A solvable generative model with a linear, one-step denoiser

Indranil Halder IHALDER@G.HARVARD.EDU
Harvard University, Cambridge, MA, USA

Abstract

We develop an analytically tractable single-step diffusion model based on a linear denoiser and present an explicit formula for the Kullback-Leibler divergence between the generated and sampling distribution, taken to be isotropic Gaussian, showing the effect of finite diffusion time and noise scale. Our study further reveals that the monotonic fall phase of Kullback-Leibler divergence begins when the training dataset size reaches the dimension of the data points. Finally, for large-scale practical diffusion models, we explain why a higher number of diffusion steps enhances production quality based on the theoretical arguments presented before.

1. Introduction

In recent years, generative artificial intelligence has made tremendous advancements - be it image, audio, video, or text domains—on an unprecedented scale. Diffusion models [32, 39, 65, 69, 70] are among the most successful frameworks [55, 58, 60, 62]. The quality of the generated images can be enhanced through guided diffusion at the cost of reduced diversity [13, 20, 23, 31, 75]. Also, experimentally it is observed that increasing diffusion steps leads to more visually appealing images [42]. Theoretically understanding this phenomena and generalization ability [16, 29, 40, 41, 46, 54, 56, 68, 77, 79] of the diffusion models is a challenging task. Keeping this goal in mind, we introduce and study a linear denoiser based generative model that is analytically tractable and features some of the properties of a single denoising step in a realistic diffusion model.

1.1. Our contributions

Our main contributions to this paper are as follows:

1. We define a linear denoiser based generative model. Within the framework of the model, we present explicit formula for the Kullback-Leibler divergence between generated and sampling distribution, taken to be isotropic Gaussian, showing the effect of finite diffusion time and noise scale. In particular, our formula shows we can recover the sampling distribution from the generative model only if the noise scale is small enough compared to certain function of diffusion time.

2. We establish that aforementioned Kullback-Leibler divergence starts to decrease monotonically with addition of new training data when the size of the training set reaches the dimension of the data points as opposed to an exponential scale indicated by the curse of dimensionality.

3. For a realistic diffusion model on Gaussian mixture training set, we quantify the fact that larger diffusion step leads to better production quality. In addition, we show that the theorem that we proved before gives us theoretical explanation of this fact.

1.2. Related works

The main theoretical setup of our work is that of higher dimensional statistics, i.e, when the dimension and number of train data size both scale large simultaneously staying proportional to each other. In the context of linear regression [2, 25, 26, 30, 43, 52], kernel regression [11, 15, 48, 64, 66, 67, 72], and random feature models [1, 6, 7, 21, 24, 28, 30, 33, 47–49, 51, 78] method of deterministic equivalence has been used extensively for discussions of higher dimensional statistics.

Traditionally diffusion models are trained with datasets whose size is much smaller compared to the exponential of the data dimension. For example a comonly used dataset for training diffusion models is laion-high-resolution that contains around 10^8 images of dimension 1024×1024. Motivated by these facts, in this paper we study a specific linear denoiser based generative model that captures a single diffusion step of the realistic diffusion model using the method of deterministic equivalence. For stochastic differential equation based models, there are notable works discussing bound on the distance between sampling and generated distribution under the assumption of a given score estimation error [8, 17, 18, 44, 63]. On the other hand, in our work, we focus on the error in denoising for a single step taking into account finite sample size.

2. A generative model based on a linear denoiser

In this section, we define and study a linear denoiser based generative model which is analogous to a one step diffusion model. Before explaining the model, we note certain basic facts about the diffusion process based on a finite number of samples. Given n samples $\rho(x)$ can be approximated by the Dirac delta distribution $\hat{\rho}(0,x) \equiv \frac{1}{n}\sum_{k=1}^{n}\delta(x-x_k), x\in\mathbb{R}^d$. The time evolution of the probability distribution under Ornstein-Uhlenbeck diffusion process is given by (see Appendix A for more details)

$$\hat{\rho}(t,x) = \frac{1}{n}\sum_{k=1}^{n}\mathcal{N}(x|x_k e^{-t}, 1-e^{-2t}) \tag{2.1}$$

This motivates us to sample $Y_k, k=1,2,..,n$ from the underlying distribution $\rho(x)$ and add noise $Z_k\sim\mathcal{N}(0,\boldsymbol{I}_d)$ to it to obtain noisy samples

$$X_k = e^{-T}Y_k + \sqrt{\Delta_T}Z_k, \Delta_T = \lambda\delta_T = \lambda(1-e^{-2T}) \tag{2.2}$$

Here T is the diffusion time cut-off and λ is a free hyperparameter that controls the amount of noise added [1].

The denoiser based model, trained on the data above, as input takes a noisy sample X and generates a clean sample Y. In this paper, we consider a linear model $Y=\hat{\theta}_0+\hat{\theta}_1 X$ as prototype denoiser for analytical tractability. The parameters $\hat{\theta}_0,\hat{\theta}_1$ are solution to the linear regression problem of predicting $\{Y_k\}$ given $\{X_k\}$ and given by[2]

$$\hat{\theta}_1^T = (x^T x)^{-1}x^T y, \quad \hat{\theta}_0 = \hat{Y}-\theta_1\hat{X} \quad \hat{X}=\frac{1}{n}\sum_{k=1}^{n}X_k, \quad \hat{Y}=\frac{1}{n}\sum_{k=1}^{n}Y_k \tag{2.3}$$

Here x,y are $n\times d$ dimensional matrices whose k-th row is $(X_k-\hat{X})^T, (Y_k-\hat{Y})^T$ respectively.

1. This corresponds to scaling the noise term in (A.13) by a factor of $\sqrt{\lambda}$.
2. A natural generalization of this is to feature kernel regression instead of linear regression.

To generate samples from the trained diffusion model we first draw X from $\mathcal{N}(\mu_X, \sigma_X^2 \boldsymbol{I}_d)$ where with

$$\mu_X = e^{-T}\hat{Y}, \quad \sigma_X^2 = e^{-2T}\frac{1}{nd}\sum_{k=1}^{n}|Y_k - \hat{Y}|^2 + \Delta_T \tag{2.4}$$

and then use the diffusion model to predict corresponding $Y = \hat{\theta}_0 + \hat{\theta}_1 X$. The generated probability distribution for a given set $\{(X_k, Y_k), k = 1, 2, .., n\}$ is

$$\rho_G(Y|\{(X_k, Y_k)\}) = \mathcal{N}(Y|\hat{\theta}_0 + \hat{\theta}_1\mu_X, \sigma_X^2\hat{\theta}_1^T\hat{\theta}_1) \tag{2.5}$$

Effect of finite diffusion time

In this subsection, we study the effect of finite diffusion time T and noise scale λ on generalization error for the linear diffusion model defined above. We restrict our discussion to sampling from isotropic Gaussian distribution $\rho = \mathcal{N}(\mu, \sigma^2\boldsymbol{I}_d)$. The distance between the underlying distribution ρ and the generated ρ_G distribution from the diffusion model as given in (2.5) can be measured in terms of Kullback–Leibler divergence. Further it can be decomposed as $\text{KL}(\rho||\rho_G) = \text{KL}_{\text{mean}} + \text{KL}_{\text{var}} \geq \text{KL}_{\text{var}}$. Where the contributions $\text{KL}_{\text{mean}}, \text{KL}_{\text{var}}$ are related to the difference between generated and the underlying distribution in mean and variance

$$\text{KL}_{\text{mean}}(\rho_G|\rho) = \frac{1}{2\sigma^2}(\mu - \hat{\mu}_G)^T(\mu - \hat{\mu}_G), \quad \text{KL}_{\text{var}}(\rho_G|\rho) = \frac{1}{4}\text{Tr}\left(\left(\frac{\hat{\Sigma}_G}{\sigma^2} - I\right)^2\right) \tag{2.6}$$

Here $\hat{\mu}_G = \hat{\theta}_0 + \hat{\theta}_1\mu_X, \hat{\Sigma}_G = \sigma_X^2\hat{\theta}_1^T\hat{\theta}_1$. The inequality follows because KL_{mean} is a positive semi-definite quantity. We numerically show in figure 1 that there exists a regime of small λ where $\text{KL}_{\text{mean}} \ll \text{KL}_{\text{var}}$ making the inequality above an approximate equality.

