U-NETS AS BELIEF PROPAGATION: EFFICIENT CLASSIFICATION, DENOISING, AND DIFFUSION IN GENERATIVE HIERARCHICAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

U-Nets are among the most widely used architectures in computer vision, renowned for their exceptional performance in applications such as image segmentation, denoising, and diffusion modeling. However, a theoretical explanation of the U-Net architecture design has not yet been fully established.

This paper introduces a novel interpretation of the U-Net architecture by studying certain generative hierarchical models, which are tree-structured graphical models extensively utilized in both language and image domains. With their encoder-decoder structure, long skip connections, and pooling and up-sampling layers, we demonstrate how U-Nets can naturally implement the belief propagation denoising algorithm in such generative hierarchical models, thereby efficiently approximating the denoising functions. This leads to an efficient sample complexity bound for learning the denoising function using U-Nets within these models. Additionally, we discuss the broader implications of these findings for diffusion models in generative hierarchical models. We also demonstrate that the conventional architecture of convolutional neural networks (ConvNets) is ideally suited for classification tasks within these models. This offers a unified view of the roles of ConvNets and U-Nets, highlighting the versatility of generative hierarchical models in modeling complex data distributions.

1 INTRODUCTION

U-Nets are one of the most prominent network architectures in computer vision, primarily employed for tasks such as image segmentation, denoising (Ronneberger et al., 2015; Zhou et al., 2018; Siddique et al., 2021; Oktay et al., 2018), and diffusion modeling (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019; Song et al., 2020). These networks are structured as encoder-decoder convolutional neural networks equipped with long skip connections, and their input and output typically maintain the same dimensions. While U-Nets have demonstrated exceptional performance across a variety of applications, the theoretical foundations of their key components—including the encoder-decoder structure, long skip connections, and the pooling and up-sampling layers—remain inadequately understood. Notably, long skip connections have a significant impact on performance as shown in empirical studies (Drozdzal et al., 2016; Wang et al., 2022). Existing explanations, often anecdotal, suggest their efficacy stems from improved information propagation and reduction of the vanishing gradient issue, but a thorough theoretical exploration is still lacking.

In this paper, we introduce a novel interpretation of the U-Net architecture, viewing it through the lens of neural network approximation. We posit that:

U-Nets naturally approximate the belief propagation denoising algorithm in certain generative hierarchical models.

The generative hierarchical model discussed herein is a tree-structured graphical model, which has been widely employed in language and image generative modeling (Chomsky, 1959; Lee, 1996; Allen-Zhu & Li, 2023; Li et al., 2000; Willsky, 2002; Jin & Geman, 2006). We detail the precise definition of such generative hierarchical models in Section 2. A series of recent work (Mossel, 2016; Sclocchi et al., 2024; Tomasini & Wyart, 2024; Petrini et al., 2023; Kadkhodaie et al., 2023a)

have pioneered the use of generative hierarchical models in studying classification tasks and diffusion models. Kadkhodaie et al. (2023a) has empirically shown that U-Nets can effectively learn the denoising function for these models. Furthermore, as noted by Sclocchi et al. (2024), the belief propagation denoising algorithm, which computes the denoising function in these models exactly, includes a downward process and an upward process, with the latter reusing the intermediate results from the downward process. In Section 4, we demonstrate how the belief propagation algorithm naturally induces the encoder-decoder structure, the long skip connections, and the pooling and upsampling operations of the U-Nets. This gives rise to an efficient sample complexity bound for learning the denoising function in generative hierarchical models using U-Nets.

In addition to our main findings, in Section 3, we demonstrate that the standard architecture of convolutional neural networks (ConvNets) is well-suited for classification tasks within the same generative hierarchical model. We provide efficient sample complexity results to support this assertion. This offers a unified perspective on the role of both ConvNets and U-Nets in image classification and denoising tasks, and also highlights the versatility of generative hierarchical models in modeling data distributions across language and image domains.

2 THE GENERATIVE HIERARCHICAL MODEL

To define the generative hierarchical model, we start by introducing some key notations. Consider a tree $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ with a height of L, where we conventionally designate the root of the tree r as layer 0. For each node $v \in \mathcal{V}$, we denote $\mathrm{pa}(v)$ as the parent of v, $\mathcal{C}(v)$ as the children of v, and $\mathcal{N}(v)$ as the siblings of v. We denote $\mathcal{V}^{(\ell)}$ as the set of nodes at layer ℓ. We assume that for any $v \in \mathcal{V}^{(\ell-1)}$, the number of children is precisely $m^{(\ell)}$ for $\ell \in [L]$. The leaf nodes $v \in \mathcal{V}^{(L)}$ have no children. Additionally, for each $v \in \mathcal{V}^{(\ell)}$, we assume an ordering function (a bijection) $\iota : \mathcal{C}(v) \to [m^{(\ell)}]$, ensuring that any child $v' \in \mathcal{C}(v)$ possesses a unique rank $\iota(v') \in [m^{(\ell)}]$. We denote the number of nodes at layer ℓ as $d^{(\ell)}$, and the number of nodes at layer L as $d = d^{(L)}$. We further denote $\underline{m} = (m^{(\ell)})_{\ell \in [L]}$, and $\|\underline{m}\|_1 = \sum_{\ell=1}^{L} m^{(\ell)}$. By these definitions and assumptions, we have $d^{(\ell)} = \prod_{1 \le s \le \ell} m^{(s)}$, and $1 = d^{(0)} \le d^{(1)} \le \cdots \le d^{(L)} = d$.

For each layer $\ell = 0, \ldots, L$, every tree node $v \in \mathcal{V}^{(\ell)}$ is associated with a variable $x_v^{(\ell)} \in [S]$ for some $S \in \mathbb{N}_{\ge 2}$. (For simplicity, we use the same variable space $[S]$ across all layers, although our framework can accommodate variations across different layers ℓ.) We denote $\boldsymbol{x}^{(\ell)} = (x_v^{(\ell)})_{v \in \mathcal{V}^{(\ell)}} \in [S]^{d^{(\ell)}}$ as the variables at layer ℓ. The variables at the leaves, $\boldsymbol{x} = \boldsymbol{x}^{(L)} \in [S]^d$ are considered the observed covariates, exemplified by the pixel representation of an image. Conversely, the root node variable $y = x_{\mathrm{r}}^{(0)} \in [S]$ is treated as the associated label. Variables for the intermediate layers $\{\boldsymbol{x}^{(\ell)}\}_{1 \le \ell \le L-1}$ remain unobserved.

The generative hierarchical model. We consider a specific type of generative hierarchical model (GHM)[1] , which is a joint distribution $\mu_\star$ over variables

$$(y = x^{(0)} \in [S], \quad \boldsymbol{x}^{(1)} \in [S]^{d^{(1)}}, \quad \ldots, \quad \boldsymbol{x}^{(L-1)} \in [S]^{d^{(L-1)}}, \quad \boldsymbol{x}^{(L)} = \boldsymbol{x} \in [S]^d),$$

associated with a set of functions $\{\psi^{(\ell)} : [S] \times [S]^{m^{(\ell)}} \to \mathbb{R}_{\ge 0}\}_{\ell \in [L]}$, defined as

$$\begin{aligned} &\mu_\star(y, \boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(L-1)}, \boldsymbol{x}) \\ &\propto \psi^{(1)}(y, \boldsymbol{x}^{(1)}) \cdot \Big(\textstyle\prod_{v \in \mathcal{V}^{(1)}} \psi^{(2)}(x_v^{(1)}, x_{\mathcal{C}(v)}^{(2)}) \Big) \cdots \Big(\textstyle\prod_{v \in \mathcal{V}^{(L-1)}} \psi^{(L)}(x_v^{(L-1)}, x_{\mathcal{C}(v)}) \Big). \end{aligned} \tag{GHM}$$

The formula specifies that any two nodes $v_1, v_2 \in \mathcal{V}^{(\ell)}$ within the same level uses the same function $\psi^{(\ell)}$, thereby embedding specific invariance properties into $\mu_\star$. Consequently, this GHM ensures that $(x_v)_{v \succeq v_1} \overset{d}{=} (x_v)_{v \succeq v_2}$ for any $v_1, v_2 \in \mathcal{V}^{(\ell)}$. The notation $v \succeq v_1$ denotes that v is either

[1] We define "generative hierarchical models" as general probabilistic models with a hierarchical structure. The specific model discussed in this paper is an instance of such generative hierarchical models. These models are also known by various other names, including "hierarchical generative models", "latent hierarchical models", "Bayesian hierarchical models", "hierarchical Markov random fields", among others.

