{t\mathbf{r}}{2} - \sqrt{t}\mathbf{z}\right)\phi(\mathbf{z}; \mathbf{0}, \mathbf{I})\mathrm{d}\mathbf{z}\right|\mathrm{d}\mathbf{r} \qquad \text{(C.39)} \\
&\quad + \int_{\|\mathbf{r}\|_2 \le S(d,t)} \left|\frac{1}{2}\left(\frac{t\mathbf{r}}{2} - \sqrt{t}\mathbf{z}\right)^\top \nabla^2 p_{\text{data}}(\mathbf{r})\left(\frac{t\mathbf{r}}{2} - \sqrt{t}\mathbf{z}\right)\right)\phi(\mathbf{z}; \mathbf{0}, \mathbf{I})\mathrm{d}\mathbf{z}\right|\mathrm{d}\mathbf{r}\bigg), \qquad \text{(C.40)}
\end{aligned}$$

where (i) involves the Taylor expansion $\alpha_t^{-1} = 1 + t/2 + \mathcal{O}(t^2)$, $h_t = t + \mathcal{O}(t^2)$ and the fact that

$$p_{\text{data}}(e^{t/2}(\mathbf{r} - \sqrt{h_t}\mathbf{z})) = p_{\text{data}}(\mathbf{r}) + \nabla p_{\text{data}}(\mathbf{r})\left(\frac{t\mathbf{r}}{2} - \sqrt{t}\mathbf{z}\right) + \frac{1}{2}\left(\frac{t\mathbf{r}}{2} - \sqrt{t}\mathbf{z}\right)^\top \nabla^2 p_{\text{data}}(\mathbf{r})\left(\frac{t\mathbf{r}}{2} - \sqrt{t}\mathbf{z}\right).$$

The integrals associated with the kernel function $\phi(\mathbf{z}; \mathbf{0}, \mathbf{I})$ satisfy

$$\int p_{\text{data}}(\mathbf{r})\phi(\mathbf{z}; \mathbf{0}, \mathbf{I})\mathrm{d}\mathbf{z} = p_{\text{data}}(\mathbf{r}) \tag{C.41}$$

and

$$\int \nabla p_{\text{data}}(\mathbf{r})\left(\frac{t\mathbf{r}}{2}-\sqrt{t}\mathbf{z}\right)\phi(\mathbf{z};\mathbf{0},\mathbf{I})\mathrm{d}\mathbf{z} = \frac{t}{2}\nabla \log p_{\text{data}}(\mathbf{r})\mathbf{r}p_{\text{data}}(\mathbf{r}) = \mathcal{O}(t\|\mathbf{r}\|_2\|\nabla \log p_{\text{data}}(\mathbf{r})\|_2 \cdot p_{\text{data}}(\mathbf{r})). \tag{C.42}$$

Moreover, since the Hessian matrix satisfies the following property

$$\nabla^2 p_{\text{data}}(\mathbf{r}) = (\nabla^2 \log p_{\text{data}}(\mathbf{r}) + \nabla \log p_{\text{data}}(\mathbf{r})\nabla \log p_{\text{data}}(\mathbf{r})^\top) \cdot p_{\text{data}}(\mathbf{r}), \tag{C.43}$$

we deduce

$$\begin{aligned}
&\int \left(\frac{1}{2}\left(\frac{t\mathbf{r}}{2}-\sqrt{t}\mathbf{z}\right)^\top \nabla^2 p_{\text{data}}(\mathbf{r})\left(\frac{t\mathbf{r}}{2}-\sqrt{t}\mathbf{z}\right)\right)\phi(\mathbf{z};\mathbf{0},\mathbf{I})\mathrm{d}\mathbf{z} \\
&\overset{(i)}{=} \operatorname{tr}\left(\frac{1}{8}t^2\nabla^2 p_{\text{data}}(\mathbf{r})\mathbf{r}\mathbf{r}^\top + \frac{1}{2}t\nabla^2 p_{\text{data}}(\mathbf{r})\right) \\
&\overset{(ii)}{=} \operatorname{tr}\left(\left(\frac{1}{8}t^2\mathbf{r}\mathbf{r}^\top + \frac{1}{2}t\right)(\nabla^2 \log p_{\text{data}}(\mathbf{r}) + \nabla \log p_{\text{data}}(\mathbf{r})\nabla \log p_{\text{data}}(\mathbf{r})^\top) \cdot p_{\text{data}}(\mathbf{r})\right) \\
&= \mathcal{O}\left((t^2\|\mathbf{r}\|_2^2 + t)\operatorname{tr}\left((\nabla^2 \log p_{\text{data}}(\mathbf{r}) + \nabla \log p_{\text{data}}(\mathbf{r})\nabla \log p_{\text{data}}(\mathbf{r})^\top) \cdot p_{\text{data}}(\mathbf{r})\right)\right).
\end{aligned} \tag{C.44}$$

where (i) is follows from $\int \mathbf{z}\mathbf{z}^\top\phi(\mathbf{z};\mathbf{0},\mathbf{I})\mathrm{d}\mathbf{z} = \mathbf{I}_d$ and rearranging terms, and (ii) follows (C.43).