Theorem 1 *When the linear diffusion model described above is trained on n samples from isotropic Gaussian distribution $\rho = \mathcal{N}(\mu, \sigma^2\boldsymbol{I}_d)$ in the limit of $n \to \infty$ holding $\alpha = d/n$ fixed, following lower bound on the KL divergence between generated and sampling distribution holds $KL(\rho||\rho_G) \geq KL_{var}$ If further we restrict ourselves to small noise scale $\lambda = \hat{\lambda}\sigma^2 e^{-2T}, \hat{\lambda} \ll 1$, then an explicit expression for the statistical expectation value of KL_{var} can be obtained order by order in $\hat{\lambda}$ based on the theory of deterministic equivalence. More specifically for $\alpha < 1$ we have*

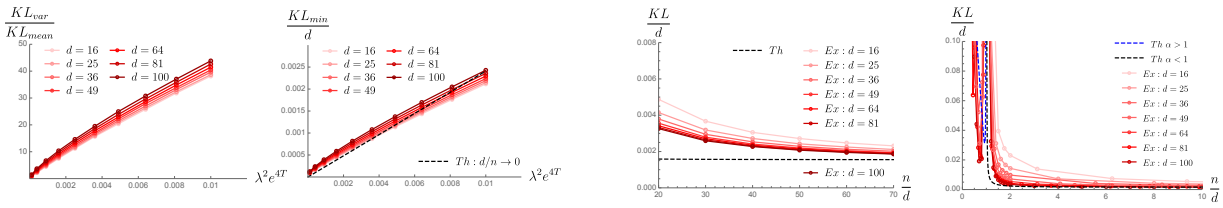
$$\langle KL_{var}\rangle = \frac{d\alpha\hat{\lambda}e^{-4T}\left(e^{2T}-1\right)}{2(1-\alpha)} + \frac{d\hat{\lambda}^2e^{-8T}\left(e^{2T}-1\right)^2}{4(1-\alpha)^3}\left(\alpha^2 + (1-\alpha)^3e^{4T} + 4\alpha(1-\alpha)^2e^{2T}\right) + \mathcal{O}(\hat{\lambda}^3) \tag{2.7}$$

and for $\alpha > 1$

$$\langle KL_{var}\rangle = d\frac{\alpha-1}{4\alpha} + \frac{d\hat{\lambda}e^{-4T}\left(e^{2T}-1\right)}{2(\alpha-1)} + \frac{d\hat{\lambda}^2e^{-8T}\left(e^{2T}-1\right)^2}{4(\alpha-1)^3\alpha}(\alpha^3 + (\alpha-1)^3e^{4T} + 4\alpha(\alpha-1)^2e^{2T}) + \mathcal{O}(\hat{\lambda}^3) \tag{2.8}$$

Proof See Appendix C. ■

Lemma 2 *For $n > d$, KL_{var} is a monotonically decreasing function of n/d.*



(a) (b)

Figure 1: In figure (a), we plot various contributions to KL divergence between the generated data from the linear denoiser based generative model and sampling distribution taken to be an isotropic Gaussian of mean $\mu = 10$ and diagonal standard deviation $\sigma = 1$ of dimension d. We have fixed the diffusion time cut-off $T = 2$ and varied the noise scale $\lambda = \hat{\lambda}e^{-4}$. The train dataset size $n = 10^4 \gg d$. From the plot on left we see there exists a regime of parameters when $\text{KL}_{\text{mean}} \ll \text{KL}_{\text{var}}$. This justifies our assumption of ignoring KL_{mean} in analytic calculation presented in appendix C. The plot on the right compares the numerical results against the theoretical result and shows that the minimum KL divergence attainable in $d/n \to 0$ limit scales quadratically with the noise parameter $\hat{\lambda} = \lambda e^{2T}/\sigma^2$ for small values of the later. In figure (b), we have fixed the diffusion time to be $T = 2$ with noise scale $\lambda = 0.8e^{-4}$. On the left, we plot KL divergence between the generated and sampling distribution after truncation to the quadratic order in λ, as given in (C.10), in the regime of small d/n comparing experimental data (in red) and theoretical result for the lower bound as given in (2.7) (in black). The numerical results on the right plot shows that KL divergence between the generated and underlying distribution scales as d times solely a function of $\alpha = d/n$ without additional n, d dependence as we take n, d large keeping their ratio fixed. This fact is analytically established in (2.7),(2.8) and the analytical expression is plotted in black for $\alpha < 1$ and blue $\alpha > 1$.

Proof Derivative of RHS in (2.7) with respect to α is positive. ■

Lemma 3 *In $n/d \to \infty$ limit, KL_{var}/d scales as $\lambda^2 e^{4T}(1 - e^{-2T})^2$. Hence we conclude we can recover the underlying sampling distribution in this limit only if $\lambda e^{2T}(1 - e^{-2T}) \ll 1$.*

Proof Consider $\alpha \to 0$ limit of RHS in (2.7). ■

Thse are in agreement with the plot in figure 1.

3. Non-linear diffusion model

From figure 2, it is clear that production quality of a realistic diffusion model improves with higher diffusion steps s. This fact can be explained based on our theoretical analysis as follows. In the diffusion model we start from a clear image x_0 and then obtain noisy images for steps $t = 1, 2, ..., s$ from (equation (2) of [32])

$$x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\lambda}\sqrt{\beta_t} Z, \quad Z \sim \mathcal{N}(0, \boldsymbol{I}_d) \tag{3.1}$$

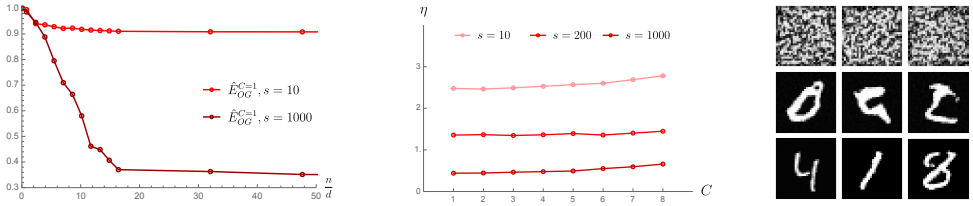


Figure 2: The plot is based on PyTorch-based implementation of the algorithm in [32]. The denoiser has the structure of U-Net [61] with additional residual connections consisting of positional encoding of the image and attention layers [27, 57, 73]. The train dataset is equal weight Gaussian mixture model (GMM) with C components $i = 0, 1, 2, .., C-1$ of dimension $d = 64$. The i-th component is an isotropic Gaussian of mean $\mu_i = \mu_0 + (i - (C-1)/2)\sigma_0, \mu_0 = 0.5$ and standard deviation $\sigma_i = 0.1$. The number of samples in the original dataset is $N = 10^4$. Plot on the left shows how increasing diffusion steps reduces $\hat{E}_{OG}$ - it is the scaled value of the distance between generated and original distribution E_{OG} by its value at $n = 10$ for $C = 1$. For the plot in the middle the training is done for 10 epoch with batch size 128. The plot in the middle shows a linear dependence of error $\eta = -\log(1 - E_{OG}(d/n \to 0)/E_{OG}(d/n \to \infty))$ on sample complexity C. On the right, we have generated images from the model with same hyper-parameter configurations on MNIST dataset. We can clearly see that image quality improves as diffusion steps increase.

Our observation is that it becomes identical to (2.2) if we choose the noise schedule as follows

$$\beta_t = 1 - e^{-2\beta t/s} = (2t/s)\beta + \mathcal{O}(\beta^2), \beta \ll 1 \tag{3.2}$$

with the map $x_t \to X, x_{t-1} \to Y, \beta t/s \to T$. If instead of using a non-linear neural network we use the linear denoiser to predict a Gaussian approximation of x_{t-1} from x_t we can use lemma 3 and find that $\mathrm{KL}_{\mathrm{var}}(t)/d$ scales as $\lambda^2\beta_t^2 = 4\lambda^2\beta^2(t/s)^2$. This shows that performance of the final denoising step is improved by a factor of s^2 compared to single step diffusion. In fact, performance of each step improves except the first one between x_s, x_{s-1} which remain the same. This suggests as we increase s overall production quality of the model will improve in agreement with the findings in figure 2.[3] This argument also predicts that we need to have $\lambda\beta \ll 1$ for good quality generated images.

The choice of different noise schedules β_t [19, 36, 53] induces distinct linear models for each single step of the diffusion process. In the future, these models can be analyzed using techniques analogous to those employed in the present study, thereby enabling a systematic comparison of the performance associated with various noise schedules.

Moreover, the analytical framework developed here may be extended to encompass the analysis of (a stack of) wide neural networks, either in the kernel approximation regime [4, 12, 14, 22, 37, 38, 45, 59] or in mean field regime [9, 10, 50, 76] in place of the linear diffusion model considered in this work.

3. For this purpose, the distance between original and generated distribution is calculated using

$$\mathrm{E}_{OG} = \sum_{i=1}^{d} \int dx\, (\rho_{O,i}(x) - \rho_{G,i}(x))^2, \quad \rho_{O/G,i}(x) \equiv \frac{1}{|S_{O/G}|} \sum_{k=1}^{|S_{O/G}|} \mathcal{N}(x|x_k^i, \epsilon_{O,i}^2), \quad \epsilon_{O,i}^2 = \frac{\hat{\Sigma}_{O,i}}{|S_{O/G}|^2} \tag{3.3}$$

Here $\hat{\Sigma}_{O,i}$ is the empirical variance obtained from the original dataset S_O.

Appendix A. Foundations of diffusion-driven generative models

In this Appendix we review the connection between stochastic interpolant and stochastic differential equation based generative models [3, 34, 42, 71, 80]. Given two probability density functions ρ_0, ρ_1, one can construct a stochastic interpolant between ρ_0 and ρ_1 as follows

$$x(t) = X(t, x_0, x_1) + \lambda_0(t)z, \qquad t \in [0, 1] \tag{A.1}$$

where the function X, λ_0 satisfies

$$\begin{aligned} &X(0, x_0, x_1) = x_0, \quad X(1, x_0, x_1) = x_1, \quad ||\partial_t X(t, x_0, x_1)|| \leq C||x_0 - x_1|| \\ &\lambda_0(0) = 0, \quad \lambda_0(1) = 0, \quad \lambda_0(t) \geq 0 \end{aligned} \tag{A.2}$$

for some positive constant C. Here x_0, x_1, z are drawn independently from a probability measure ρ_0, ρ_1 and standard normal distribution $\mathcal{N}(0, \boldsymbol{I})$. The probability distribution $\rho(t, x)$ of the process $x(t)$ satisfies the transport equation[4]

$$\partial_t \rho + \nabla \cdot (b\rho) = 0, \quad \rho(0, x) = \rho_0(x), \quad \rho(1, x) = \rho_1(x), \tag{A.3}$$

where we defined the velocity[5]

$$b(t, x) = \mathbb{E}[\dot{x}(t)|x(t) = x] = \mathbb{E}[\partial_t X(t, x_0, x_1) + \dot{\lambda}_0(t)z|x(t) = x]. \tag{A.4}$$