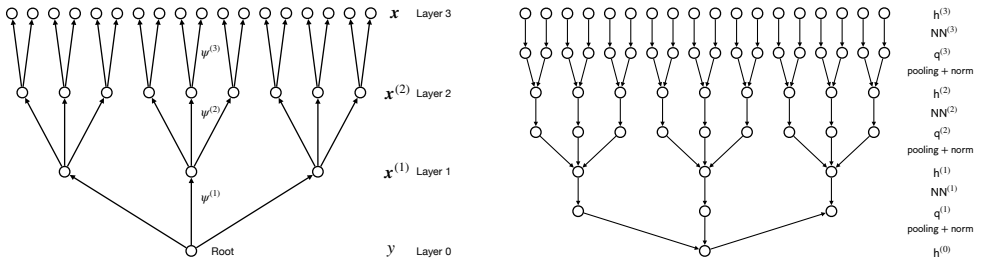

Figure 1: Left: The generative hierarchical model with 3 layers and $m^{(1)} = 3$, $m^{(2)} = 3$, and $m^{(3)} = 2$ children in each layer. Right: A 3-layer convolutional neural network.

identical to or a descendant of v_1. We will explore later how this invariance property interacts with the convolutional structure of the neural networks to be introduced.

Throughout the paper, we impose a specific assumption on the ψ functions in GHM, namely that each ψ function can be factorized into a product of functions that depend solely on the ordering of the child. This assumption is essential for enabling convolutional neural networks to approximate the associated belief propagation algorithm.

Assumption 1 (Factorization of ψ)**.** *For each layer $\ell \in [L]$ and node $v \in \mathcal{V}^{(\ell-1)}$, we have*

$$\psi^{(\ell)}(x_v^{(\ell-1)}, x_{\mathcal{C}(v)}^{(\ell)}) = \prod_{v' \in \mathcal{C}(v)} \psi_{\iota(v')}^{(\ell)}(x_v^{(\ell-1)}, x_{v'}^{(\ell)}). \tag{1}$$

We also state a technical assumption concerning the boundedness of these ψ functions of GHM.

Assumption 2 (Boundedness of ψ)**.** *For any layer $\ell \in [L]$ and child node rank $\iota \in [m^{(\ell)}]$, the transition probabilities are bounded as follows:*

$$1/K \le \min_{x,x'} \psi_\iota^{(\ell)}(x, x') \le \max_{x,x'} \psi_\iota^{(\ell)}(x, x') \le K. \tag{2}$$

It is helpful to think about the joint distribution $\mu_\star$ as a tree-structured Markov process, which admits the factorization

$$\begin{aligned} &\mu_\star(y, \boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(L-1)}, \boldsymbol{x}) \\ &= \mu_\star(y)\mu_\star(\boldsymbol{x}^{(1)}|y) \cdot \Big(\prod_{v \in \mathcal{V}^{(1)}} \mu_\star(x_{\mathcal{C}(v)}^{(2)}|x_v^{(1)})\Big) \cdots \Big(\prod_{v \in \mathcal{V}^{(L-1)}} \mu_\star(x_{\mathcal{C}(v)}|x_v^{(L-1)})\Big), \end{aligned} \tag{3}$$

where we abuse the notation to denote $\mu_\star(x_{\mathcal{C}(v)}^{(\ell)}|x_v^{(\ell-1)})$ as the conditional probability of $x_{\mathcal{C}(v)}^{(\ell)}$ given $x_v^{(\ell-1)}$, and $\mu_\star(y)$ as the marginal probability of y. Indeed, any graphical model specified as in Eq. (1) can be cast into the form of (3). Furthermore, It can be checked that Eq. (3) coincides with (GHM) upon taking $\psi^{(\ell)}(x_v^{(\ell-1)}, x_{\mathcal{C}(v)}^{(\ell)}) = \mu_\star(x_{\mathcal{C}(v)}^{(\ell)}|x_v^{(\ell-1)})$ for $\ell \ge 1$, and $\psi^{(1)}(y, \boldsymbol{x}^{(1)}) = \mu_\star(y)\mu_\star(\boldsymbol{x}^{(1)}|y)$. In this scenario, the factorization assumption (1) implies that $(x_{v'}^{\ell})_{v' \in \mathcal{C}(v)}$ given $x_v^{(\ell-1)}$ are conditionally independent. We include a schematic plot of the generative hierarchical model with 3 layers as in Figure 1(left).

GHMs as natural models for languages and images. In the field of linguistics, GHMs are very similar to context-free grammars (CFGs) (Chomsky, 1959; Lee, 1996; Allen-Zhu & Li, 2023). The generative process of a context-free grammar involves creating valid strings or sentences based on a given set of production rules (the ψ functions) that dictate how symbols can be extended to form new strings. Starting with an initial symbol (the label y), the generation proceeds iteratively by applying production rules until all symbols belong to the terminal set, thus forming a complete sentence (the covariate $\boldsymbol{x}$).

In computer vision, GHMs are often utilized to model natural images (Li et al., 2000; Willsky, 2002; Jin & Geman, 2006), where they are sometimes referred to as multi-resolution Markov models. The hypothetical image generation process begins with a high-level concept of the image (represented

by the label y), which is then iteratively refined using a production rule (the ψ function). This refinement continues through successive resolution levels until the detail reaches the pixel level, resulting in the final image (the covariate $\boldsymbol{x}$).

We remark that a series of studies (Sclocchi et al., 2024; Tomasini & Wyart, 2024; Petrini et al., 2023; Kadkhodaie et al., 2023a) have provided both theoretical and empirical evidence supporting the efficacy of GHMs as powerful tools for modeling the properties of combinatorial data.

3 The warm-up problem: Classification in GHMs

In this section, we consider the warm-up problem of classification within the GHM. In the subsequent section, we will investigate the denoising task, where the results and intuitions will be similar and parallel to those presented in this section.

In the classification task, consider the scenario where we observe a set of iid samples $\{(\boldsymbol{x}_i, y_i)\}_{i\in[n]}$ drawn from $\mu_\star$ under the GHM. Our objective is to learn a probabilistic classifier $\hat{\mu}(y|\boldsymbol{x})$ from this dataset. With a suitable loss function, the optimal classifier is the Bayes classifier $\mu_\star(y|\boldsymbol{x})$, which represents the true conditional probability of y given $\boldsymbol{x}$. We aim to examine the sample complexity of learning this classifier through empirical risk minimization over the class of convolutional networks.

The ConvNet architecture. We here introduce the convolutional neural network (ConvNet) architecture used for classification, represented as $\mu_{\rm NN}(\cdot|\boldsymbol{x}) \in \Delta([S])$ for input $\boldsymbol{x} \in [S]^d$. Initially, we set $\mathsf{h}_v^{(L)} = x_v \in [S]$ for each node $v \in \mathcal{V}^{(L)}$. The operational flow of the network unfolds as follows:

$$\begin{aligned}
\mathsf{q}_v^{(\ell)} &= \mathrm{NN}_{\iota(v)}^{(\ell)}(\mathsf{h}_v^{(\ell)}) \in \mathbb{R}^S, && \ell \in [L],\ \ v \in \mathcal{V}^{(\ell)},\\
\mathsf{h}_v^{(\ell-1)} &= \mathrm{normalize}\Big(\textstyle\sum_{v'\in\mathcal{C}(v)} \mathsf{q}_{v'}^{(\ell)}\Big) \in \mathbb{R}^S, && \ell \in [L],\ \ v \in \mathcal{V}^{(\ell-1)}, \qquad \text{(ConvNet)}\\
\mu_{\rm NN}(\cdot|\boldsymbol{x}) &= \mathrm{softmax}(\mathsf{h}_{\rm r}^{(0)}) \in \Delta([S]).
\end{aligned}$$