Therefore, by substituting (C.41), (C.42) and (C.44) into (C.39) and (C.40), we obtain that

$$\begin{aligned}
(\text{C.37}) &= \mathcal{O}\Bigg(\int_{\|\mathbf{r}\|_2 \le S(d,t)} \Big(t\|\mathbf{r}\|_2\|\nabla \log p_{\text{data}}(\mathbf{r})\|_2 \\
&\qquad + \operatorname{tr}\left((t^2\|\mathbf{r}\|_2^2 + t)(\nabla^2 \log p_{\text{data}}(\mathbf{r}) + \nabla \log p_{\text{data}}(\mathbf{r})\nabla \log p_{\text{data}}(\mathbf{r})^\top)\right)\Big)p_{\text{data}}(\mathbf{r})\mathrm{d}\mathbf{r}\Bigg) \\
&\overset{(i)}{=} \mathcal{O}\Bigg(tS(d,t)\sqrt{\mathbb{E}_{\mathbf{R}_0\sim P_{\text{data}}}[\|\nabla \log p_{\text{data}}(\mathbf{R}_0)\|_2^2]} \\
&\qquad + (t^2S^2(d,t) + t)\operatorname{tr}\left(\int (\nabla^2 \log p_{\text{data}}(\mathbf{r}) + \nabla \log p_{\text{data}}(\mathbf{r})\nabla \log p_{\text{data}}(\mathbf{r})^\top)p_{\text{data}}(\mathbf{r})\mathrm{d}\mathbf{r}\right)\Bigg) \\
&\overset{(ii)}{=} \mathcal{O}\big(t\sqrt{d}S(d,t) + t^2dS^2(d,t) \cdot L_s(\sigma_{\max}^2 + 1)\big) = \mathcal{O}\big(dtL_s(\sigma_{\max}^2 + 1)\big).
\end{aligned} \tag{C.45}$$

where (i) is due to the Cauchy-Schwarz inequality and $\|\mathbf{r}\|_2 \le S(d,t)$, and (ii) invokes the upper bound (C.46) in Lemma C.5.

Combining the upper bound of (C.36) and (C.37) in (C.38) and (C.45), we obtain the desired result. □

Lemma C.5. *Suppose Assumptions 1-3 holds. Then, it holds*

$$\mathbb{E}_{\mathbf{R}_0\sim P_{\text{data}}}\|\nabla \log p_{\text{data}}(\mathbf{R}_0)\|_2^2 = \mathcal{O}(dL_s(\sigma_{\max}^2 + 1)). \tag{C.46}$$

Proof of Lemma C.5. Taking $t = 0$ in the formula of $\nabla \log p_t$ in (12) of Lemma 1, we have

$$\nabla \log p_{\text{data}}(\mathbf{r}) = \mathbf{s}_{\text{sub}}(\boldsymbol{\Gamma}_0 \boldsymbol{\beta}^\top \boldsymbol{\Lambda}_0^{-1} \mathbf{r}, 0) - \boldsymbol{\Lambda}_0^{-\frac{1}{2}} (\mathbf{I} - \boldsymbol{\Lambda}_0^{-\frac{1}{2}} \boldsymbol{\beta} \boldsymbol{\Gamma}_0 \boldsymbol{\beta}^\top \boldsymbol{\Lambda}_0^{-\frac{1}{2}}) \boldsymbol{\Lambda}_0^{-\frac{1}{2}} \mathbf{r}.$$

Under Assumption 3, for any $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^d$, it holds that

$$\begin{aligned} \|\nabla \log p_{\text{data}}(\mathbf{r}_1) - \nabla \log p_{\text{data}}(\mathbf{r}_2)\|_2 &\le L_s \|\boldsymbol{\Gamma}_0 \boldsymbol{\beta}^\top \boldsymbol{\Lambda}_0^{-1}\|_{\text{op}} \|\mathbf{r}_1 - \mathbf{r}_2\|_2 + \|\boldsymbol{\Lambda}_0^{-1}\|_{\text{op}} \|\mathbf{r}_1 - \mathbf{r}_2\|_2 \\ &\le \frac{L_s(\sigma_{\max}^2 + 1)}{\sigma_d^2} \cdot \|\mathbf{r}_1 - \mathbf{r}_2\|_2, \end{aligned} \tag{C.47}$$

where the last equality follows from $\|\boldsymbol{\Gamma}_0\|_{\text{op}} \le \sigma_{\max}^2$ and $\|\boldsymbol{\Lambda}_0^{-1}\|_{\text{op}} \le 1/\sigma_d^2$. This indicates that the Lipschitz constant of $\nabla \log p_{\text{data}}$ is bounded by $L_s(1 + \sigma_{\max}^2)/\sigma_d^2$. Furthermore, we have

$$\begin{aligned} \mathbb{E}_{\mathbf{R}_0 \sim P_{\text{data}}} \left[\|\nabla \log p_{\text{data}}(\mathbf{R}_0)\|_2^2 \right] &= \text{tr} \left(\int \nabla \log p_{\text{data}}(\mathbf{r}) \nabla \log p_{\text{data}}(\mathbf{r})^\top p_{\text{data}}(\mathbf{r}) \mathrm{d}\mathbf{r} \right) \\ &\overset{(i)}{=} \text{tr} \left(- \int \nabla^2 \log p_{\text{data}}(\mathbf{r})^\top p_{\text{data}}(\mathbf{r}) \mathrm{d}\mathbf{r} \right) \\ &= \mathcal{O}(d L_s(\sigma_{\max}^2 + 1)), \end{aligned}$$

where (i) is due to the integration by parts and the last inequality follows from invoking (C.47). □

D Additional Details of the Numerical Study with Synthetic Data

Here we explain additional details of the numerical experiment setup for Section 6. Following the standard setup in the econometrics literature (Bai and Ng 2002, 2023), we construct the ground-truth environment of high-dimensional asset returns using a sub-Gaussian factor model. Specifically, the universe consists of $d = 2048$ assets, whose returns are driven by $k = 16$ latent factors. Here, the choice of d as a power of 2 enhances the computational efficiency.