One can estimate the velocity field by minimizing

$$L_b[\hat{b}] = \int_0^1 \mathbb{E}\left(\frac{1}{2}||\hat{b}(t, x(t))||^2 - \left(\partial_t X(t, x_0, x_1) + \dot{\lambda}_0(t)z\right) \cdot \hat{b}(t, x(t))\right) dt \tag{A.5}$$

It's useful to introduce the score function $s(t, x)$ for the probability distribution for making the connection to the stochastic differential equation

$$s(t, x) = \nabla \log \rho(t, x) = -\lambda_0^{-1}(t)\mathbb{E}(z|x(t) = x) \tag{A.6}$$

It can be estimated by minimizing

$$L_s[\hat{s}] = \int_0^1 \mathbb{E}\left(\frac{1}{2}||\hat{s}(t, x(t))||^2 + \lambda_0^{-1}(t)z \cdot \hat{s}(t, x(t))\right) dt \tag{A.7}$$

The score function also can be obtained by minimizing the following alternative objective function known as the Fisher divergence

$$\begin{aligned} L_F[\hat{s}] &= \tfrac{1}{2}\int_0^1 \mathbb{E}\left(||\hat{s}(t, x(t)) - \nabla \log \rho(t, x)||^2\right) dt \\ &= \int_0^1 \mathbb{E}\left(\frac{1}{2}||\hat{s}(t, x(t))||^2 + \nabla \cdot \hat{s}(t, x(t)) + \frac{1}{2}||\nabla \log \rho(t, x))||^2\right) dt \end{aligned} \tag{A.8}$$

4. Here we are using the notation $\nabla = \nabla_x$.
5. The expectation is taken independently over $x_0 \sim \rho_0, x_1 \sim \rho_1$ and $z \sim \mathcal{N}(0, \boldsymbol{I})$. Here $\mathcal{N}(0, \boldsymbol{I})$ is normalized Gaussian distribution of appropriate dimension with vanishing mean and variance.

To obtain the second line we have ignored the boundary term. Note that for the purpose of minimization the last term is a constant and hence it plays no role hence Fisher divergence can be minimized from a set of samples drawn from ρ easily even if the explicit form of ρ is not known [35]. However, the estimation of $\nabla \cdot \hat{s}(t, x(t))$ is computationally expensive and in practice one uses denoising score matching for estimating the score function [74].

It is easy to put eq. (A.3) into Fokker-Planck-Kolmogorov form

$$\begin{aligned} \partial_t \rho + \nabla \cdot (b_F \rho) &= +\lambda(t) \Delta \rho, \qquad b_F(t,x) = b(t,x) + \lambda(t) s(t,x) \\ \partial_t \rho + \nabla \cdot (b_B \rho) &= -\lambda(t) \Delta \rho, \qquad b_B(t,x) = b(t,x) - \lambda(t) s(t,x) \end{aligned} \tag{A.9}$$

For an arbitrary function $\lambda(t) \geq 0$. From this, we can read off the Itô SDE as follows[6]

$$\begin{aligned} dX_t^F &= b_F(t, X_t^F) dt + \sqrt{2\lambda(t)}\, dW_t \\ dX_t^B &= b_B(t, X_t^B) dt - \sqrt{2\lambda(t)}\, dW_{1-t} \end{aligned} \tag{A.10}$$

The first equation is solved forward in time from the initial data $X^F_{t=0} \sim \rho_0$ and the second one is solved backward in time from the final data $X^B_{t=1} \sim \rho_1$. One can recover the probability distribution ρ from the SDE using Feynman–Kac formulae[7]

$$\begin{aligned} \rho(t,x) &= \mathbb{E}\left(e^{\int_t^0 \nabla \cdot b_F(t, Y_t^B) dt} \rho_0(Y^B_{t=0}) | Y_t^B = x \right) \\ &= \mathbb{E}\left(e^{\int_t^1 \nabla \cdot b_B(t, Y_t^F) dt} \rho_1(Y^F_{t=1}) | Y_t^F = x \right) \end{aligned} \tag{A.12}$$

In the domain of image generation, we don't know the exact functional form of the sampling distribution $\rho(x)$. However we have access to a finite number of samples from it and the goal is to generate more data points from the unknown probability density $\rho(x)$. Traditional likelihood maximization techniques would assume a trial density function ρ_θ and try to adjust θ so that likelihood for obtaining known samples is maximized. In this process determination of the normalization of ρ_θ is computationally expensive as it requires multi-dimensional integration (typically it is required for each step of the optimization procedure for θ). Diffusion based generative models are an alternative [32, 65, 70]. In this section, we review basic notions of these stochastic differential equation based models. In particular, we examine an exactly solvable stochastic differential equation (SDE). The Itô SDE under consideration is known as the Ornstein-Uhlenbeck Langevin dynamics and is expressed by:

$$dX_t^F = -X_t^F dt + \sqrt{2} dW_t, \quad X_t^F \sim \rho(t). \tag{A.13}$$

The score function associated with the stochastic process will be denoted as

$$s(t,x) = \nabla_x \log \rho(t,x) = \frac{1}{\rho(t,x)} \nabla_x \rho(t,x) \tag{A.14}$$

6. Here W_t represents a standard Wiener process, i.e., $W_t - tW_1 = N_t$ is a zero-mean Gaussian stochastic process that satisfies $\mathbb{E}[N_t N_t^{\mathsf{T}}] = t(1-t)\boldsymbol{I}$.

7. A class of exactly solvable models are given by (Ornstein-Uhlenbeck dynamics discussed in the main text is a special case of this equation)

$$dX_t^F = X_t^F \frac{d}{dt}(\log \eta(t)) dt + \sqrt{\eta(t)^2 \frac{d}{dt}\left(\frac{\sigma(t)^2}{\eta(t)^2}\right)} dW_t, \quad X_t^F \sim \mathcal{N}(\eta(t) X_0^F, \sigma(t)^2) \tag{A.11}$$

Where η, σ are two positive functions satisfying $\eta(0) = 1, \sigma(0) = 0$.

The probability density ρ satisfies the transport equation (see (A.3))

$$\begin{aligned}\partial_t \rho(t,x) &= \nabla \cdot ((x + s(t,x))\rho(t,x)) \\ &= \nabla^2 \rho(t,x) + x.\nabla\rho(t,x) + d\rho(t,x).\end{aligned} \tag{A.15}$$

The dimension of the data is defined to be given by $d = \dim(x)$. The time evolution of the probability distribution is exactly solvable and given by

$$\rho(t, X_t^F) = \int dX_0^F \; \rho(0, X_0^F)\, \mathcal{N}(X_t^F | X_0^F e^{-t}, 1 - e^{-2t}). \tag{A.16}$$

Suppose we know the probability density $\rho(0,x)$ exactly. One way to sample from it would be to use the knowledge of the exact score function $s(t,x)$ in the reverse diffusion process (see (A.10)), i.e,

$$dX_t^B = (-X_t^B - 2s(t, X_t^B))dt - \sqrt{2}\, dW_{1-t} \tag{A.17}$$

starting from a late time distribution $\rho(T,x)$ (it is assumed that we know how to sample from $\rho(T,x)$).

Appendix B. Principle of deterministic equivalence

In this appendix we review the theory of large random matrices leading to the principle of deterministic equivalence. A $d \times d$ Hermitian random matrix A with measure $d\mu_A$ is called an invariant random matrix if the measure satisfies

$$d\mu_A(A) = d\mu_A(U^\dagger A U) \tag{B.1}$$

for any unitary matrix U. In the limit of $d \to \infty$, the theory is conveniently described in terms of the single eigenvalue density ρ_A (normalized to unity) that can be obtained from the resolvent or the Stieltjes transform

$$G_A(z) = \langle \frac{1}{d} \mathrm{Tr}\left(\frac{1}{z - A}\right)\rangle = \int \frac{\rho_A(\lambda) d\lambda}{z - \lambda} \implies \rho_A(\lambda) = -\frac{1}{\pi} \lim_{\epsilon \to 0+} \Im(G_A(\lambda + i\epsilon)) \tag{B.2}$$

The moment generating function is given by

$$M_A(z) = \frac{1}{z} G_A\left(\frac{1}{z}\right) - 1 = \langle \frac{1}{d} \mathrm{Tr} \sum_{i=1}^{\infty} A^i z^i \rangle \tag{B.3}$$

R, S transformation of the eigenvalue density are defined by

$$\begin{aligned} R_A(z) &= G_A^{inv}(z) - \frac{1}{z}, \quad G_A^{inv}(G_A(z)) = z \\ S_A(z) &= \frac{z+1}{z} M_A^{inv}(z), \quad M_A^{inv}(M_A(z)) = z \end{aligned} \tag{B.4}$$

R, S transformations are useful when we study the matrix model to the leading order in large d limit as we explain next. Two invariant random matrices A, B are called free to the leading order in large d limit if they are independent. Free sum and free product of A, B are defined as follows

$$\begin{aligned} A \boxplus B &= U^\dagger A U + V^\dagger B V \\ A \star B &= \sqrt{A} B \sqrt{A} \end{aligned} \tag{B.5}$$

Here U, V are are sampled independently from uniform measure on the unitary group, i.e., Haar random unitary. It can be shown that for two invariant, independent random matrices A, B the moment generating function of $A \boxplus B$ and $A + B$ coincides, similarly moment generating function of $A \star B$ and AB coincides (to the leading order in large d, i.e., when they are free). Furthermore following identity holds to the leading order in large d limit for two free matrices A, B

$$R_{A\boxplus B}(z) = R_A(z) + R_B(z), \quad S_{A\star B}(z) = S_A(z)S_B(z) \tag{B.6}$$