The functions $\mathrm{NN}_{\iota(v)}^{(\ell)} : \mathbb{R}^{S_{\rm in}^{(\ell)}} \to \mathbb{R}^S$ (which adjust their input dimensionality $S_{\rm in}^{(\ell)}$ based on layer depth, $S_{\rm in}^{(\ell)} = (S+1)\cdot 1\{\ell \neq L\} + 2\cdot 1\{\ell = L\}$) and $\mathrm{normalize} : \mathbb{R}^S \to \mathbb{R}^S$ are defined as follows:

$$\mathrm{NN}_\iota^{(\ell)}(\mathsf{h}) = W_{1,\iota}^{(\ell)} \cdot \mathrm{ReLU}(W_{2,\iota}^{(\ell)} \cdot \mathrm{ReLU}(W_{3,\iota}^{(\ell)} \cdot [\mathsf{h}; 1])), \tag{4}$$

$$\mathrm{normalize}(h)_s = h_s - \max_{s'\in[S]} h_{s'}. \tag{5}$$

The dimensions of the weight matrices within the ConvNet are specified as follows:

$$\boldsymbol{W} = \Big\{ \{W_{1,\iota}^{(\ell)} \in \mathbb{R}^{S\times D}\}_{\iota\in[m^{(\ell)}]}, \{W_{2,\iota}^{(\ell)} \in \mathbb{R}^{D\times D}\}_{\iota\in[m^{(\ell)}]}, \{W_{3,\iota}^{(\ell)} \in \mathbb{R}^{D\times S_{\rm in}^{(\ell)}}\}_{\iota\in[m^{(\ell)}]} \Big\}_{\ell\in[L]}. \tag{6}$$

Furthermore, we denote $\mu_{\rm NN}^{\boldsymbol{W}}$ as the ConvNet classifier $\mu_{\rm NN}$ parameterized by $\boldsymbol{W}$. A schematic illustration of a ConvNet with 3 layers is provided in Figure 1(right).

Remark 1 (An explanation of the "ConvNet" architecture)**.** *We remark that the neural network layers described in (ConvNet) are different from the "convolution operations" typically seen in practice. The convolution operations used in practice involve computing the inner products between convolutional filters and image patches, whereas in (ConvNet) and (4), a point-wise product is employed instead. Despite this, these layers are still referred to as convolutional layers because the mapping from $\{\mathsf{h}_v^{(\ell)}\}_{v\in\mathcal{V}^{(\ell)}}$ to $\{\mathsf{q}_v^{(\ell)}\}_{v\in\mathcal{V}^{(\ell)}}$, as per the first line of (ConvNet), preserves the translation-invariance property. Specifically, we use the same function $\mathrm{NN}_\iota^{(\ell)}$ across different inputs $\mathsf{h}_{v_1}^{(\ell)}$ and $\mathsf{h}_{v_2}^{(\ell)}$ as long as $\iota(v_1) = \iota(v_2) = \iota$.*

Additionally, the "normalization operator" defined in (5) differs from commonly used ones. We adopt this specific form for technical reasons, to effectively control the approximation error.

Despite these differences from standard convolutional networks, (ConvNet) represents an iterative composition of convolutional layers, pooling layers, and normalization layers, aligning closely with the architecture of convolutional networks used in practice. Figure 1(right) shows the sequence of these operations in detail.

The ERM estimator. In the classification task, we employ empirical risk minimization over ConvNets as outlined in the following equation:

$$\widehat{\boldsymbol{W}} = \arg\min_{\boldsymbol{W}\in\mathcal{W}_{d,\underline{m},L,S,D,B}} \Big\{\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) = \frac{1}{n}\sum_{i=1}^{n}\mathrm{loss}(y_i, \mu_{\mathrm{NN}}^{\boldsymbol{W}}(\cdot|\boldsymbol{x}_i))\Big\}, \tag{7}$$

where, for simplicity of analysis, we opt for the square loss rather than the more commonly used cross-entropy loss:

$$\mathrm{loss}(y, \mu_{\mathrm{NN}}^{\boldsymbol{W}}(\cdot|\boldsymbol{x})) = \textstyle\sum_{s=1}^{S}\Big(1\{y=s\} - \mu_{\mathrm{NN}}^{\boldsymbol{W}}(s|\boldsymbol{x})\Big)^2. \tag{8}$$

The parameter space for the ConvNets is defined as:

$$\mathcal{W}_{d,\underline{m},L,S,D,B} := \Big\{\boldsymbol{W} \text{ as defined in (6)} : |\!|\!|\boldsymbol{W}|\!|\!| := \max_{j\in[3]}\max_{\ell\in[L]}\max_{\iota\in[m^{(\ell)}]} \|W_{j,\iota}^{(\ell)}\|_{\mathrm{op}} \le B\Big\}. \tag{9}$$

We anticipate that the empirical risk minimizer, $\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}$, could learn the Bayes classifier $\mu_\star$, as the global minimizer of the population risk over all conditional distributions yields the Bayes classifier:

$$\mu_\star(\cdot|\cdot) = \arg\min_{\mu}\Big\{\mathsf{R}(\mu) = \mathbb{E}[\mathrm{loss}(y, \mu(\cdot|\boldsymbol{x}))]\Big\}.$$

In our theoretical analysis, we measure the discrepancy between $\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}$ and $\mu_\star$ using the squared Euclidean distance:

$$\mathsf{D}_2^2(\mu, \mu_\star) = \mathbb{E}_{\boldsymbol{x}\sim\mu_\star}\Big[\textstyle\sum_{s=1}^{S}\Big(\mu(s|\boldsymbol{x}) - \mu_\star(s|\boldsymbol{x})\Big)^2\Big]. \tag{10}$$

Sample complexity bound. The subsequent theorem establishes the bound of the D_2^2-distance between the ConvNet estimator $\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}$ and the true Bayes classifier $\mu_\star$.

Theorem 1 (Learning to classify using ConvNets)**.** *Let Assumption 1 and 2 hold. Let $\mathcal{W}_{d,\underline{m},L,S,D,B}$ be the set defined as in Eq. (9), where $D \ge S^2K^2d\cdot 3^L$ and $B = \mathrm{Poly}(d, S, K, 3^L, D)$. Let $\widehat{\boldsymbol{W}}$ be the empirical risk minimizer as in Eq. (7). Then with probability at least $1-\eta$, we have*

$$\mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \mu_\star) \le C\cdot\Big(\frac{S^4K^4d^2\cdot 3^{2L}}{D^2} + \sqrt{\frac{LD(D+2S+1)\|\underline{m}\|_1\log(d\|\underline{m}\|_1 DS\cdot 3^L) + \log(1/\eta)}{n}}\Big). \tag{11}$$

The proof of Theorem 1 is detailed in Section E.

Remark 2. *To ensure the D_2^2-distance is less than ϵ^2, Theorem 1 requires to take*

$$D = \Theta(S^2K^2d3^L/\epsilon), \quad n = \tilde{\Theta}(LS^4K^4d^2 3^{2L}\|\underline{m}\|_1/\epsilon^6), \tag{12}$$

where $\tilde{\Theta}$ hides a logarithmic factor $\log(d\|\underline{m}\|_1 S\cdot 3^L/(\eta\epsilon))$. The dependency on any of the parameters $(S, d, 3^L, K, \|\underline{m}\|_1, \epsilon)$ could potentially be refined by imposing additional assumptions on the ψ functions or through a more detailed analysis of approximation and generalization. This question of improving rates remains open for future work.

Consider a simplified scenario where $S = 2$, K is constant, and $m^{(\ell)} = m \ge 3$ for each $\ell \in [L]$, leading to $d = m^L$. In this setup, the sample complexity gives

$$n = \tilde{\Theta}(L^2 m\cdot d^{2+2\log_m 3}/\epsilon^6) \le \tilde{\Theta}(L^2 m\cdot d^4/\epsilon^6), \tag{13}$$

exhibiting a polynomial dependence on d and $1/\epsilon$. Such polynomial scaling aligns with existing literature (Poggio et al., 2017; Malach & Shalev-Shwartz, 2020; Schmidt-Hieber, 2020; Allen-Zhu & Li, 2022; Petrini et al., 2023), which indicates that learning hierarchical models using multi-layer networks avoids the curse of dimensionality. Theorem 1 serves as a warm-up result in the classification context. In Section 4, we aim to extend similar methodologies to address denoising problems, employing analogous proof strategies.