Denote $\boldsymbol{\mu}_F = (\mu_{F1}, \mu_{F2}, \ldots, \mu_{Fk})$ as the expected return and $\boldsymbol{\Sigma}_F = \text{diag}\{\sigma_{F1}^2, \sigma_{F2}^2, \ldots, \sigma_{Fk}^2\}$ the covariance matrix of the latent factor. In addition, denote $\boldsymbol{\Sigma}_\varepsilon = \text{diag}\{\sigma_{\varepsilon 1}^2, \sigma_{\varepsilon 2}^2, \ldots, \sigma_{\varepsilon d}^2\}$ as the covariance of the idiosyncratic noise of the asset. We then construct samples from the ground-truth environment as follows:

1. **Latent Factor.** The components of $\boldsymbol{\mu}_F$ are drawn i.i.d. from Uniform$(0, 0.1)$ and we set $\sigma_{Fi} = 1.5\mu_{Fi}$ for $i = 1, 2, \ldots, k$ to ensure that the volatility scales proportionally to the corresponding mean.

2. **Factor Loadings.** We generate the factor loading matrix $\boldsymbol{\beta} \in \mathbb{R}^{d \times k}$, where each element is drawn i.i.d. from $\mathcal{N}(0, 1)$, ensuring that the loadings are symmetrically distributed with comparable magnitudes across assets and factors.

3. **Idiosyncratic Risk.** $\{\sigma_{\varepsilon i}\}_{i=1}^d$ are drawn i.i.d. from Uniform$(0, 0.4)$, ensuring uncorrelated idiosyncratic returns across assets.

4. **Asset Return.** We generate a total of $2^{13} = 8192$ simulated samples. Asset returns are sampled i.i.d. according to the following procedure. First, the factor is drawn from a multivariate normal distribution $\mathbf{F} \sim \mathcal{N}(\boldsymbol{\mu}_F, \boldsymbol{\Sigma}_F)$. Then the asset-specific noise terms are drawn i.i.d. from $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\varepsilon)$. Finally, the asset return is constructed by $\mathbf{R} = \boldsymbol{\beta}\mathbf{F} + \boldsymbol{\varepsilon}$. We denote by μ_{Ri} and σ_{Ri} the mean and standard deviation of the ground-truth return for asset i, where $i = 1, 2, \ldots, d$.

Summary Statistics of the Synthetic Data. To show that our simulation setting is close to the realistic market scenario, we benchmark our simulation set-up against the S&P 500 index. Specifically, denote by $\mu_{\text{S\&P 500},i}$ and $\sigma_{\text{S\&P500},i}$ the mean and standard deviation of historical returns for stock i in the S&P 500 index over the period 2000–2020. Table D.1 reports the summary statistics of $\{\mu_{Ri}\}_{i=1}^{d}$ and compares them with $\{\mu_{\text{S\&P500},i}\}_{i=1}^{500}$. The range of both the simulated mean and standard deviation of returns closely matches that of the empirical quantities of stocks in the S&P 500 index.

In addition, the variance of the factors accounts for 50.42% of the total variance in our synthetic data, which corresponds to the population R-squared.

Table D.1: Summary statistics for simulation return data and comparison with S&P 500 over the period 2000-2020.

	Mean	**Std**	**Min**	**25%**	**50%**	**75%**	**Max**
Synthetic $\{\mu_{Ri}\}$	0.000	0.235	-0.809	-0.154	-0.007	0.155	0.751
S&P 500 $\{\mu_{\text{S\&P500},i}\}$	0.070	0.234	-0.817	-0.057	-0.124	0.182	0.929
Synthetic $\{\sigma_{Ri}\}$	0.475	0.126	0.243	0.377	0.473	0.576	0.739
S&P 500 $\{\sigma_{\text{S\&P500},i}\}$	0.380	0.142	0.203	0.273	0.345	0.450	0.725

Data Preprocessing. We preprocess the data in the following steps.

1. First, we sort the asset returns by their variance, prioritizing those with greater variability for subsequent analysis.

2. Next, we normalize the data by subtracting the mean return of each asset and reshape the data from a one-dimensional vector of length 2^{11} into a two-dimensional matrix of size $(2^5, 2^6)$. This reshaping step ensures compatibility with the 2D-Unet architecture and allows the model to effectively leverage spatial hierarchies in the data.

Training. We train our diffusion factor model using a 2-dimensional U-Net architecture (Ronneberger, Fischer, and Brox 2015), which is a convolutional encoder-decoder network with skip connections. The U-Net serves as a practical implementation of the encoder-decoder architecture analyzed in theory. Specifically, in our theoretical analysis, we consider a linear encoder-decoder architecture, which first projects the input data into a low-dimensional latent space and then reconstructs it back to the original space. This *project-then-lift network architecture* not only reduces

the number of trainable parameters by operating in a low-dimensional latent space, but also effectively captures data intrinsic structures such as the factor model in Assumption 1. In experiments, U-Net implements the same encoder-decoder architecture with nonlinear networks, which is further shown to accommodate complex real-world data, such as the U.S. stock returns in Section 7 and Appendix E. Nonetheless, whether the mappings are linear or nonlinear is *secondary*, since linear transformations are only assumed to enable tractable analysis and are standard in the theoretical study of deep learning (Baldi and Hornik 1989, Chen et al. 2012, 2023a, Weitzner et al. 2025).