Now we turn to application of these ideas. Consider $d \times d$ matrix $\hat{\Sigma} = x^T x/n$ where each row of x (there are n rows) is drawn from $\mathcal{N}(0, \Sigma)$. Then it can be written as a free product of Σ and a white Wishart matrix W (corresponds to $x^T x/n$ where each row of x is drawn from $\mathcal{N}(0, \boldsymbol{I}_d)$): $\hat{\Sigma} = \Sigma \star W$. From the definition of S transformation it follows that

$$M_{\hat{\Sigma}}(z) = \frac{1}{\frac{S_{\hat{\Sigma}}(M_{\hat{\Sigma}}(z))}{z} - 1} = \frac{1}{\frac{S_{\Sigma}(M_{\hat{\Sigma}}(z))S_W(M_{\hat{\Sigma}}(z))}{z} - 1} = M_\Sigma\left(\frac{z}{S_W(M_{\hat{\Sigma}}(z))}\right) \tag{B.7}$$

To obtain the final equality we used self-consistency of the equation itself. To recast this equation in a compact way we define $d\mathrm{f}^1_A(z) = -M_A(-1/z) = \langle \mathrm{Tr}\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-1}\rangle/d$. In terms of this new quantity we have

$$d\mathrm{f}^1_{\hat{\Sigma}}(\hat{R}) = d\mathrm{f}^1_{\Sigma}(R), \quad \hat{R} = R(1 - \alpha\, d\mathrm{f}^1_{\Sigma}(R)) \tag{B.8}$$

To obtain this equation we used knowledge of S transformation of white Wishart matrices $S_W(z) = 1/(1 + \alpha z), \alpha = d/n$. This equation is valid only leading order in large d, n limit with fixed α. It is known as the principle of deterministic equivalence. See [5] and references therein for a recent discussion of it.

Appendix C. Generalization error from deterministic equivalence

In this appendix, we provide proof of the main theorem in the paper (2.7), (2.8). Consider the scenario when the underlying sampling distribution is an isotropic Gaussian $\rho = \mathcal{N}(\mu, \sigma^2 I_d)$. The linear diffusion model $Y = \theta_0 + \theta_1 X$ is trained to solve the following linear regression problem

$$\begin{aligned} &Y_k = e^T X_k + Z_k, \quad X_k \sim \mathcal{N}(\mu_X, \Sigma = \sigma_X^2 I_d), \quad Z_k \sim \mathcal{N}(0, \Delta_T I_d), \quad k = 1, \ldots, n \\ &\mu_X = e^{-T}\mu, \quad \sigma_X^2 = e^{-2T}\sigma^2 + \Delta_T, \quad \Delta_T = \lambda(1 - e^{-2T}) \end{aligned} \tag{C.1}$$

Here X_k, Z_k are taken independent of each other.[8] The optimal value of the weights $\hat{\theta}_0, \hat{\theta}_1$ that minimizes the standard square loss are given by

$$\hat{\theta}_1^T = (x^T x + n\hat{R})^{-1} x^T y, \quad \hat{\theta}_0 = \hat{Y} - \hat{\theta}_1 \hat{X} \quad \hat{X} = \frac{1}{n}\sum_{k=1}^n X_k, \quad \hat{Y} = \frac{1}{n}\sum_{k=1}^n Y_k \tag{C.2}$$

Here x, y are $n \times d$ dimensional matrices whose k-th row is $(X_k - \hat{X})^T, (Y_k - \hat{Y})^T$ respectively (e.g. $x_{iA} = (X_i - \hat{X})_A$ etc.) and $\hat{R}$ is a scalar ridge parameter. Once trained the diffusion model generates data from $\rho_G = \mathcal{N}(\hat{\mu}_G, \hat{\Sigma}_G)$, $\hat{\mu}_G = \hat{\theta}_0 + \hat{\theta}_1 \mu_X, \hat{\Sigma}_G = \sigma_X^2 \hat{\theta}_1^T \hat{\theta}_1$.

8. Comparing first and second moment of $Y_k = a(T)X_k + c(T) + Z_k$, for X_k given as in (2.2), with the expected answer, we can see that in the domain considered here, i.e, $\lambda = \hat{\lambda}\sigma^2 e^{-2T}$, $\hat{\lambda} \ll 1$, the solution to the leading order in $\hat{\lambda}$ is indeed given by (C.1).

KL divergence between two PDF $\rho_1 = \mathcal{N}(\mu_1, \Sigma_1), \rho_2 = \mathcal{N}(\mu_2, \Sigma_2)$ is given by

$$\begin{aligned}\text{KL}(\rho_1|\rho_2) =& \int \rho_1(x) \log \frac{\rho_1(x)}{\rho_2(x)} dx \\ =& \frac{\text{Tr}(\Sigma_2^{-1}\Sigma_1) - \text{Tr}(I)}{2} - \frac{1}{2}\log|\Sigma_2^{-1}\Sigma_1| + \frac{1}{2}(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\end{aligned} \tag{C.3}$$

We choose $\mu_2 = \mu, \Sigma_2 = \sigma^2 I_d$ to correspond to the underlying distribution and $\mu_1 = \hat{\mu}_G, \Sigma_1 = \hat{\Sigma}_G$ corresponds to the generated distribution. This simplifies the formula above to

$$\begin{aligned}\text{KL}(\rho_G|\rho) =& \frac{1}{2}(\text{Tr}\left(\frac{\hat{\Sigma}_G}{\sigma^2}\right) - \text{Tr}(I)) - \frac{1}{2}\text{Tr}\log\left(\frac{\hat{\Sigma}_G}{\sigma^2}\right) + \frac{1}{2\sigma^2}(\mu-\hat{\mu}_G)^T(\mu-\hat{\mu}_G) \\ \geq& \frac{1}{2}(\text{Tr}\left(\frac{\hat{\Sigma}_G}{\sigma^2}\right) - \text{Tr}(I)) - \frac{1}{2}\text{Tr}\log\left(\frac{\hat{\Sigma}_G}{\sigma^2}\right)\end{aligned} \tag{C.4}$$

To go to the second line we have ignored the positive semi-definite term related to difference in mean between generated and underlying distribution. We proceed to calculate the variance term in KL divergence above. It follows that

$$\begin{aligned}\hat{\theta}_1^T &= (x^T x + n\hat{R})^{-1} x^T y \\ &= (x^T x + n\hat{R})^{-1} x^T (x\bar{\theta}_1^T + z) \\ &= e^T(1 - \hat{R}(\hat{\Sigma}+\hat{R})^{-1}) + (\hat{\Sigma}+\hat{R})^{-1}\frac{x^T z}{n}\end{aligned} \tag{C.5}$$

We have defined $\bar{\theta}_1 = e^T I_d, \hat{\Sigma} = x^T x/n$ for later convenience. Next we calculate

$$\begin{aligned}\hat{\theta}_1^T\hat{\theta}_1 =& \bar{\theta}_1^T\bar{\theta}_1 + \hat{\Sigma}_{\theta_1} \\ \hat{\Sigma}_{\theta_1} =& (\hat{\Sigma}+\hat{R})^{-1}\frac{x^T z}{n}\frac{z^T x}{n}(\hat{\Sigma}+\hat{R})^{-1} + e^{2T}\hat{R}^2(\hat{\Sigma}+\hat{R})^{-2} - 2e^{2T}\hat{R}(\hat{\Sigma}+\hat{R})^{-1} \\ &+ e^T\left(\frac{z^T x}{n}(\hat{\Sigma}+\hat{R})^{-1} + (\hat{\Sigma}+\hat{R})^{-1}\frac{x^T z}{n}\right) - e^T\hat{R}\,(\hat{\Sigma}+\hat{R})^{-1}\left(\frac{z^T x}{n} + \frac{x^T z}{n}\right)(\hat{\Sigma}+\hat{R})^{-1}\end{aligned} \tag{C.6}$$

Plugging this back in the expression of $\hat{\Sigma}_G$ we get

$$\begin{aligned}\frac{\hat{\Sigma}_G}{\sigma^2} =& (1+\sigma^{-2}e^{2T}\Delta_T)(I + e^{-2T}\hat{\Sigma}_{\theta_1}) = I + \hat{\sigma}_G \\ \hat{\sigma}_G =& (e^{-2T} + \sigma^{-2}\Delta_T)\hat{\Sigma}_{\theta_1} + e^{2T}\sigma^{-2}\Delta_T I\end{aligned} \tag{C.7}$$

Appearance of logarithm in the KL divergence makes it difficult to calculate its statistical expectation value. In next sub-section we develop a controlled expansion for this purpose.