Proof strategy. The proof strategy begins by decomposing the squared distance between the learned model and the true Bayes classifier into approximation and generalization error terms. The generalization error is bounded using a standard chaining argument, leading to a rate of $\tilde{O}(\sqrt{d_{\rm p}/n})$, where $d_{\rm p}$ denotes the number of ConvNet parameters. The focus then shifts to controlling the approximation error. This is done by first introducing the belief propagation and message passing algorithm for computing the Bayes classifier, and subsequently showing that ConvNets can approximate this algorithm effectively. A detailed outline of the proof strategy is provided in Appendix A.1, with the complete proof presented in Appendix E.

4 Denoising and diffusion in GHMs

In this section, we consider the denoising task within the GHM. Consider the joint distribution of noisy and clean covariates $(\boldsymbol{z}, \boldsymbol{x})$, generated from the following: $\boldsymbol{x} \sim \mu_\star$ represents the clean covariates, and $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{g}$ where $\boldsymbol{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ denotes the independent isotropic Gaussian noise. For simplicity in notation and with a slight abuse of notations, we continue to refer to the joint distribution of $(\boldsymbol{z}, \boldsymbol{x})$ as $\mu_\star$.

We consider a scenario where a set of iid samples $\{(\boldsymbol{z}_i, \boldsymbol{x}_i)\}_{i\in[n]} \sim_{iid} \mu_\star$ is drawn from the distribution. Our objective is to learn a denoiser $\boldsymbol{m}(\boldsymbol{z})$ from this dataset. With a suitable loss function, the optimal denoiser is the Bayes denoiser $\boldsymbol{m}_\star(\boldsymbol{z}) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{z})\sim\mu_\star}[\boldsymbol{x}|\boldsymbol{z}]$, which calculates the posterior expectation of $\boldsymbol{x}$ given $\boldsymbol{z}$. We aim to examine the sample complexity of learning this denoiser through empirical risk minimization over the class of U-Nets. The approaches and results of this section closely align with those discussed in Section 3 on the classification task.

The U-Net architecture. We here introduce the U-Net architecture used for denoising, represented as $\boldsymbol{m}_{\rm NN}(\boldsymbol{z}) \in \mathbb{R}^d$ for input $\boldsymbol{z} \in \mathbb{R}^d$. Initially, we set $\mathsf{h}^{(L)}_{\downarrow,v} = (-(x - z_v)^2/2)_{x\in[S]} \in \mathbb{R}^S$ for $v \in \mathcal{V}^{(L)}$ for each node $v \in \mathcal{V}^{(L)}$. The operational flow of the network unfolds as follows:

$$
\begin{aligned}
\mathsf{q}^{(\ell)}_{\downarrow,v} &= \mathrm{NN}^{(\ell)}_{\downarrow,\iota(v)}(\mathrm{normalize}(\mathsf{h}^{(\ell)}_{\downarrow,v})) \in \mathbb{R}^S, && \ell \in [L], \quad v \in \mathcal{V}^{(\ell)},\\
\mathsf{h}^{(\ell-1)}_{\downarrow,v} &= \textstyle\sum_{v'\in\mathcal{C}(v)} \mathsf{q}^{(\ell)}_{\downarrow,v'} \in \mathbb{R}^S, && \ell \in [L], \quad v \in \mathcal{V}^{(\ell-1)},\\
\mathsf{u}^{(\ell)}_{\uparrow,v} &= \mathsf{b}^{(\ell-1)}_{\uparrow,\mathrm{pa}(v)} \in \mathbb{R}^S, \quad (\text{with } \mathsf{b}^{(0)}_{\uparrow,\mathrm{r}} = \mathsf{h}^{(0)}_{\downarrow,\mathrm{r}}) && \ell \in [L], \quad v \in \mathcal{V}^{(\ell)}, \qquad \text{(UNet)}\\
\mathsf{b}^{(\ell)}_{\uparrow,v} &= \mathrm{NN}^{(\ell)}_{\uparrow,\iota(v)}(\mathrm{normalize}(\mathsf{u}^{(\ell)}_{\uparrow,v} - \mathsf{q}^{(\ell)}_{\downarrow,v})) + \mathsf{h}^{(\ell)}_{\downarrow,v} \in \mathbb{R}^S, && \ell \in [L], \quad v \in \mathcal{V}^{(\ell)},\\
\boldsymbol{m}_{\rm NN}(\boldsymbol{z})_v &= \textstyle\sum_{s\in[S]} s \cdot \mathrm{softmax}(\mathsf{b}^{(L)}_{\uparrow,v})_s, && v \in \mathcal{V}^{(L)}.
\end{aligned}
$$

The functions $\{\mathrm{NN}^{(\ell)}_{\downarrow,\iota}, \mathrm{NN}^{(\ell)}_{\uparrow,\iota} : \mathbb{R}^S \to \mathbb{R}^S\}_{\ell\in[L]}$ and $\mathrm{normalize} : \mathbb{R}^S \to \mathbb{R}^S$ are defined as follows:

$$
\mathrm{NN}^{(\ell)}_{\diamond,\iota}(\mathsf{h})_s = W^{(\ell)}_{1,\diamond,\iota} \cdot \mathrm{ReLU}(W^{(\ell)}_{2,\diamond,\iota} \cdot \mathrm{ReLU}(W^{(\ell)}_{3,\diamond,\iota} \cdot [\mathsf{h}; 1])), \quad s \in [S], \diamond \in \{\downarrow,\uparrow\}, \ell \in [L], \iota \in [m^{(\ell)}], \tag{14}
$$

$$
\mathrm{normalize}(h)_s = h_s - \max_{s'\in[S]} h_{s'}. \tag{15}
$$

The dimensions of the weight matrices within the U-Net are specified as follows:

$$
\boldsymbol{W} = \left\{ \{W^{(\ell)}_{1,\diamond,\iota} \in \mathbb{R}^{S\times D}\}_{\iota\in[m^{(\ell)}]}, \{W^{(\ell)}_{2,\diamond,\iota} \in \mathbb{R}^{D\times D}\}_{\iota\in[m^{(\ell)}]}, \{W^{(\ell)}_{3,\diamond,\iota} \in \mathbb{R}^{D\times(S+1)}\}_{\iota\in[m^{(\ell)}]} \right\}_{\ell\in[L],\diamond\in\{\downarrow,\uparrow\}} \tag{16}
$$

Furthermore, we denote $\boldsymbol{m}^{\boldsymbol{W}}_{\rm NN}$ as the U-Net denoiser $\boldsymbol{m}_{\rm NN}$ parameterized by $\boldsymbol{W}$. A schematic illustration of a U-Net with 3 layers is provided in Figure 2.

Remark 3 (An explanation of the "U-Net" architecture)**.** *As noted in our discussion of the convolutional network as in Remark 1, the "convolutional layers" and the "normalization operator" described in (UNet) are different from practical implementations. However, we continue to use these terms because they retain core characteristics of their practical counterparts.*

An important feature of (UNet) is its encoder-decoder architecture and the inclusion of long skip connections, which closely mirror practical implementations. Specifically, the encoder sequence in (UNet) progresses as $\mathsf{h}^{(L)}_{\downarrow} \to \mathsf{q}^{(L)}_{\downarrow} \to \mathsf{h}^{(L-1)}_{\downarrow} \to \cdots \to \mathsf{q}^{(1)}_{\downarrow} \to \mathsf{h}^{(0)}_{\downarrow}$*, consisting of a series*

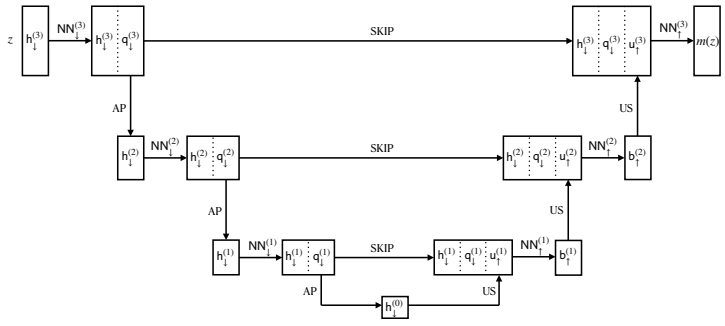

Figure 2: A U-Net with $L = 3$. "AP" stands for average-pooling. "US" stands for up-sampling. "SKIP" stands for long skip connections.