Besides, guided by the theoretical results in Sections 3-5, we choose the latent space dimension—the bottleneck dimension of the encoder and decoder architecture—based on the underlying factor model. More specifically, in synthetic experiments, we set the bottleneck width equal to the true factor dimension, $k = 16$. The U-Net has approximately one billion parameters and is trained to approximate the score function by minimizing the empirical loss defined in (7). To assess performance under different data regimes, we set the number of training samples to be $N = 2^9, 2^{10}, \ldots, 2^{13}$. For fairness, both `Diff Method` and `Emp Method` are trained on the *same* N simulated returns for each N. The `Diff Method` then uses the trained model to generate 2^{13} new samples for latent subspace recovery and distribution estimation, while the `Emp Method` provides empirical estimates solely on N training samples. To ensure that the results are robust rather than due to chance, each experiment is repeated five times.

E Additional Details of the Empirical Analysis

In this section, we provide further details on the empirical setup in Section 7 and report robustness analysis with respect to transaction costs, risk aversions, norm constraints, and model update frequency.

E.1 Data Preprocessing, Training, and Evaluation

Data Selection and Preprocessing. We select and preprocess the stock return data in the following steps:

1. We first exclude stocks with more than 5% missing values and then select the 512 stocks with the largest market capitalizations from the remaining universe.

2. Rank the selected stocks by return volatility in descending order.

3. Within each rolling window of the training data, we standardize the returns by subtracting the (empirical) mean and dividing by the (empirical) standard deviation for each stock.

4. Winsorize returns for each stock at 2.5% each side by resampling non-extreme values with the same sign, which preserves the empirical distribution while mitigating the influence of outliers (Tukey 1962).

Training and Sampling. We employ a 2D-UNet architecture with approximately one billion parameters to train our diffusion factor model. For real data, the true factor dimension is unknown; following common practice, we set the bottleneck width to 8 as an educated estimate consistent with factor dimensions commonly used in the literature (Fama and French 1993, Bai and Ng 2002, Onatski 2010, Fama and French 2015b, Fan, Guo, and Zheng 2022, Bai and Ng 2023). Following a similar setup as Lyu et al. (2022), we set the total number of training steps to $T = 200$ and apply early stopping at $T' = 180$ for the sampling of time-reversed process (4). For the downstream evaluation on mean-variance portfolios and factor-tangency portfolios, we use the trained model to generate $2^{12} = 4096$ new samples for each rolling window.

Performance Evaluations. Next, we specify the performance evaluation metrics used in Section 7.

1. SR is defined as $\hat{\mu}/\hat{\sigma}$, where $\hat{\mu}$ and $\hat{\sigma}$ denote the sample mean and standard deviation, respectively, of excess portfolio returns over the testing periods.

2. CER is defined as $\hat{\mu} - \frac{1}{2}\eta\hat{\sigma}^2$, where η is the risk aversion parameter.

3. MDD is defined as
$$\text{MDD} = \max_{t\in\mathcal{D}_t}\left(\frac{\max_{s\le t} V_s - V_t}{\max_{s\le t} V_s}\right),$$
where $\mathcal{D}_t$ contains all the dates of the test set and V_t denotes the portfolio value on day t.

4. TO on day t is defined as
$$\text{TO}_t = \sum_{i\in\mathcal{A}_t}\left|w_{i,t} - \frac{w_{i,t-1}(1+r_{i,t-1})}{\sum_{i=1}^{N} w_{i,t-1}(1+r_{i,t-1})}\right|,$$
where $\mathcal{A}_t$ contains all assets of the test set on day t, $w_{i,t}$ is the target weight of stock i on day t, and $r_{i,t}$ denotes the return of stock i on day t.

We visualize the return distribution generated by our diffusion factor model for selected assets in Figure E.1 (trained on data from May 1, 2009 to April 30, 2014), which is compared with the observed training data. The generated data distribution is smoother and closely approximates the empirical distribution.

E.2 Robustness Analysis on Transaction Costs and Risk Aversion

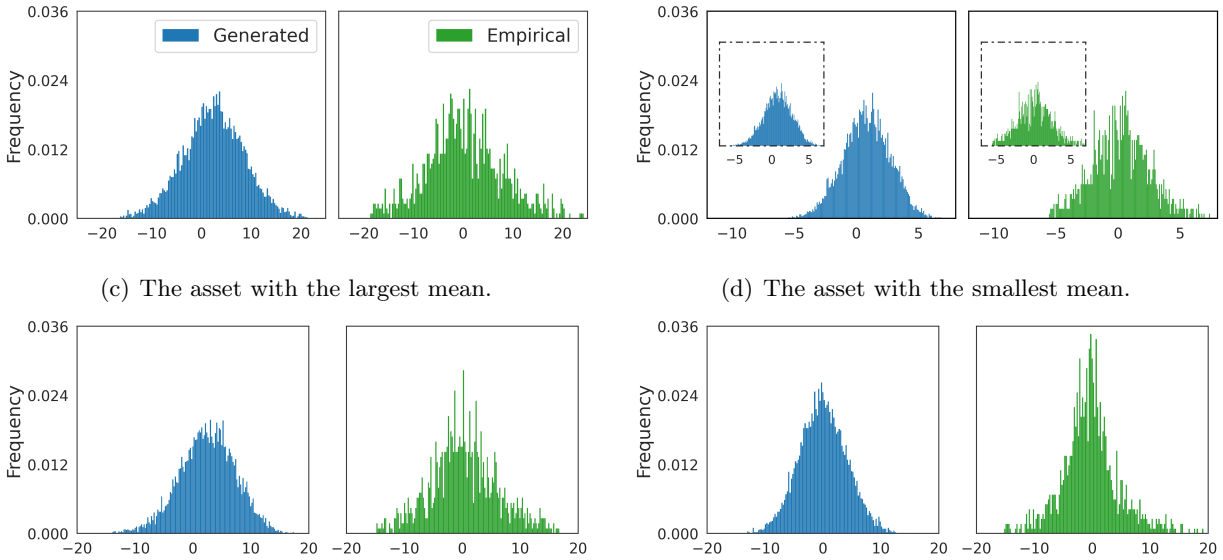
For mean-variance portfolios with $\eta = 3$, we report out-of-sample portfolio performance under the scenario without transaction costs in Table E.1. The `Diff Emp+Diff Emp` outperforms all other methods, which are consistent with those observed in the scenario with transaction costs.