Analytic tractability and various approximations

From (C.7) we see that the generated distribution remains close to the original underlying distribution if both Δ_T and $\hat{\sigma}_G$ remain small. To this end, we focus on the following limit:

$\lambda = \hat{\lambda}\sigma^2 e^{-2T}, \hat{R} = \lambda\hat{r}$. In this regime $\hat{\sigma}_G \sim \hat{\lambda}$. Further taking $\hat{\lambda} \ll 1$ makes $\hat{\sigma}_G$ small and we can approximate

$$\log\left(\frac{\hat{\Sigma}_G}{\sigma^2}\right) = \log(I + \hat{\sigma}_G) = \hat{\sigma}_G - \frac{1}{2}\hat{\sigma}_G^2 + \mathcal{O}(\hat{\sigma}_G^3) \tag{C.8}$$

Plugging this back into the expression of KL divergence (C.4) we get $\mathrm{KL}(\rho||\rho_G) = \mathrm{KL}_{\text{mean}} + \mathrm{KL}_{\text{var}}$, where

$$\mathrm{KL}_{\text{mean}}(\rho_G|\rho) = \frac{1}{2\sigma^2}(\mu - \hat{\mu}_G)^T(\mu - \hat{\mu}_G), \quad \mathrm{KL}_{\text{var}}(\rho_G|\rho) = \frac{1}{4}\mathrm{Tr}\left(\left(\frac{\hat{\Sigma}_G}{\sigma^2} - I\right)^2\right) \tag{C.9}$$

We focus on the variance term. Plugging back expressions from previous analysis

$$\begin{aligned}
\mathrm{KL}(\rho_G|\rho)_{\text{var}} =& \frac{1}{4}\mathrm{Tr}\left(\frac{\hat{\Sigma}_G}{\sigma^2} - I\right)^2 = \frac{1}{4}\mathrm{Tr}(\hat{\sigma}_G^2) \\
=& \frac{1}{4}\mathrm{Tr}(((e^{-2T} + \sigma^{-2}\Delta_T)\hat{\Sigma}_{\theta_1} + \sigma^{-2}e^{2T}\Delta_T)^2) \\
=& \frac{1}{4}(e^{-2T} + \sigma^{-2}\Delta_T)^2\mathrm{Tr}\hat{\Sigma}_{\theta_1}^2 \\
&+ \frac{1}{2}(e^{-2T} + \sigma^{-2}\Delta_T)\sigma^{-2}e^{2T}\Delta_T\mathrm{Tr}\hat{\Sigma}_{\theta_1} + \frac{d}{4}(\sigma^{-2}e^{2T}\Delta_T)^2
\end{aligned} \tag{C.10}$$

We are interested in statistical average of the expression above. We consider the following higher dimensional statistics limit: $n \to \infty, d \to \infty$ keeping $\alpha = d/n$ fixed. In this limit, we can take advantage of principle of deterministic equivalence discussed in previous appendix:

$$d\mathrm{f}^1_{\hat{\Sigma}}(\hat{R}) = d\mathrm{f}^1_{\Sigma}(R), \quad d\mathrm{f}^n_{\hat{\Sigma}}(\hat{R}) = \frac{1}{d}\langle\mathrm{Tr}\hat{\Sigma}^n(\hat{\Sigma} + \hat{R})^{-n}\rangle, \quad \hat{R} = R(1 - \alpha\, d\mathrm{f}^1_{\Sigma}(R)) \tag{C.11}$$

Since x, z are statistically independent and z has zero mean, we get

$$\mathrm{Tr}\langle\hat{\Sigma}_{\theta_1}\rangle = \mathrm{Tr}\langle(\hat{\Sigma} + \hat{R})^{-1}\frac{x^Tz}{n}\frac{z^Tx}{n}(\hat{\Sigma} + \hat{R})^{-1} + e^{2T}\hat{R}^2(\hat{\Sigma} + \hat{R})^{-2} - 2e^{2T}\hat{R}(\hat{\Sigma} + \hat{R})^{-1}\rangle \tag{C.12}$$

The first term is simplified after performing statistical average over z

$$(x^Tzz^Tx)_{AB} = x^T_{Ai}z_{iC}z_{jC}x_{jB} \to nd\Delta_T\hat{\Sigma}_{AB} \tag{C.13}$$

The factor of d came from sum over C (we are using the convention of repeated index implies sum). The first term becomes $\alpha\Delta_T$ times

$$\mathrm{Tr}\langle(\hat{\Sigma} + \hat{R})^{-1}\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-1}\rangle = \mathrm{Tr}\langle\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-2}\rangle = -\partial_{\hat{R}}\langle\mathrm{Tr}\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-1}\rangle = -d\partial_{\hat{R}}\langle d\mathrm{f}^1_{\Sigma}(R)\rangle \tag{C.14}$$

For the case we are considering,

$$d\mathrm{f}^1_{\Sigma=\sigma_X^2 I_d}(R) = d\mathrm{f}^1_{\Sigma=I_d}(\sigma_X^{-2}R) = \frac{1}{1 + \sigma_X^{-2}R} \tag{C.15}$$

Putting these expressions together the first term becomes

$$\alpha d\Delta_T\frac{\sigma_X^{-2}}{(1 + \sigma_X^{-2}R)^2}\partial_{\hat{R}}R = \alpha d\Delta_T\sigma_X^{-2}\frac{(d\mathrm{f}^1_{\Sigma}(R))^2}{1 - \alpha d\mathrm{f}^2_{\Sigma}(R)} \tag{C.16}$$

To obtain the second line we have used the following identity

$$d\mathrm{f}_\Sigma^{n+1}(R) = \left(1 + \frac{R}{n}\partial_R\right) d\mathrm{f}_\Sigma^n(R), \quad \partial_{\hat{R}} R = \frac{1}{1 - \alpha d\mathrm{f}_\Sigma^2(R)} \tag{C.17}$$

The third term is $-2e^{2T}$ times

$$\mathrm{Tr}\langle \hat{R}(\hat{\Sigma} + \hat{R})^{-1}\rangle = d(1 - \ d\mathrm{f}_\Sigma^1(R)) \tag{C.18}$$

The second term is e^{2T} times

$$\begin{aligned}\mathrm{Tr}\langle (\hat{R}(\hat{\Sigma} + \hat{R})^{-1})^2\rangle &= \mathrm{Tr}\langle 1 - 2\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-1} + \hat{\Sigma}^2(\hat{\Sigma} + \hat{R})^{-2}\rangle \\ &= d - 2d\ d\mathrm{f}_{\Sigma(R)}^1 + d\ d\mathrm{f}_{\hat{\Sigma}}^2(\hat{R}) \\ &= d - 2d\ d\mathrm{f}_\Sigma^1(R) + d(1 + \frac{\hat{R}}{1 - \alpha d\mathrm{f}_\Sigma^2(R)}\partial_R d\mathrm{f}_\Sigma^1(R))\end{aligned} \tag{C.19}$$

Combining all these we get

$$\begin{aligned}\mathrm{Tr}\langle \hat{\Sigma}_{\theta_1}\rangle =& \alpha d\Delta_T \sigma_X^{-2} \frac{(d\mathrm{f}_\Sigma^1(R))^2}{1 - \alpha d\mathrm{f}_\Sigma^2(R)} - 2e^{2T} d(1 - \ d\mathrm{f}_\Sigma^1(R)) \\ &+ e^{2T} d\left(2 - 2\ d\mathrm{f}_\Sigma^1(R) + \frac{\hat{R}}{1 - \alpha d\mathrm{f}_\Sigma^2(R)}\partial_R d\mathrm{f}_\Sigma^1(R)\right)\end{aligned} \tag{C.20}$$

Now we turn to evaluate $\mathrm{Tr}\langle \hat{\Sigma}_{\theta_1}^2\rangle$. We want to keep track of terms that are order d and ignore sub-leading terms. This restricts possible contractions of z, z^T. We get a factor of d only from contractions that happen next to each other. Keeping only those terms

$$\begin{aligned}\mathrm{Tr}\langle \hat{\Sigma}_{\theta_1}^2\rangle \approx& \langle \alpha^2\Delta_T^2 \mathrm{Tr}(\hat{\Sigma}^2(\hat{\Sigma} + \hat{R})^{-4}) + e^{4T}\hat{R}^4 \mathrm{Tr}(\hat{\Sigma} + \hat{R})^{-4} + 4e^{4T}\hat{R}^2 \mathrm{Tr}(\hat{\Sigma} + \hat{R})^{-2} \\ &+ 2\alpha\Delta_T e^{2T}\hat{R}^2 \mathrm{Tr}(\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-4}) - 4\alpha\Delta_T e^{2T}\hat{R}\mathrm{Tr}\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-3} - 4e^{4T}\hat{R}^3 \mathrm{Tr}(\hat{\Sigma} + \hat{R})^{-3} \\ &+ 2\alpha\Delta_T e^{2T} \mathrm{Tr}(\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-2})\rangle\end{aligned} \tag{C.21}$$

All these expectation values can be calculated from a generic term of the form for integer $a > 0, b \geq 0$

$$\begin{aligned}C_{a,b} = \langle \mathrm{Tr}(\hat{\Sigma}^a(\hat{\Sigma} + \hat{R})^{-(a+b)})\rangle &= \frac{(-1)^b}{a(a+1)\dots(a+b-1)}\partial_{\hat{R}}^b \langle \mathrm{Tr}(\hat{\Sigma}^a(\hat{\Sigma} + \hat{R})^{-a})\rangle \\ &= d\frac{\Gamma(a)}{\Gamma(a+b)}(-\partial_{\hat{R}})^b d\mathrm{f}_{\hat{\Sigma}}^a(\hat{R})\end{aligned} \tag{C.22}$$

Another identity that is useful is the following

$$\begin{aligned}B_a &= \langle \mathrm{Tr}\hat{R}^a(\hat{\Sigma} + \hat{R})^{-a}\rangle \\ &= \langle \mathrm{Tr}(\hat{R}^{a-1}(\hat{\Sigma} + \hat{R})^{-(a-1)} - \hat{R}^{a-1}\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-a})\rangle \\ &= \langle \mathrm{Tr}(1 - \sum_{i=1}^{a} \hat{R}^{a-i}\hat{\Sigma}(\hat{\Sigma} + \hat{R})^{-(a-i+1)})\rangle \\ &= d - \sum_{i=1}^{a} \hat{R}^{a-i} C_{1,a-i}\end{aligned} \tag{C.23}$$