of convolutional, average pooling, and normalization layers. This part of the architecture is the same as (ConvNet) for the classification task. The decoder sequence ascends as $\mathsf{b}_\uparrow^{(0)} \to \mathsf{u}_\uparrow^{(1)} \to \mathsf{b}_\uparrow^{(1)} \to \cdots \to \mathsf{b}_\uparrow^{(L-1)} \to \mathsf{u}_\uparrow^{(L)} \to \boldsymbol{m}_{\mathrm{NN}}$*, consisting of a series of convolutional, up-sampling, and normalization layers. Moreover, the computation of* $\mathsf{b}_\uparrow^{(\ell)}$ *utilizes* $\mathsf{u}_\uparrow^{(\ell)}$ *from the upward sequence, and* $(\mathsf{h}_\downarrow^{(\ell)}, \mathsf{q}_\downarrow^{(\ell)})$ *from the downward process, which is enabled by the long skip connections. This encoder-decoder architecture and the long skip connections are effectively visualized in Figure 2.*

The ERM estimator. In the denoising task, we employ empirical risk minimization over U-Nets as outlined in the following equation:

$$\widehat{\boldsymbol{W}} = \arg\min_{\boldsymbol{W} \in \mathcal{W}_{d,\underline{m},L,S,D,B}} \Big\{ \widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) = \frac{1}{nd} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}(\boldsymbol{z}_i)\|_2^2 \Big\}. \tag{17}$$

The parameter space for the U-Nets is defined as:

$$\mathcal{W}_{d,\underline{m},L,S,D,B} := \Big\{ \boldsymbol{W} \text{ as defined in (16)} : |\!|\!|\boldsymbol{W}|\!|\!| := \max_{\diamond \in \{\downarrow,\uparrow\}} \max_{j \in [3]} \max_{\ell \in [L]} \max_{\iota \in [m^{(\ell)}]} \|W_{j,\iota}^{(\ell)}\|_{\mathrm{op}} \le B \Big\}. \tag{18}$$

We anticipate that the empirical risk minimizer, $\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}$, could learn the Bayes denoiser $\boldsymbol{m}_\star$, as the global minimizer of the population risk over all functions yields the Bayes denoiser:

$$\boldsymbol{m}_\star(\cdot) = \arg\min_{\boldsymbol{m}} \Big\{ \mathsf{R}(\boldsymbol{m}) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{z}) \sim \mu_\star} [d^{-1} \|\boldsymbol{x} - \boldsymbol{m}(\boldsymbol{z})\|_2^2] \Big\}.$$

In our theoretical analysis, we measure the discrepancy between $\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}$ and $\boldsymbol{m}_\star$ using the squared Euclidean distance:

$$\mathsf{D}_2^2(\boldsymbol{m}, \boldsymbol{m}_\star) = \mathbb{E}_{\boldsymbol{z} \sim \mu_\star} \Big[d^{-1} \|\boldsymbol{m} - \boldsymbol{m}_\star(\boldsymbol{z})\|_2^2 \Big]. \tag{19}$$

Sample complexity bound. The subsequent theorem establishes the bound of the D_2^2-distance between the U-Net estimator $\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}$ and the true Bayes denoiser $\boldsymbol{m}_\star$.

Theorem 2 (Learning to denoise using U-Nets)**.** *Let Assumption 1 and 2 hold. Let* $\mathcal{W}_{d,\underline{m},L,S,D,B}$ *be the set defined as in Eq. (18), where* $B = \mathrm{Poly}(d, S, K, 18^L, D)$*. Let* $\widehat{\boldsymbol{W}}$ *be the empirical risk minimizer as in Eq. (17). Then with probability at least* $1 - \eta$*, we have*

$$\mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \boldsymbol{m}_\star) \le C \cdot \Big(\frac{S^6 K^4 d^2 \cdot 18^{2L}}{D^2} + S^2 \cdot \sqrt{\frac{LD(D+2S+1)\|\underline{m}\|_1 \log(d\|\underline{m}\|_1 DS \cdot 18^L) + \log(1/\eta)}{n}} \Big).$$

The proof of Theorem 2 is detailed in Section H.

Remark 4. *To ensure the D_2^2-distance is less than ϵ^2, Theorem 2 requires to take*

$$D = \Theta(S^3K^2d18^L/\epsilon), \quad n = \tilde{\Theta}(LS^{10}K^4d^2 18^{2L}\|\underline{m}\|_1/\epsilon^6), \tag{20}$$

where $\tilde{\Theta}$ hides a logarithmic factor $\log(d\|\underline{m}\|_1 S \cdot 18^L/(\eta\epsilon))$. Similar to Theorem 1, the dependency on any of the parameters $(S, d, 18^L, K, \|\underline{m}\|_1, \epsilon)$ could potentially be refined by imposing additional assumptions on the ψ functions or through a more detailed analysis of approximation and generalization. This question of improving rates remains open for future work.

Consider a simplified scenario where $S = 2$, K is constant, and $m^{(\ell)} = m \geq 3$ for each $\ell \in [L]$, leading to $d = m^L$. In this setup, the sample complexity gives

$$n = \tilde{\Theta}(L^2 m \cdot d^{2+2\log_m 18}/\epsilon^6) \leq \tilde{\Theta}(L^2 m \cdot d^8/\epsilon^6), \tag{21}$$

exhibiting a polynomial dependence on d and $1/\epsilon$. Although the degree of the polynomial is substantial and potentially improvable, this gives the first polynomial sample complexity result for learning the Bayes denoiser in a hierarchical model using the U-Nets.

Connection to the task of diffusion generative modeling. The denoising task examined in this section is closely related to the diffusion model approach (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019; Song et al., 2020) to generative modeling. Diffusion models involve learning a generative model from a dataset of n independent and identically distributed samples $\{\boldsymbol{x}_i\}_{i=1}^n$, drawn from an unknown data distribution $\mu_\star \in \mathcal{P}([S]^d)$. The goal is to generate new samples $\hat{\boldsymbol{x}} \sim \hat{\mu}$ that match the distribution $\mu_\star$. Most diffusion model formulations involve a series of steps that are closely related to the denoising task described earlier. Here, we illustrate how they work using a variant of diffusion models, the stochastic localization process (Eldan, 2013; El Alaoui et al., 2022; Montanari & Wu, 2023; Celentano, 2022; Montanari, 2023):

- **Step 1.** Fit approximate denoising functions $\hat{\boldsymbol{m}}_t : \mathbb{R}^d \to \mathbb{R}^d$ for $t \in [0, T]$. This is done by minimizing the empirical risk over a class of neural networks $\mathcal{F}$ (i.e., the denoising task discussed in this section):

$$\hat{\boldsymbol{m}}_t \equiv \arg\min_{\mathrm{NN}_t \in \mathcal{F}} \frac{1}{nd} \sum_{i=1}^n \|\boldsymbol{x}_i - \mathrm{NN}_t(t \cdot \boldsymbol{x}_i + \sqrt{t} \cdot \boldsymbol{g}_i)\|_2^2, \quad \boldsymbol{g}_i \sim_{iid} \mathcal{N}(\boldsymbol{0}, \mathbf{I}_d). \tag{ERM}$$

- **Step 2.** Simulate a discretized version of a stochastic differential equation (SDE) starting from zero, whose drift term gives the approximate denoising function:

$$\mathrm{d}\boldsymbol{z}_t = \hat{\boldsymbol{m}}_t(\boldsymbol{z}_t) \cdot \mathrm{d}t + \mathrm{d}\boldsymbol{B}_t, \ \ t \in [0, T], \quad \boldsymbol{z}_0 = \boldsymbol{0}, \tag{SDE}$$

  and generate an approximate sample $\hat{\boldsymbol{x}} = \boldsymbol{z}_T/T \in \mathbb{R}^d$ at the final time T.