For the case of $\eta = 5$, we report out-of-sample portfolio performance under scenarios with and without transaction costs in Table E.2 and plot the cumulative returns with transaction cost in

Figure E.1: Examples of asset return distribution (the blue histogram is constructed using samples generated from the diffusion model and the green one uses actual data samples.)

(a) The asset with the largest variance.

(b) The asset with the smallest variance.



(c) The asset with the largest mean.

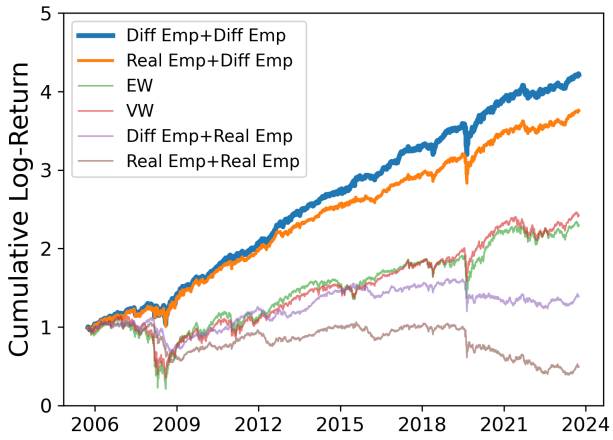
(d) The asset with the smallest mean.

Table E.1: Performance of different portfolios without transaction costs for $\eta = 3$ (model updated quarterly).

Method	Mean	Std	SR	CER	MDD (%)	TO
Methods based on real observed data						
`EW`	0.106	0.206	0.516	0.043	52.437	**3.031**
`VW`	0.103	0.220	0.468	0.031	57.322	3.464
`Real Emp+Real Emp`	0.077	0.126	0.608	0.053	33.642	46.722
`Real BS+Real Emp`	0.070	0.124	0.565	0.047	32.092	45.612
`Real OLSE+Real Emp`	0.053	0.125	0.427	0.030	33.188	45.952
`Real Emp+Real LW`	0.075	0.121	0.617	0.053	32.264	38.827
`Real BS+Real LW`	0.069	**0.119**	0.575	0.047	31.558	37.900
`Real OLSE+Real LW`	0.053	0.120	0.438	0.031	33.503	38.543
Methods based on diffusion-generated data						
`Diff Emp+Diff Emp`	**0.273**	0.157	**1.740**	**0.236**	32.159	28.751
`Diff BS+Diff Emp`	0.269	0.155	1.729	0.232	32.113	27.978
`Diff OLSE+Diff Emp`	0.268	0.155	1.728	0.232	32.110	27.876
`Diff Emp+Diff LW`	0.233	0.150	1.547	0.199	32.048	26.353
`Diff BS+Diff LW`	0.230	0.149	1.539	0.196	31.995	25.773
`Diff OLSE+Diff LW`	0.229	0.149	1.539	0.196	31.991	25.697
Methods based on both real observed data and diffusion-generated data						
`Real Emp+Diff Emp`	0.210	0.149	1.410	0.177	**30.098**	23.313
`Diff Emp+Real Emp`	0.095	0.133	0.720	0.069	33.777	29.323

Figure E.2. The `Diff Emp+Diff Emp` outperforms all other methods with the highest Mean, SR, and CER. These results are consistent with those observed in the case of $\eta = 3$.

Figure E.2: Cumulative returns of different portfolios in log scale with transaction cost for $\eta = 5$ (model updated quarterly).



E.3 Robustness Analysis on Norm Constraints

As a robustness check on the choice of norm constraints, for mean-variance portfolios, we solve the target weight by replacing the ℓ_∞-norm constraint in (34) with an ℓ_1-norm constraint $\|\boldsymbol{\omega}\|_1 = \sum_{i=1}^{d} |w_i| \leq 3$. We report the out-of-sample portfolio performance with and without transaction costs in Table E.3 and plot the cumulative returns with transaction costs (in log scale) in Figure E.3. The `Diff Emp+Diff Emp` achieves the highest Mean, SR, and CER, which is similar to the results under the ℓ_∞-norm constraint.

Figure E.3: Cumulative returns of different portfolios in log scale with transaction cost under ℓ_1 norm constraints.

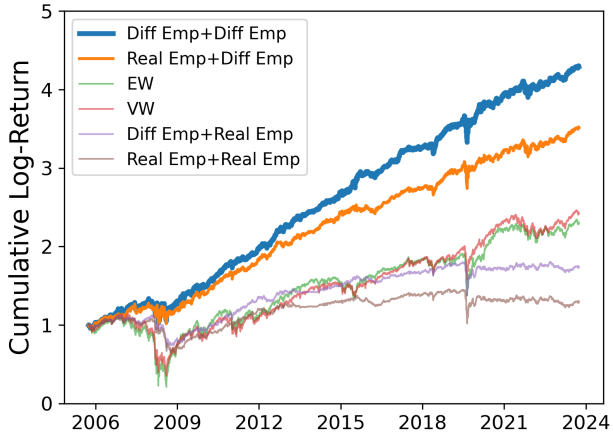


Table E.2: Performance of different portfolios with and without transaction costs for $\eta = 5$.