Now we turn to calculate expression for the symbols defined above. To get an explicit formula for $C_{a,b}$ first we replace the derivative with respect to $\hat{R}$ by a derivative with respect to R with the chain rule given in the second equation on (C.17). Next we use the recursion relation in the first equation on (C.17) to express everything in terms of $d\mathrm{f}^1_{\hat{\Sigma}}(\hat{R}) \simeq d\mathrm{f}^1_{\Sigma}(R)$. Finally to perform the derivatives we use the self-consistency equation of the ridge parameter given in the last equation on (C.11). Finally we use (C.15). Once $C_{a,b}$s are computed we use the recursion relation to compute B_as. The expression for these quantities for $\alpha > 1$ are complicated. They are given as follows

$$
\begin{aligned}
C_{1,1} &= \frac{2d\sigma_X^2\left(R+\sigma_X^2\right)^2}{\left|-\left((\alpha-1)\sigma_X^4\right)+2R\sigma_X^2+R^2\right|\left(\left|-\left((\alpha-1)\sigma_X^4\right)+2R\sigma_X^2+R^2\right|+R^2+2R\sigma_X^2+(\alpha+1)\sigma_X^4\right)}\\
C_{1,2} &= \frac{d\sigma_X^2\left(R+\sigma_X^2\right)^3}{\left|-\alpha\sigma_X^4+\sigma_X^4+2R\sigma_X^2+R^2\right|^3}\\
C_{1,3} &= \frac{d\sigma_X^2\left(R+\sigma_X^2\right)^4\left(R^2+2R\sigma_X^2+(\alpha+1)\sigma_X^4\right)\mathrm{sgn}\left(-\left((\alpha-1)\sigma_X^4\right)+2R\sigma_X^2+R^2\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}\\
C_{2,1} &= \begin{cases} \frac{d\left(R+\sigma_X^2\right)^3\left(3R^2\sigma_X^2+R^3-3(\alpha-1)R\sigma_X^4+(\alpha-1)^2\sigma_X^6\right)}{\alpha\sigma_X^2\left(-R^2-2R\sigma_X^2+(\alpha-1)\sigma_X^4\right)^3} & \text{When } R\left(R+2\sigma_X^2\right)<(\alpha-1)\sigma_X^4 \\ \frac{d\sigma_X^4\left((\alpha+1)R^3+3R^2\sigma_X^2-3(\alpha-1)R\sigma_X^4+(\alpha-1)^2\sigma_X^6\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^3} & \text{Otherwise} \end{cases}\\
C_{2,2} &= \frac{d\sigma_X^4\left(R+\sigma_X^2\right)^4\left((\alpha+1)R^2-2(\alpha-1)R\sigma_X^2+(\alpha-1)^2\sigma_X^4\right)\mathrm{sgn}\left(-\left((\alpha-1)\sigma_X^4\right)+2R\sigma_X^2+R^2\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}\\
C_{2,3} &= \frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^7}(d\sigma_X^4\left(R+\sigma_X^2\right)^5((\alpha+1)R^4+3((\alpha-2)\alpha+2)R^2\sigma_X^4-(\alpha-4)R^3\sigma_X^2\\
&\quad+(\alpha-4)(\alpha-1)R\sigma_X^6+(\alpha-1)^2(\alpha+1)\sigma_X^8)\mathrm{sgn}\left(-\left((\alpha-1)\sigma_X^4\right)+2R\sigma_X^2+R^2\right))\\
C_{3,1} &= \begin{cases} \frac{1}{\alpha\sigma_X^2\left(-R^2-2R\sigma_X^2+(\alpha-1)\sigma_X^4\right)^5}(d\left(R+\sigma_X^2\right)^4((\alpha-1)^2(\alpha+15)R^2\sigma_X^8-20(\alpha-1)R^3\sigma_X^6 \\ \quad -5(\alpha-3)R^4\sigma_X^4+6R^5\sigma_X^2+R^6-6(\alpha-1)^3R\sigma_X^{10} \\ \quad +(\alpha-1)^4\sigma_X^{12})) & \text{When } R\left(R+2\sigma_X^2\right)<(\alpha-1)\sigma_X^4 \\ \frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}(d\sigma_X^6((\alpha(\alpha+3)+1)R^6+(\alpha-1)^2(\alpha+15)R^2\sigma_X^8 \\ \quad +4(\alpha-1)((\alpha-1)\alpha-5)R^3\sigma_X^6 \\ +(\alpha((\alpha-12)\alpha+6)+15)R^4\sigma_X^4-2((\alpha-5)\alpha-3)R^5\sigma_X^2-6(\alpha-1)^3R\sigma_X^{10}+(\alpha-1)^4\sigma_X^{12})) & \end{cases}\\
C_{3,2} &= \frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^7}(d\sigma_X^6\left(R+\sigma_X^2\right)^5((\alpha(\alpha+3)+1)R^4+3\left(\alpha^3-3\alpha+2\right)R^2\sigma_X^4\\
&\quad+2\left(-3\alpha^2+\alpha+2\right)R^3\sigma_X^2-4(\alpha-1)^3R\sigma_X^6+(\alpha-1)^4\sigma_X^8)\mathrm{sgn}\left(-\alpha\sigma_X^4+\sigma_X^4+2R\sigma_X^2+R^2\right))\\
C_{3,3} &= \frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^9}(d\sigma_X^6\left(R+\sigma_X^2\right)^6((\alpha(\alpha+3)+1)R^6+6\left(-\alpha^2+\alpha+1\right)R^5\sigma_X^2\\
&\quad+3(\alpha-1)^2(\alpha(2\alpha-3)+5)R^2\sigma_X^8-4(\alpha-1)(2(\alpha-2)\alpha+5)R^3\sigma_X^6+3(\alpha(2(\alpha-1)\alpha-3)+5)R^4\sigma_X^4\\
&\quad-6(\alpha-1)^3R\sigma_X^{10}+(\alpha-1)^4(\alpha+1)\sigma_X^{12})\mathrm{sgn}\left(-\alpha\sigma_X^4+\sigma_X^4+2R\sigma_X^2+R^2\right))
\end{aligned}
\tag{C.24}
$$

$$
\begin{aligned}
B_1 &= \begin{cases} \frac{dR}{R+\sigma_X^2} & \text{When } R\left(R+2\sigma_X^2\right) \geq (\alpha-1)\sigma_X^4 \\ d - \frac{d\left(\frac{R}{\sigma_X^2}+1\right)}{\alpha} & \text{Otherwise} \end{cases} \\
B_2 &= \begin{cases} \frac{dR^2}{R^2+2R\sigma_X^2-\alpha\sigma_X^4+\sigma_X^4} & \text{When } R\left(R+2\sigma_X^2\right) \geq (\alpha-1)\sigma_X^4 \\ d\left(-\frac{1}{\alpha} - \frac{R^2}{R^2+2R\sigma_X^2-\alpha\sigma_X^4+\sigma_X^4}+1\right) & \text{Otherwise} \end{cases} \\
B_3 &= \begin{cases} \frac{dR^3\left(3R^2\sigma_X^2+R^3+3R\sigma_X^4-\left(\left(\alpha^2-1\right)\sigma_X^6\right)\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^3} & \text{When } R\left(R+2\sigma_X^2\right) \geq (\alpha-1)\sigma_X^4 \\ \frac{d\left(R-(\alpha-1)\sigma_X^2\right)^3\left(3R^2\sigma_X^2+R^3+3R\sigma_X^4-\left((\alpha-1)\sigma_X^6\right)\right)}{\alpha\left(-R^2-2R\sigma_X^2+(\alpha-1)\sigma_X^4\right)^3} & \text{Otherwise} \end{cases} \\
B_4 &= \begin{cases} \frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}(dR^4(4\left(-\alpha^2+\alpha+5\right)R^3\sigma_X^6+(\alpha((\alpha-12)\alpha+6)+15)R^2\sigma_X^8 \\ \quad +(\alpha+15)R^4\sigma_X^4+6R^5\sigma_X^2+R^6+2(\alpha((\alpha-6)\alpha+2)+3)R\sigma_X^{10} \\ \quad +(\alpha-1)^2(\alpha(\alpha+3)+1)\sigma_X^{12})) & \text{When } R\left(R+2\sigma_X^2\right) \geq (\alpha-1)\sigma_X^4 \\ \frac{d\left(R-(\alpha-1)\sigma_X^2\right)^4\left(-5(\alpha-3)R^2\sigma_X^8+(\alpha+15)R^4\sigma_X^4+20R^3\sigma_X^6+6R^5\sigma_X^2+R^6-6(\alpha-1)R\sigma_X^{10}+(\alpha-1)^2\sigma_X^{12}\right)}{\alpha\left(-R^2-2R\sigma_X^2+(\alpha-1)\sigma_X^4\right)^5} & \text{Otherwise} \end{cases}
\end{aligned}
\tag{C.25}
$$

Explicit expression of some of these symbols that will be required later is given below for $\alpha < 1$.