Standard analysis shows that by replacing the fitted denoising functions $\hat{\boldsymbol{m}}_t(\boldsymbol{z})$ with the true denoising functions $\boldsymbol{m}_t(\boldsymbol{z})$ in Eq. (SDE) and allowing $T \to \infty$, we can effectively recover the original data distribution $\mu_\star$. Consequently, the quality of samples generated from diffusion models hinges on two critical factors: (1) How well the fitted denoising functions $\hat{\boldsymbol{m}}_t$ (ERM) approximate the true denoising functions $\boldsymbol{m}_t$; (2) How accurately the SDE discretization scheme approximates the continuous process as in (SDE).

Recent work has made substantial progress in addressing these two theoretical questions: controlling the SDE discretization error, assuming a reliable denoising function estimator is available (Chen et al., 2022a; 2023a; Lee et al., 2023; Li et al., 2023; Benton et al., 2023); and controlling the denoising function approximation error through neural networks (Oko et al., 2023; Chen et al., 2023b; Mei & Wu, 2023). However, these works have not explained the benefit of employing U-Net in image diffusion modeling, which is the primary focus of the current work. Indeed, by integrating the sample complexity bounds for learning denoising functions, as established in Theorem 2, with standard SDE discretization error bounds, such as the result established in Benton et al. (2023), it is straightforward to derive an end-to-end error bound for the sampling process of diffusion models in GHMs, similar to the strategy of Oko et al. (2023); Chen et al. (2023b); Mei & Wu (2023)[2].

[2]We note that the stochastic localization formulation is equivalent to the DDPM diffusion model, differing only in parametrization (Montanari, 2023). In the DDPM model, U-Nets serve to approximate the score function. The score function is a linear combination of the denoising function with an identity map, as per Tweedie's formula. Consequently, Theorem 2 can be readily adapted to establish a sample complexity bound for learning this score function.

Proof strategy. The proof strategy for the denoising task parallels that of the classification task. The squared distance between the learned model and the true Bayes denoiser is decomposed into approximation and generalization error terms. The generalization error is bounded via a standard parameter counting argument. The approximation error is controlled by first introducing the belief propagation and message passing algorithm for computing the Bayes denoiser and then showing that U-Nets can effectively approximate this algorithm. A detailed outline of the proof strategy is provided in Appendix A.2, with the complete proof presented in Appendix H.

5 Further related work

Generative hierarchical models. Hierarchical modeling of data distributions has been proposed in a series of works (Mossel, 2016; Poggio et al., 2017; Malach & Shalev-Shwartz, 2020; Schmidt-Hieber, 2020; Allen-Zhu & Li, 2022; Petrini et al., 2023; Sclocchi et al., 2024; Tomasini & Wyart, 2024; Cagnetta & Wyart, 2024; Garnier-Brun et al., 2024; Kadkhodaie et al., 2023a;b). While the hierarchical models in Poggio et al. (2017); Malach & Shalev-Shwartz (2020); Schmidt-Hieber (2020); Allen-Zhu & Li (2022) remain deterministic, Mossel (2016); Petrini et al. (2023); Sclocchi et al. (2024); Tomasini & Wyart (2024); Cagnetta & Wyart (2024); Garnier-Brun et al. (2024) studied the generative version of hierarchical models. The diffusion model for multi-scale image distribution representations has been empirically examined in Kadkhodaie et al. (2023a;b), which demonstrated that U-Nets are effective in modeling denoising algorithms. The theoretical and empirical evidence presented in Sclocchi et al. (2024); Tomasini & Wyart (2024); Petrini et al. (2023) underscores the effectiveness of generative hierarchical models in capturing the combinatorial properties of image datasets. Given their significant relevance to this work, we delve deeper into these studies.

Contributions of Petrini et al. (2023); Sclocchi et al. (2024); Tomasini & Wyart (2024). The series of works on hierarchical generative models (Petrini et al., 2023; Sclocchi et al., 2024; Tomasini & Wyart, 2024) inspired the current study. Sclocchi et al. (2024) first pointed out that the belief propagation denoising algorithm of hierarchical models consists of downward and upward processes. Through mean-field analysis on a random generative hierarchical model, they identified a phase transition phenomenon, aligning with empirical observations in diffusion models, thereby providing strong evidence of the efficacy of these models in handling combinatorial data properties. Petrini et al. (2023), on the other hand, first introduced these models in a classification context. Petrini et al. (2023); Tomasini & Wyart (2024) demonstrated that learning hierarchical models using multi-layer networks circumvents the curse of dimensionality. Specifically, they theoretically and empirically characterized the sample complexity, showing that it remains polynomial in dimension when learning convolutional networks under random generative rules. On the other hand, in the absence of correlations, they showed that the sample complexity is again exponential in the dimension, even for hierarchical generative models. This learning incapability is not captured by our analysis which does not consider optimization.

ConvNets and U-Nets and their implicit bias. Convolutional networks (LeCun et al., 1989; 1998; Krizhevsky et al., 2012; Szegedy et al., 2015; He et al., 2016) have become the state-of-the-art architecture for image classification and have been the backbone for many computer vision tasks. U-Nets (Ronneberger et al., 2015; Zhou et al., 2018; Siddique et al., 2021; Oktay et al., 2018) have been particularly well-suited for image segmentation and denoising tasks (Ronneberger et al., 2015), and have served as the backbone architecture for diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019; Song et al., 2020). A series of theoretical works has explained the inductive bias of CNNs (Bruna & Mallat, 2013; Gunasekar et al., 2018; Bietti & Mairal, 2019b;a; Scetbon & Harchaoui, 2020; Li et al., 2020; Bietti, 2021; Mei et al., 2021; Misiakiewicz & Mei, 2022; Cagnetta et al., 2023; Favero et al., 2021; Bietti et al., 2021; Xiao, 2022; Petrini et al., 2023; Tomasini & Wyart, 2024; Wang & Wu, 2024). However, they mostly focused on the classification and regression setting and were not concerned with the role of U-Nets in denoising tasks. The implicit bias of U-Nets has been theoretically investigated in Williams et al. (2024); Falck et al. (2022), where they found that the U-Nets are conjugate to the ResNets. In contrast, we demonstrate that U-Nets can effectively approximate the belief propagation denoising algorithms of GHMs. We note that Cui et al. (2023) analyzed the learning dynamics for a simple U-Net in diffusion models.

Neural networks approximation of algorithms. A recent line of work has investigated the expressiveness of neural networks through an algorithm approximation viewpoint (Wei et al., 2022; Bai et al., 2024; Giannou et al., 2023; Liu et al., 2022a; Marwah et al., 2021; 2023; Lin et al., 2023; Mei & Wu, 2023). In particular, Wei et al. (2022); Bai et al. (2024); Giannou et al. (2023); Liu et al. (2022a); Lin et al. (2023) demonstrate that transformers can efficiently approximate several algorithm classes, such as gradient descent, reinforcement learning algorithms, and even Turing machines. In the context of diffusion models, Mei & Wu (2023) shows that ResNets can efficiently approximate the score function of high-dimensional graphical models by approximating the variational inference algorithm. Our work is closely related to Mei & Wu (2023), except that we study neural network approximation in a different statistical model and network architecture.

From a practical viewpoint, a line of work has focused on neural network denoising by unrolling iterative denoising algorithms into deep networks (Gregor & LeCun, 2010; Zheng et al., 2015; Zhang & Ghanem, 2018; Papyan et al., 2017; Ma et al., 2021; Chen et al., 2018; Borgerding et al., 2017; Monga et al., 2021; Yu et al., 2023a;b). While this literature has primarily focused on devising better denoising algorithms, our work leverages this perspective to develop neural network approximation theory and explain existing network architectures.

Related theory of diffusion models. In recent years, diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019; Song et al., 2020) have emerged as a leading approach for generative modeling. Neural network-based score function approximation has been recently studied from the function approximation viewpoint in Oko et al. (2023); Chen et al. (2023b); Yuan et al. (2023); Shah et al. (2023); Biroli & Mézard (2023), and from the algorithm approximation viewpoint in Mei & Wu (2023). Theoretical studies of other aspects of diffusion models include Liu et al. (2022b); Li et al. (2023); Lee et al. (2023); Chen et al. (2022b; 2023d; 2022a; 2023c;a); Benton et al. (2023); El Alaoui et al. (2022); Montanari & Wu (2023); Celentano (2022); Ghio et al. (2023); Biroli & Mézard (2023); Biroli et al. (2024); Cui et al. (2023); Fu et al. (2024); Wu et al. (2024). For a comprehensive introduction to the theory of diffusion models, see the recent review (Chen et al., 2024).