Method	Mean	Std	SR	CER	MDD (%)	TO
Panel A: Without Transaction Costs						
Methods based on real observed data						
EW	0.106	0.206	0.516	0.000	52.437	**3.031**
VW	0.103	0.220	0.468	-0.018	57.322	3.464
Real Emp+Real Emp	0.071	0.124	0.573	0.033	32.477	45.764
Real BS+Real Emp	0.067	0.123	0.546	0.029	31.390	45.359
Real OLSE+Real Emp	0.057	0.124	0.462	0.019	32.432	45.405
Real Emp+Real LW	0.070	0.120	0.584	0.034	31.503	38.025
Real BS+Real LW	0.066	**0.119**	0.556	0.031	31.657	37.706
Real OLSE+Real LW	0.057	0.120	0.438	0.021	32.850	38.004
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.234**	0.146	**1.607**	**0.181**	31.836	22.226
Diff BS+Diff Emp	0.232	0.145	1.597	0.179	31.804	21.881
Diff OLSE+Diff Emp	0.231	0.145	1.595	0.179	31.802	21.834
Diff Emp+Diff LW	0.207	0.142	1.452	0.156	32.409	21.410
Diff BS+Diff LW	0.205	0.142	1.445	0.155	32.442	21.176
Diff OLSE+Diff LW	0.205	0.142	1.444	0.155	32.456	21.144
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.203	0.141	1.153	0.154	**30.194**	20.049
Diff Emp+Real Emp	0.082	0.127	0.643	0.041	32.531	25.921
Panel B: With Transaction Costs						
Methods based on real observed data						
EW	0.100	0.206	0.486	-0.006	53.128	**3.031**
VW	0.096	0.220	0.437	-0.025	58.086	3.464
Real Emp+Real Emp	-0.020	0.126	-0.160	-0.060	45.671	45.764
Real BS+Real Emp	-0.023	0.125	-0.186	-0.063	46.280	45.359
Real OLSE+Real Emp	-0.034	0.126	-0.267	-0.073	51.562	45.405
Real Emp+Real LW	-0.006	0.121	-0.051	-0.043	39.176	38.025
Real BS+Real LW	-0.009	**0.120**	-0.077	-0.046	39.598	37.706
Real OLSE+Real LW	-0.019	0.121	-0.160	-0.056	43.379	38.004
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.190**	0.147	**1.292**	**0.136**	32.459	22.226
Diff BS+Diff Emp	0.188	0.146	1.285	0.134	32.532	21.881
Diff OLSE+Diff Emp	0.188	0.146	1.284	0.134	32.551	21.834
Diff Emp+Diff LW	0.164	0.144	1.143	0.113	33.410	21.410
Diff BS+Diff LW	0.163	0.143	1.138	0.112	33.435	21.176
Diff OLSE+Diff LW	0.163	0.143	1.138	0.112	33.448	21.144
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.163	0.142	1.153	0.113	**31.111**	20.049
Diff Emp+Real Emp	0.030	0.128	0.234	-0.011	35.536	25.921

Table E.3: Performance of different portfolios with and without transaction costs under ℓ_1-norm constraints.

Method	Mean	Std	SR	CER	MDD (%)	TO
Panel A: Without Transaction Costs						
Methods based on real observed data						
EW	0.106	0.206	0.516	0.043	52.437	**3.031**
VW	0.103	0.220	0.468	0.031	57.322	3.464
Real Emp+Real Emp	0.056	0.114	0.487	0.036	34.460	16.514
Real BS+Real Emp	0.051	0.114	0.446	0.031	34.637	16.410
Real OLSE+Real Emp	0.040	0.114	0.354	0.021	35.004	16.418
Real Emp+Real LW	0.057	0.114	0.501	0.038	34.369	16.250
Real BS+Real LW	0.052	**0.113**	0.456	0.032	34.534	15.968
Real OLSE+Real LW	0.042	0.114	0.370	0.023	34.863	15.952
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.239**	0.164	**1.454**	**0.199**	28.627	21.141
Diff BS+Diff Emp	0.234	0.163	1.436	0.194	29.512	20.790
Diff OLSE+Diff Emp	0.234	0.163	1.434	0.194	29.663	20.731
Diff Emp+Diff LW	0.206	0.160	1.289	0.168	31.520	20.343
Diff BS+Diff LW	0.202	0.159	1.274	0.164	32.292	20.081
Diff OLSE+Diff LW	0.202	0.159	1.273	0.164	32.401	20.038
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.191	0.158	1.208	0.154	**28.414**	19.440
Diff Emp+Real Emp	0.066	0.115	0.574	0.046	34.225	18.574
Panel B: With Transaction Costs						
Methods based on real observed data						
EW	0.100	0.206	0.486	0.037	53.128	**3.031**
VW	0.096	0.220	0.437	0.024	58.086	3.464
Real Emp+Real Emp	0.022	0.115	0.196	0.003	35.023	16.514
Real BS+Real Emp	0.018	0.114	0.156	-0.002	35.198	16.410
Real OLSE+Real Emp	0.008	0.115	0.066	-0.012	39.104	16.418
Real Emp+Real LW	0.025	0.114	0.215	0.005	34.926	16.250
Real BS+Real LW	0.020	**0.113**	0.174	0.000	35.089	15.968
Real OLSE+Real LW	0.010	0.114	0.090	-0.009	39.040	15.952
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.197**	0.165	**1.190**	**0.156**	28.859	21.141
Diff BS+Diff Emp	0.193	0.164	1.175	0.152	29.749	20.790
Diff OLSE+Diff Emp	0.192	0.164	1.174	0.152	29.896	20.731
Diff Emp+Diff LW	0.165	0.160	1.029	0.126	31.709	20.343
Diff BS+Diff LW	0.162	0.160	1.016	0.124	32.464	20.081
Diff OLSE+Diff LW	0.162	0.160	1.015	0.124	32.570	20.038
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.152	0.159	0.959	0.115	**28.680**	19.440
Diff Emp+Real Emp	0.047	0.115	0.411	0.027	34.525	18.574