$$
\begin{aligned}
B_1 &= \frac{dR}{R+\sigma_X^2} \\
B_2 &= \frac{dR^2}{R^2+2R\sigma_X^2-\alpha\sigma_X^4+\sigma_X^4} \\
B_3 &= \frac{dR^3\left(3R^2\sigma_X^2+R^3+3R\sigma_X^4-\left(\left(\alpha^2-1\right)\sigma_X^6\right)\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^3} \\
B_4 &= \frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}(dR^4(4\left(-\alpha^2+\alpha+5\right)R^3\sigma_X^6+(\alpha((\alpha-12)\alpha+6)+15)R^2\sigma_X^8 \\
&\quad +(\alpha+15)R^4\sigma_X^4+6R^5\sigma_X^2+R^6+2(\alpha((\alpha-6)\alpha+2)+3)R\sigma_X^{10}+(\alpha-1)^2(\alpha(\alpha+3)+1)\sigma_X^{12}))
\end{aligned}
\tag{C.26}
$$

$$
\begin{aligned}
C_{1,1} &= \frac{d\sigma_X^2}{R^2+2R\sigma_X^2-\alpha\sigma_X^4+\sigma_X^4} \\
C_{1,2} &= \frac{d\sigma_X^2\left(R+\sigma_X^2\right)^3}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^3} \\
C_{1,3} &= \frac{d\sigma_X^2\left(R+\sigma_X^2\right)^4\left(R^2+2R\sigma_X^2+(\alpha+1)\sigma_X^4\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5} \\
C_{2,1} &= \frac{d\sigma_X^4\left((\alpha+1)R^3+3R^2\sigma_X^2-3(\alpha-1)R\sigma_X^4+(\alpha-1)^2\sigma_X^6\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^3} \\
C_{2,2} &= \frac{d\sigma_X^4\left(R+\sigma_X^2\right)^4\left((\alpha+1)R^2-2(\alpha-1)R\sigma_X^2+(\alpha-1)^2\sigma_X^4\right)}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}
\end{aligned}
\tag{C.27}
$$

$$
\begin{aligned}
C_{2,3} = &\frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^7} d\sigma_X^4\left(R+\sigma_X^2\right)^5((\alpha+1)R^4+3((\alpha-2)\alpha+2)R^2\sigma_X^4 \\
&-(\alpha-4)R^3\sigma_X^2+(\alpha-4)(\alpha-1)R\sigma_X^6+(\alpha-1)^2(\alpha+1)\sigma_X^8) \\
C_{3,1} = &\frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^5}(d\sigma_X^6((\alpha(\alpha+3)+1)R^6+(\alpha-1)^2(\alpha+15)R^2\sigma_X^8 \\
&+4(\alpha-1)((\alpha-1)\alpha-5)R^3\sigma_X^6+(\alpha((\alpha-12)\alpha+6)+15)R^4\sigma_X^4-2((\alpha-5)\alpha-3)R^5\sigma_X^2 \\
&-6(\alpha-1)^3R\sigma_X^{10}+(\alpha-1)^4\sigma_X^{12})) \\
C_{3,2} = &\frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^7} d\sigma_X^6\left(R+\sigma_X^2\right)^5((\alpha(\alpha+3)+1)R^4+3\left(\alpha^3-3\alpha+2\right)R^2\sigma_X^4 \\
&+2\left(-3\alpha^2+\alpha+2\right)R^3\sigma_X^2-4(\alpha-1)^3R\sigma_X^6+(\alpha-1)^4\sigma_X^8) \\
C_{3,3} = &\frac{1}{\left(R^2+2R\sigma_X^2-\left((\alpha-1)\sigma_X^4\right)\right)^9}(d\sigma_X^6\left(R+\sigma_X^2\right)^6((\alpha(\alpha+3)+1)R^6+6\left(-\alpha^2+\alpha+1\right)R^5\sigma_X^2 \\
&+3(\alpha-1)^2(\alpha(2\alpha-3)+5)R^2\sigma_X^8-4(\alpha-1)(2(\alpha-2)\alpha+5)R^3\sigma_X^6 \\
&+3(\alpha(2(\alpha-1)\alpha-3)+5)R^4\sigma_X^4-6(\alpha-1)^3R\sigma_X^{10}+(\alpha-1)^4(\alpha+1)\sigma_X^{12}))
\end{aligned}
\tag{C.28}
$$

In terms of these symbols we have the following explicit formula

$$
\begin{aligned}
\mathrm{Tr}\langle\hat{\Sigma}_{\theta_1}\rangle =&\alpha\Delta_T C_{1,1}+e^{2T}(B_2-2B_1) \\
\mathrm{Tr}\langle\hat{\Sigma}_{\theta_1}^2\rangle =&\alpha^2\Delta_T^2 C_{2,2}+e^{4T}B_4+4e^{4T}B_2 \\
&+2\alpha\Delta_T e^{2T}\hat{R}^2C_{1,3}-4\alpha\Delta_T e^{2T}\hat{R}C_{1,2}-4e^{4T}B_3+2\alpha\Delta_T e^{2T}C_{1,1}
\end{aligned}
\tag{C.29}
$$

As a summary our final expression for variance term in KL divergence is given by (C.10) along with (C.17),(C.22),(C.23) and (C.29). Since the expression is fairly complicated we won't present explicit formula for it. To understand the implications of the formula we look at ridgeless limit $\hat{R}\to 0$.

Putting all the results together, for $\alpha<1$, the ridgeless formula takes the following form

$$
\begin{aligned}
\langle \mathrm{KL}(\rho_G|\rho)_{\mathrm{var}}\rangle =&\frac{1}{4}(e^{-2T}+\sigma^{-2}\Delta_T)^2(2e^{2T}\frac{\alpha}{1-\alpha}\Delta_T d\,\sigma_X^{-2}+\frac{\alpha^2}{(1-\alpha)^3}\Delta_T^2 d\,\sigma_X^{-4}) \\
&+\frac{1}{2}(e^{-2T}+\sigma^{-2}\Delta_T)(\sigma^{-2}\Delta_T e^{2T})(\frac{\alpha}{1-\alpha}\Delta_T d\,\sigma_X^{-2})+\frac{d}{4}(\sigma^{-2}e^{2T}\Delta_T)^2 \\
=&\frac{d\alpha\hat{\lambda}e^{-4T}\left(e^{2T}-1\right)}{2(1-\alpha)}+\frac{d\hat{\lambda}^2e^{-8T}\left(e^{2T}-1\right)^2\left(\alpha^2+(1-\alpha)^3e^{4T}+4\alpha(1-\alpha)^2e^{2T}\right)}{4(1-\alpha)^3}
\end{aligned}
\tag{C.30}
$$

If we further consider $\alpha\to 0$ approximation we see that $\langle \mathrm{KL}(\rho_G|\rho)_{\mathrm{var}}\rangle\propto d\lambda^2$ to the leading order. Also note that $\langle \mathrm{KL}(\rho_G|\rho)_{\mathrm{var}}\rangle/d$ is an increasing function of α in this regime.

Ridgeless limit for $\alpha > 1$ is more involved and it is given by

$$\begin{aligned}\langle \mathrm{KL}(\rho_G|\rho)_{\mathrm{var}}\rangle =& \frac{1}{4}(e^{-2T}+\sigma^{-2}\Delta_T)^2(2e^{2T}\frac{1}{(\alpha-1)}\Delta_T d\,\sigma_X^{-2}+\frac{\alpha^2}{(\alpha-1)^3}\Delta_T^2 d\,\sigma_X^{-4}+e^{4T}d\left(1-\frac{1}{\alpha}\right))\\ &+\frac{1}{2}(e^{-2T}+\sigma^{-2}\Delta_T)(\sigma^{-2}\Delta_T e^{2T})(\frac{1}{\alpha-1}\Delta_T d\,\sigma_X^{-2}-e^{2T}d\left(1-\frac{1}{\alpha}\right))+\frac{d}{4}(\sigma^{-2}e^{2T}\Delta_T)^2\\ =& d\frac{\alpha-1}{4\alpha}+\frac{d\hat{\lambda}e^{-4T}\left(e^{2T}-1\right)}{2(\alpha-1)}+\frac{d\hat{\lambda}^2e^{-8T}\left(e^{2T}-1\right)^2\left(\alpha^3+(\alpha-1)^3e^{4T}+4\alpha(\alpha-1)^2e^{2T}\right)}{4(\alpha-1)^3\alpha}\end{aligned} \tag{C.31}$$

In this domain $\langle \mathrm{KL}(\rho_G|\rho)_{\mathrm{var}}\rangle/d$ is no longer a monotonic function of α. It is easy to see from the expression above that as $\alpha \to 1$ both from $\alpha > 1$ and $\alpha < 1$ side, KL divergence becomes unbounded.

References

[1] Ben Adlam and Jeffrey Pennington. The neural tangent kernel in high dimensions: Triple descent and a multi-scale theory of generalization. In *International Conference on Machine Learning*, pages 74–84. PMLR, 2020.

[2] Madhu S Advani, Andrew M Saxe, and Haim Sompolinsky. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, 2020.

[3] Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions, 2023.

[4] Alexander Atanasov, Blake Bordelon, Sabarish Sainathan, and Cengiz Pehlevan. The onset of variance-limited behavior for networks in the lazy and rich regimes. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JLINxPOVTh7.

[5] Alexander Atanasov, Jacob A. Zavatone-Veth, and Cengiz Pehlevan. Scaling and renormalization in high-dimensional regression, 2024. URL https://arxiv.org/abs/2405.00592.

[6] Francis Bach. High-dimensional analysis of double descent for linear regression with random projections. *SIAM Journal on Mathematics of Data Science*, 6(1):26–50, 2024.

[7] Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining neural scaling laws. *arXiv preprint arXiv:2102.06701*, 2021.

[8] Joe Benton, Valentin De Bortoli, Arnaud Doucet, and George Deligiannidis. Nearly d-linear convergence bounds for diffusion models via stochastic localization, 2024. URL https://arxiv.org/abs/2308.03686.

[9] Blake Bordelon and Cengiz Pehlevan. Self-consistent dynamical field theory of kernel evolution in wide neural networks. *Advances in Neural Information Processing Systems*, 35:32240–32256, 2022.

[10] Blake Bordelon and Cengiz Pehlevan. Dynamics of finite width kernel and prediction fluctuations in mean field neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

[11] Blake Bordelon, Abdulkadir Canatar, and Cengiz Pehlevan. Spectrum dependent learning curves in kernel regression and wide neural networks. In *International Conference on Machine Learning*, pages 1024–1034. PMLR, 2020.