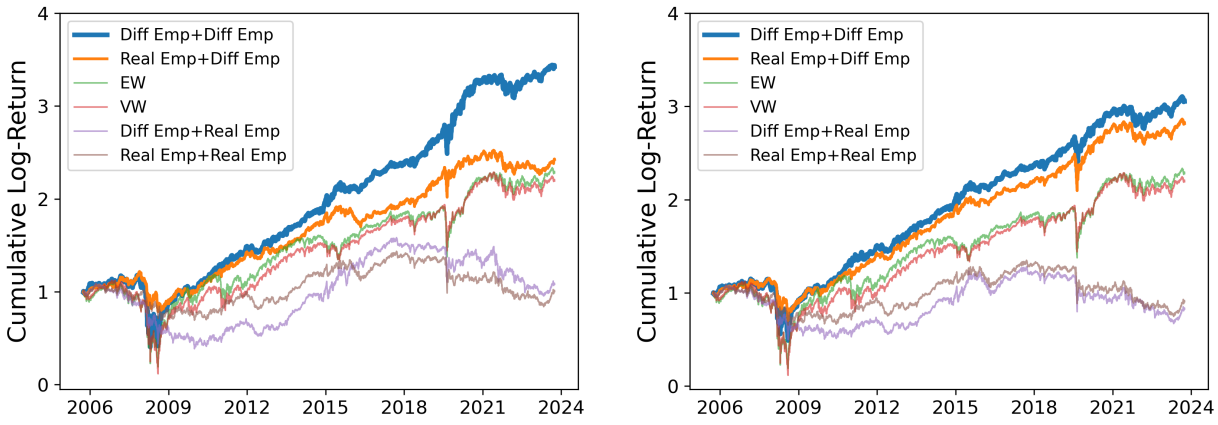
E.4 Robustness Analysis on Update Frequency

As a robustness check on the model update frequency, we also evaluate an annual update scheme with a rolling five-year window. Specifically, on May 1 of each year T, we reselect and pre-process the stocks, and update the model parameters using training data from May 1 of year $T-5$ to April 30 of year T. We test the model on data from May 1 of year T to April 30 of year $T+1$ to evaluate out-of-sample performance.

For mean-variance portfolios, we report out-of-sample portfolio performance under scenarios without and with transaction costs in Tables E.4 and E.5 for $\eta = 3$ and $\eta = 5$, respectively. Additionally, we plot the cumulative returns with transaction cost (in log scale) for $\eta = 3$ and $\eta = 5$ in Figure E.4. `Diff Emp+Diff Emp` outperforms all alternatives, which is consistent with the result observed under quarterly updates.

Figure E.4: Cumulative returns of different portfolios in log scale with transaction cost (model updated annually).

(a) $\eta = 3$. (b) $\eta = 5$.



Moreover, for factor tangency portfolios, we report Sharpe ratios across varying number of factors in Table E.6 and plot the correlation heatmaps between top eight factors estimated using diffusion-based methods and traditional factors in `FF Method` in Figure E.5. The diffusion-based methods exhibit higher Sharpe ratios and outperform all other approaches. The diffusion-generated factors are notably correlated with traditional factors, with Mkt-RF, LT-REV, and MOM as the three dominant factors across all three methods. Overall, the findings are similar to those under quarterly updates.

Table E.4: Performance of different portfolios with and without transaction costs for $\eta = 3$ (model updated annually).