[12] Blake Bordelon, Abdulkadir Canatar, and Cengiz Pehlevan. Spectrum dependent learning curves in kernel regression and wide neural networks, 2021. URL https://arxiv.org/abs/2002.02561.

[13] Arwen Bradley and Preetum Nakkiran. Classifier-free guidance is a predictor-corrector, 2024. URL https://arxiv.org/abs/2408.09000.

[14] Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature Communications*, 12(1), May 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-23103-1. URL http://dx.doi.org/10.1038/s41467-021-23103-1.

[15] Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914, 2021.

[16] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5253–5270, 2023.

[17] Hongrui Chen, Holden Lee, and Jianfeng Lu. Improved analysis of score-based generative modeling: User-friendly bounds under minimal smoothness assumptions, 2023. URL https://arxiv.org/abs/2211.01916.

[18] Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru R. Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions, 2023. URL https://arxiv.org/abs/2209.11215.

[19] Ting Chen. On the importance of noise scheduling for diffusion models, 2023. URL https://arxiv.org/abs/2301.10972.

[20] Muthu Chidambaram, Khashayar Gatmiry, Sitan Chen, Holden Lee, and Jianfeng Lu. What does guidance do? a fine-grained analysis in a simple setting, 2024. URL https://arxiv.org/abs/2409.13074.

[21] Stéphane d'Ascoli, Levent Sagun, and Giulio Biroli. Triple descent and the two kinds of overfitting: Where & why do they appear? *Advances in Neural Information Processing Systems*, 33:3058–3069, 2020.

[22] Mehmet Demirtas, James Halverson, Anindita Maiti, Matthew D. Schwartz, and Keegan Stoner. Neural network field theories: Non-gaussianity, actions, and locality, 2023. URL https://arxiv.org/abs/2307.03223.

[23] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. URL https://arxiv.org/abs/2105.05233.

[24] Oussama Dhifallah and Yue M Lu. A precise performance analysis of learning with random features. *arXiv preprint arXiv:2008.11904*, 2020.

[25] Lee H. Dicker. Ridge regression and asymptotic minimax estimation over spheres of growing dimension. *Bernoulli*, 22(1):1 – 37, 2016. doi: 10.3150/14-BEJ609. URL https://doi.org/10.3150/14-BEJ609.

[26] Edgar Dobriban and Stefan Wager. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247 – 279, 2018. doi: 10.1214/17-AOS1549. URL https://doi.org/10.1214/17-AOS1549.

[27] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

[28] Stéphane D'Ascoli, Maria Refinetti, Giulio Biroli, and Florent Krzakala. Double Trouble in Double Descent: Bias and Variance(s) in the Lazy Regime. In *International Conference on Machine Learning*, pages 2280–2290. PMLR, November 2020. URL http://proceedings.mlr.press/v119/d-ascoli20a.html. ISSN: 2640-3498.

[29] Alessandro Favero, Antonio Sclocchi, Francesco Cagnetta, Pascal Frossard, and Matthieu Wyart. How compositional generalization and creativity improve as diffusion models are trained. *arXiv preprint arXiv:2502.12089*, 2025.

[30] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *The Annals of Statistics*, 50(2):949–986, 2022.

[31] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

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

[33] Hong Hu and Yue M Lu. Universality laws for high-dimensional learning with random features. *IEEE Transactions on Information Theory*, 69(3):1932–1964, 2022.

[34] Chin-Wei Huang, Jae Hyun Lim, and Aaron Courville. A variational perspective on diffusion-based generative models and score matching, 2021. URL https://arxiv.org/abs/2106.02808.

[35] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html.

[36] Allan Jabri, David Fleet, and Ting Chen. Scalable adaptive computation for iterative generation, 2023. URL https://arxiv.org/abs/2212.11972.

[37] Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf.

[38] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks, 2020. URL https://arxiv.org/abs/1806.07572.

[39] Zahra Kadkhodaie and Eero Peter Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. In *NeurIPS 2020 Workshop on Deep Learning and Inverse Problems*, 2020. URL https://openreview.net/forum?id=RLN7K4U3UST.

[40] Zahra Kadkhodaie, Florentin Guth, Eero P Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. *arXiv preprint arXiv:2310.02557*, 2023.

[41] Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. *arXiv preprint arXiv:2412.20292*, 2024.

[42] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022. URL https://arxiv.org/abs/2206.00364.

[43] Anders Krogh and John A Hertz. Generalization in a linear perceptron in the presence of noise. *Journal of Physics A: Mathematical and General*, 25(5):1135, 1992.

[44] Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence for score-based generative modeling with polynomial complexity, 2023. URL https://arxiv.org/abs/2206.06227.

[45] Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/0d1a9651497a38d8b1c3871c84528bd4-Paper.pdf.

[46] Marvin Li and Sitan Chen. Critical windows: non-asymptotic theory for feature emergence in diffusion models. In *International Conference on Machine Learning*, pages 27474–27498. PMLR, 2024.

[47] Cosme Louart, Zhenyu Liao, and Romain Couillet. A random matrix approach to neural networks. *The Annals of Applied Probability*, 28(2):1190–1248, 2018.

[48] Bruno Loureiro, Cedric Gerbelot, Hugo Cui, Sebastian Goldt, Florent Krzakala, Marc Mezard, and Lenka Zdeborová. Learning curves of generic features maps for realistic datasets with a teacher-student model. *Advances in Neural Information Processing Systems*, 34:18137–18151, 2021.

[49] Alexander Maloney, Daniel A Roberts, and James Sully. A solvable model of neural scaling laws. *arXiv preprint arXiv:2210.16859*, 2022.

[50] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33), July 2018. ISSN 1091-6490. doi: 10.1073/pnas.1806579115. URL http://dx.doi.org/10.1073/pnas.1806579115.

[51] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Generalization error of random feature and kernel methods: hypercontractivity and kernel matrix concentration. *Applied and Computational Harmonic Analysis*, 59:3–84, 2022.

[52] Preetum Nakkiran. More data can hurt for linear regression: Sample-wise double descent. *arXiv preprint arXiv:1912.07242*, 2019.

[53] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models, 2021. URL https://arxiv.org/abs/2102.09672.

[54] Maya Okawa, Ekdeep Singh Lubana, Robert P. Dick, and Hidenori Tanaka. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task. *arXiv preprint arXiv:2310.09336*, 2023.

[55] OpenAI. Sora. 2024. URL https://openai.com/index/sora/.

[56] Core Francisco Park, Maya Okawa, Andrew Lee, Ekdeep S Lubana, and Hidenori Tanaka. Emergence of hidden capabilities: Exploring learning dynamics in concept space. *Advances in Neural Information Processing Systems*, 37:84698–84729, 2024.

[57] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[58] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. URL https://arxiv.org/abs/2204.06125.

[59] Daniel A. Roberts, Sho Yaida, and Boris Hanin. *The Principles of Deep Learning Theory: An Effective Theory Approach to Understanding Neural Networks*. Cambridge University Press, May 2022. ISBN 9781316519332. doi: 10.1017/9781009023405. URL http://dx.doi.org/10.1017/9781009023405.

[60] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

[61] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. URL https://arxiv.org/abs/1505.04597.

[62] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. URL https://arxiv.org/abs/2205.11487.

[63] Kulin Shah, Sitan Chen, and Adam Klivans. Learning mixtures of gaussians using the ddpm objective, 2023. URL https://arxiv.org/abs/2307.01178.

[64] James B Simon, Madeline Dickens, Dhruva Karkada, and Michael Deweese. The eigenlearning framework: A conservation law perspective on kernel ridge regression and wide neural networks. *Transactions on Machine Learning Research*, 2023.

[65] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. URL https://arxiv.org/abs/1503.03585.

[66] Peter Sollich. Learning curves for Gaussian processes. *Advances in neural information processing systems*, 11, 1998.

[67] Peter Sollich and Anason Halees. Learning curves for Gaussian process regression: Approximations and bounds. *Neural computation*, 14(6):1393–1428, 2002.

[68] G Somepalli, V Singla, M Goldblum, J Geiping, and T Goldstein. Diffusion art or digital forgery. *Investigating Data Replication in Diffusion Models*, 2022.

[69] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

[70] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[71] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.

[72] Stefano Spigler, Mario Geiger, and Matthieu Wyart. Asymptotic learning curves of kernel methods: empirical data versus teacher–student paradigm. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12):124001, 2020.

[73] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. In *European conference on computer vision*, pages 459–479. Springer, 2022.

[74] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

[75] Yuchen Wu, Minshuo Chen, Zihao Li, Mengdi Wang, and Yuting Wei. Theoretical insights for diffusion guidance: A case study for gaussian mixture models, 2024. URL https://arxiv.org/abs/2403.01639.

[76] Greg Yang and Edward J. Hu. Tensor programs iv: Feature learning in infinite-width neural networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 11727–11737. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/yang21c.html.

[77] TaeHo Yoon, Joo Young Choi, Sehyun Kwon, and Ernest K. Ryu. Diffusion probabilistic models generalize when they fail to memorize. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023. URL https://openreview.net/forum?id=shciCbSk9h.

[78] Jacob A Zavatone-Veth and Cengiz Pehlevan. Learning curves for deep structured Gaussian feature models. In *Advances in Neural Information Processing Systems*, 2023.

[79] Huijie Zhang, Jinfan Zhou, Yifu Lu, Minzhe Guo, Peng Wang, Liyue Shen, and Qing Qu. The emergence of reproducibility and generalizability in diffusion models. *arXiv preprint arXiv:2310.05264*, 2023.

[80] Qinsheng Zhang and Yongxin Chen. Diffusion normalizing flow, 2021. URL https://arxiv.org/abs/2110.07579.