Method	Mean	Std	SR	CER	MDD (%)	TO
Panel A: Without Transaction Costs						
Methods based on real observed data						
EW	0.102	0.221	0.462	0.029	58.114	**3.273**
VW	0.098	0.218	0.448	0.026	61.400	3.717
Real Emp+Real Emp	0.087	0.142	0.611	0.057	34.651	38.120
Real BS+Real Emp	0.078	0.140	0.553	0.048	31.806	37.344
Real OLSE+Real Emp	0.090	0.144	0.625	0.059	35.069	38.112
Real Emp+Real LW	0.085	0.134	0.635	0.058	31.475	32.143
Real BS+Real LW	0.076	**0.133**	0.569	0.049	31.963	31.540
Real OLSE+Real LW	0.086	0.136	0.632	0.058	35.529	32.417
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.189**	0.192	**0.983**	**0.133**	42.406	17.507
Diff BS+Diff Emp	0.185	0.190	0.972	0.131	42.021	17.203
Diff OLSE+Diff Emp	0.184	0.190	0.970	0.130	41.990	17.168
Diff Emp+Diff LW	0.155	0.169	0.917	0.112	38.046	16.332
Diff BS+Diff LW	0.152	0.168	0.906	0.110	37.908	16.115
Diff OLSE+Diff LW	0.152	0.168	0.904	0.110	37.897	16.090
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.124	0.148	0.840	0.091	**31.057**	16.752
Diff Emp+Real Emp	0.113	0.167	0.676	0.071	34.043	23.360
Panel B: With Transaction Costs						
Methods based on real observed data						
EW	0.096	0.221	0.433	0.022	58.807	**3.273**
VW	0.090	0.218	0.414	0.019	62.127	3.717
Real Emp+Real Emp	0.011	0.144	0.073	-0.021	39.327	38.120
Real BS+Real Emp	0.003	0.142	0.022	-0.027	40.017	37.344
Real OLSE+Real Emp	0.012	0.146	0.082	-0.020	45.209	38.112
Real Emp+Real LW	0.021	0.136	0.153	-0.007	**32.213**	32.143
Real BS+Real LW	0.013	**0.135**	0.094	-0.015	33.666	31.540
Real OLSE+Real LW	0.021	0.138	0.152	-0.007	43.520	32.417
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.153**	0.192	**0.797**	**0.098**	43.651	17.507
Diff BS+Diff Emp	0.150	0.191	0.788	0.096	43.267	17.203
Diff OLSE+Diff Emp	0.150	0.191	0.786	0.095	43.236	17.168
Diff Emp+Diff LW	0.122	0.170	0.720	0.079	38.127	16.332
Diff BS+Diff LW	0.120	0.168	0.711	0.077	37.974	16.115
Diff OLSE+Diff LW	0.119	0.168	0.709	0.077	37.962	16.090
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.090	0.148	0.608	0.057	34.729	16.752
Diff Emp+Real Emp	0.019	0.170	0.111	-0.024	36.913	23.360

Table E.5: Performance of different portfolios with and without transaction costs for $\eta = 5$ (model updated annually).

Method	Mean	Std	SR	CER	MDD (%)	TO
Panel A: Without Transaction Costs						
Methods based on real observed data						
EW	0.102	0.221	0.462	-0.020	58.114	**3.273**
VW	0.098	0.218	0.448	-0.021	61.400	3.717
Real Emp+Real Emp	0.080	0.140	0.568	0.030	32.223	38.121
Real BS+Real Emp	0.074	0.140	0.525	0.024	32.181	37.243
Real OLSE+Real Emp	0.061	0.142	0.426	0.010	33.213	37.553
Real Emp+Real LW	0.078	**0.133**	0.584	0.033	31.312	32.143
Real BS+Real LW	0.072	**0.133**	0.542	0.028	32.301	31.452
Real OLSE+Real LW	0.058	0.135	0.430	0.013	33.637	31.901
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.155**	0.167	**0.925**	**0.085**	45.760	17.647
Diff BS+Diff Emp	0.153	0.166	0.920	0.084	45.951	16.996
Diff OLSE+Diff Emp	0.152	0.166	0.918	0.084	45.937	16.970
Diff Emp+Diff LW	0.140	0.156	0.901	0.080	42.809	16.440
Diff BS+Diff LW	0.139	0.155	0.896	0.079	42.719	15.907
Diff OLSE+Diff LW	0.139	0.155	0.894	0.079	42.709	15.892
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.124	0.147	0.848	0.071	33.164	16.887
Diff Emp+Real Emp	0.084	0.149	0.566	0.029	**31.205**	18.639
Panel B: With Transaction Costs						
Methods based on real observed data						
EW	0.096	0.221	0.433	-0.026	58.807	**3.273**
VW	0.090	0.218	0.414	-0.029	62.127	3.717
Real Emp+Real Emp	0.005	0.143	0.035	-0.046	39.696	38.121
Real BS+Real Emp	-0.001	0.142	-0.005	-0.051	40.889	37.243
Real OLSE+Real Emp	-0.014	0.144	-0.100	-0.066	41.295	37.553
Real Emp+Real LW	0.014	**0.135**	0.107	-0.031	**33.287**	32.143
Real BS+Real LW	0.009	**0.135**	0.068	-0.036	35.225	31.452
Real OLSE+Real LW	-0.006	0.136	-0.042	-0.052	40.965	31.901
Methods based on diffusion-generated data						
Diff Emp+Diff Emp	**0.128**	0.167	**0.766**	**0.058**	47.549	17.647
Diff BS+Diff Emp	0.127	0.166	0.762	0.057	47.720	16.996
Diff OLSE+Diff Emp	0.126	0.166	0.760	0.057	47.704	16.970
Diff Emp+Diff LW	0.114	0.156	0.732	0.053	44.622	16.440
Diff BS+Diff LW	0.113	0.156	0.727	0.053	44.518	15.907
Diff OLSE+Diff LW	0.113	0.156	0.726	0.052	44.508	15.892
Methods based on both real observed data and diffusion-generated data						
Real Emp+Diff Emp	0.099	0.147	0.673	0.045	35.558	16.887
Diff Emp+Real Emp	0.043	0.149	0.288	-0.013	36.045	18.639

Table E.6: Out-of-sample Sharpe ratios of factor tangency portfolios (model updated annually). The number of factors is set to be 3, 5, 6, and 8, respectively.

# Factors	Diff+PCA	Diff+POET	Diff+RPPCA	FF	PCA	POET	RPPCA
3	1.805	1.841	**1.985**	0.648	0.402	0.872	0.631
5	2.158	2.178	**2.367**	0.726	0.453	0.930	1.250
6	2.322	2.339	**2.550**	0.861	0.528	1.356	1.701
8	2.631	2.739	**2.810**	0.881	0.673	1.463	1.892

Figure E.5: Correlation between the top 8 factors obtained using diffusion-based methods and those from the FF Method (model updated annually).