6 Conclusions and Discussions

In this paper, we introduced a novel interpretation of the U-Net architecture through the lens of generative hierarchical models. We demonstrated that their belief propagation denoising algorithm naturally induces the encoder-decoder structure, the long skip connections, and the pooling and upsampling operations of the U-Nets. We also provided an efficient sample complexity bound for learning the denoising function with U-Nets. Furthermore, we discussed the broader implications of these findings for diffusion models. We also showed that ConvNets are well-suited for classification tasks within these models. Our study offers a unified perspective on the roles of ConvNets and U-Nets, highlighting the versatility of generative hierarchical models in capturing complex data distributions across language and image domains.

The results presented in this paper offer considerable scope for enhancement. We initially assumed that the covariates $\boldsymbol{x}$ lie in the discrete space $[S]^d$, and extending these results to continuous spaces would be an intriguing direction for future research. Additionally, the dependencies of the sample complexity bound on d and $1/\epsilon$ may be amenable to improvement through more careful analysis. Moreover, the convolution operations employed in this paper are different from those commonly employed in practical settings. It would be worthwhile to explore graphical models where the belief propagation algorithm aligns more naturally with ConvNets and U-Nets that utilize standard convolution operations.

On the practical side, our theoretical findings generated a hypothesis of the functionality of each layer of the U-Nets. Verifying these hypotheses in pre-trained U-Nets, such as those used in stable diffusion models, using interpretability methods, could yield valuable insights. Furthermore, extending these results to include conditional denoising functions represents an exciting direction for future research. Finally, we hope that the insights provided in this paper could guide the design of innovative network architectures.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. How can deep learning performs deep (hierarchical) learning. 2022.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 1, context-free grammar. *arXiv preprint arXiv:2305.13673*, 2023.

Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. *Advances in neural information processing systems*, 36, 2024.

Joe Benton, Valentin De Bortoli, Arnaud Doucet, and George Deligiannidis. Linear convergence bounds for diffusion models via stochastic localization. *arXiv preprint arXiv:2308.03686*, 2023.

Alberto Bietti. Approximation and learning with deep convolutional models: a kernel perspective. *arXiv preprint arXiv:2102.10032*, 2021.

Alberto Bietti and Julien Mairal. Group invariance, stability to deformations, and complexity of deep convolutional representations. *Journal of Machine Learning Research*, 20(25):1–49, 2019a.

Alberto Bietti and Julien Mairal. On the inductive bias of neural tangent kernels. *Advances in Neural Information Processing Systems*, 32, 2019b.

Alberto Bietti, Luca Venturi, and Joan Bruna. On the sample complexity of learning under geometric stability. *Advances in neural information processing systems*, 34:18673–18684, 2021.

Giulio Biroli and Marc Mézard. Generative diffusion in very large dimensions. *arXiv preprint arXiv:2306.03518*, 2023.

Giulio Biroli, Tony Bonnaire, Valentin de Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *arXiv preprint arXiv:2402.18491*, 2024.

Mark Borgerding, Philip Schniter, and Sundeep Rangan. Amp-inspired deep networks for sparse linear inverse problems. *IEEE Transactions on Signal Processing*, 65(16):4293–4308, 2017.

Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

Francesco Cagnetta and Matthieu Wyart. Towards a theory of how the structure of language is acquired by deep neural networks. *arXiv preprint arXiv:2406.00048*, 2024.

Francesco Cagnetta, Alessandro Favero, and Matthieu Wyart. What can be learnt with wide convolutional neural networks? In *International Conference on Machine Learning*, pp. 3347–3379. PMLR, 2023.

Michael Celentano. Sudakov-fernique post-amp, and a new proof of the local convexity of the tap free energy. *arXiv preprint arXiv:2208.09550*, 2022.

Hongrui Chen, Holden Lee, and Jianfeng Lu. Improved analysis of score-based generative modeling: User-friendly bounds under minimal smoothness assumptions. In *International Conference on Machine Learning*, pp. 4735–4763. PMLR, 2023a.

Minshuo Chen, Kaixuan Huang, Tuo Zhao, and Mengdi Wang. Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data. *arXiv preprint arXiv:2302.07194*, 2023b.

Minshuo Chen, Song Mei, Jianqing Fan, and Mengdi Wang. An overview of diffusion models: Applications, guided generation, statistical rates and optimization. *arXiv preprint arXiv:2404.07771*, 2024.

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. *arXiv preprint arXiv:2209.11215*, 2022a.

Sitan Chen, Sinho Chewi, Holden Lee, Yuanzhi Li, Jianfeng Lu, and Adil Salim. The probability flow ode is provably fast. *arXiv preprint arXiv:2305.11798*, 2023c.

Sitan Chen, Giannis Daras, and Alex Dimakis. Restoration-degradation beyond linear diffusions: A non-asymptotic analysis for ddim-type samplers. In *International Conference on Machine Learning*, pp. 4462–4484. PMLR, 2023d.

Xiaohan Chen, Jialin Liu, Zhangyang Wang, and Wotao Yin. Theoretical linear convergence of unfolded ista and its practical weights and thresholds. *Advances in Neural Information Processing Systems*, 31, 2018.

Yongxin Chen, Sinho Chewi, Adil Salim, and Andre Wibisono. Improved analysis for a proximal algorithm for sampling. In *Conference on Learning Theory*, pp. 2984–3014. PMLR, 2022b.

Noam Chomsky. On certain formal properties of grammars. *Information and control*, 2(2):137–167, 1959.

Hugo Cui, Florent Krzakala, Eric Vanden-Eijnden, and Lenka Zdeborová. Analysis of learning a flow-based generative model from limited sample complexity. *arXiv preprint arXiv:2310.03575*, 2023.

Michal Drozdzal, Eugene Vorontsov, Gabriel Chartrand, Samuel Kadoury, and Chris Pal. The importance of skip connections in biomedical image segmentation. In *International workshop on deep learning in medical image analysis, international workshop on large-scale annotation of biomedical data and expert label synthesis*, pp. 179–187. Springer, 2016.

Ahmed El Alaoui, Andrea Montanari, and Mark Sellke. Sampling from the sherrington-kirkpatrick gibbs measure via algorithmic stochastic localization. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 323–334. IEEE, 2022.

Ronen Eldan. Thin shell implies spectral gap up to polylog via a stochastic localization scheme. *Geometric and Functional Analysis*, 23(2):532–569, 2013.

Fabian Falck, Christopher Williams, Dominic Danks, George Deligiannidis, Christopher Yau, Chris C Holmes, Arnaud Doucet, and Matthew Willetts. A multi-resolution framework for u-nets with applications to hierarchical vaes. *Advances in Neural Information Processing Systems*, 35:15529–15544, 2022.

Alessandro Favero, Francesco Cagnetta, and Matthieu Wyart. Locality defeats the curse of dimensionality in convolutional teacher-student scenarios. *Advances in Neural Information Processing Systems*, 34:9456–9467, 2021.

Hengyu Fu, Zhuoran Yang, Mengdi Wang, and Minshuo Chen. Unveil conditional diffusion models with classifier-free guidance: A sharp statistical theory. *arXiv preprint arXiv:2403.11968*, 2024.

Jérôme Garnier-Brun, Marc Mézard, Emanuele Moscato, and Luca Saglietti. How transformers learn structured data: insights from hierarchical filtering. *arXiv preprint arXiv:2408.15138*, 2024.

Davide Ghio, Yatin Dandi, Florent Krzakala, and Lenka Zdeborová. Sampling with flows, diffusion and autoregressive neural networks: A spin-glass perspective. *arXiv preprint arXiv:2308.14085*, 2023.

Angeliki Giannou, Shashank Rajput, Jy-yong Sohn, Kangwook Lee, Jason D Lee, and Dimitris Papailiopoulos. Looped transformers as programmable computers. *arXiv preprint arXiv:2301.13196*, 2023.

Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th international conference on international conference on machine learning*, pp. 399–406, 2010.

Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. *Advances in neural information processing systems*, 31, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Ya Jin and Stuart Geman. Context and hierarchy in a probabilistic image model. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 2, pp. 2145–2152. IEEE, 2006.

Zahra Kadkhodaie, Florentin Guth, Stéphane Mallat, and Eero P Simoncelli. Learning multi-scale local conditional probability models of images. *arXiv preprint arXiv:2303.02984*, 2023a.

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

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2, 1989.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence of score-based generative modeling for general data distributions. In *International Conference on Algorithmic Learning Theory*, pp. 946–985. PMLR, 2023.

Lillian Lee. Learning of context-free languages: A survey of the literature. 1996.

Gen Li, Yuting Wei, Yuxin Chen, and Yuejie Chi. Towards faster non-asymptotic convergence for diffusion-based generative models. *arXiv preprint arXiv:2306.09251*, 2023.

Jia Li, Robert M Gray, and Richard A Olshen. Multiresolution image classification by hierarchical modeling with two-dimensional hidden markov models. *IEEE transactions on information theory*, 46(5):1826–1841, 2000.

Zhiyuan Li, Yi Zhang, and Sanjeev Arora. Why are convolutional nets more sample-efficient than fully-connected nets? *arXiv preprint arXiv:2010.08515*, 2020.

Licong Lin, Yu Bai, and Song Mei. Transformers as decision makers: Provable in-context reinforcement learning via supervised pretraining. *arXiv preprint arXiv:2310.08566*, 2023.

Bingbin Liu, Jordan T Ash, Surbhi Goel, Akshay Krishnamurthy, and Cyril Zhang. Transformers learn shortcuts to automata. *arXiv preprint arXiv:2210.10749*, 2022a.

Xingchao Liu, Lemeng Wu, Mao Ye, and Qiang Liu. Let us build bridges: Understanding and extending diffusion generative models. *arXiv preprint arXiv:2208.14699*, 2022b.

Yao Ma, Xiaorui Liu, Tong Zhao, Yozen Liu, Jiliang Tang, and Neil Shah. A unified view on graph neural networks as graph signal denoising. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 1202–1211, 2021.

Eran Malach and Shai Shalev-Shwartz. The implications of local correlation on learning some deep functions. *Advances in Neural Information Processing Systems*, 33:1322–1332, 2020.

Tanya Marwah, Zachary Lipton, and Andrej Risteski. Parametric complexity bounds for approximating pdes with neural networks. *Advances in Neural Information Processing Systems*, 34: 15044–15055, 2021.

Tanya Marwah, Zachary Chase Lipton, Jianfeng Lu, and Andrej Risteski. Neural network approximations of pdes beyond linearity: A representational perspective. In *International Conference on Machine Learning*, pp. 24139–24172. PMLR, 2023.

Song Mei and Yuchen Wu. Deep networks as denoising algorithms: Sample-efficient learning of diffusion models in high-dimensional graphical models. *arXiv preprint arXiv:2309.11420*, 2023.

Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Learning with invariances in random features and kernel models. In *Conference on Learning Theory*, pp. 3351–3418. PMLR, 2021.

Marc Mezard and Andrea Montanari. *Information, physics, and computation*. Oxford University Press, 2009.

Theodor Misiakiewicz and Song Mei. Learning with convolution and pooling operations in kernel methods. *Advances in Neural Information Processing Systems*, 35:29014–29025, 2022.

Vishal Monga, Yuelong Li, and Yonina C Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44, 2021.

Andrea Montanari. Sampling, diffusions, and stochastic localization. *arXiv preprint arXiv:2305.10690*, 2023.

Andrea Montanari and Yuchen Wu. Posterior sampling from the spiked models via diffusion processes. *arXiv preprint arXiv:2304.11449*, 2023.

Elchanan Mossel. Deep learning and hierarchal generative models. *arXiv preprint arXiv:1612.09057*, 2016.

Kazusato Oko, Shunta Akiyama, and Taiji Suzuki. Diffusion models are minimax optimal distribution estimators. *arXiv preprint arXiv:2303.01861*, 2023.

Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, et al. Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*, 2018.

Vardan Papyan, Yaniv Romano, and Michael Elad. Convolutional neural networks analyzed via convolutional sparse coding. *The Journal of Machine Learning Research*, 18(1):2887–2938, 2017.

Judea Pearl. Reverend bayes on inference engines: A distributed hierarchical approach. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pp. 129–138. 1982.

Leonardo Petrini, Francesco Cagnetta, Umberto M Tomasini, Alessandro Favero, and Matthieu Wyart. How deep neural networks learn compositional data: The random hierarchy model. *arXiv preprint arXiv:2307.02129*, 2023.

Tomaso Poggio, Hrushikesh Mhaskar, Lorenzo Rosasco, Brando Miranda, and Qianli Liao. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. *International Journal of Automation and Computing*, 14(5):503–519, 2017.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 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*, pp. 234–241. Springer, 2015.

Meyer Scetbon and Zaid Harchaoui. Harmonic decompositions of convolutional networks. In *International Conference on Machine Learning*, pp. 8522–8532. PMLR, 2020.

Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with relu activation function. 2020.

Antonio Sclocchi, Alessandro Favero, and Matthieu Wyart. A phase transition in diffusion models reveals the hierarchical nature of data. *arXiv preprint arXiv:2402.16991*, 2024.

Kulin Shah, Sitan Chen, and Adam Klivans. Learning mixtures of gaussians using the ddpm objective. *arXiv preprint arXiv:2307.01178*, 2023.

Nahian Siddique, Sidike Paheding, Colin P Elkin, and Vijay Devabhaktuni. U-net and its variants for medical image segmentation: A review of theory and applications. *Ieee Access*, 9:82031–82057, 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

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

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Umberto Tomasini and Matthieu Wyart. How deep networks learn sparse and hierarchical data: the sparse random hierarchy model. *arXiv preprint arXiv:2404.10727*, 2024.

Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019. doi: 10.1017/9781108627771.

Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Haonan Wang, Peng Cao, Jiaqi Wang, and Osmar R Zaiane. Uctransnet: rethinking the skip connections in u-net from a channel-wise perspective with transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 2441–2449, 2022.

Zihao Wang and Lei Wu. Theoretical analysis of the inductive biases in deep convolutional networks. *Advances in Neural Information Processing Systems*, 36, 2024.

Colin Wei, Yining Chen, and Tengyu Ma. Statistically meaningful approximation: a case study on approximating turing machines with transformers. *Advances in Neural Information Processing Systems*, 35:12071–12083, 2022.

Christopher Williams, Fabian Falck, George Deligiannidis, Chris C Holmes, Arnaud Doucet, and Saifuddin Syed. A unified framework for u-net design and analysis. *Advances in Neural Information Processing Systems*, 36, 2024.

Alan S Willsky. Multiresolution markov models for signal and image processing. *Proceedings of the IEEE*, 90(8):1396–1458, 2002.

Yuchen Wu, Minshuo Chen, Zihao Li, Mengdi Wang, and Yuting Wei. Theoretical insights for diffusion guidance: A case study for gaussian mixture models. *arXiv preprint arXiv:2403.01639*, 2024.

Lechao Xiao. Eigenspace restructuring: a principle of space and frequency in neural networks. In *Conference on Learning Theory*, pp. 4888–4944. PMLR, 2022.

Yaodong Yu, Sam Buchanan, Druv Pai, Tianzhe Chu, Ziyang Wu, Shengbang Tong, Benjamin D Haeffele, and Yi Ma. White-box transformers via sparse rate reduction. *arXiv preprint arXiv:2306.01129*, 2023a.

Yaodong Yu, Tianzhe Chu, Shengbang Tong, Ziyang Wu, Druv Pai, Sam Buchanan, and Yi Ma. Emergence of segmentation with minimalistic white-box transformers. *arXiv preprint arXiv:2308.16271*, 2023b.

Hui Yuan, Kaixuan Huang, Chengzhuo Ni, Minshuo Chen, and Mengdi Wang. Reward-directed conditional diffusion: Provable distribution estimation and reward improvement. *arXiv preprint arXiv:2307.07055*, 2023.

Jian Zhang and Bernard Ghanem. Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1828–1837, 2018.

Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1529–1537, 2015.

Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*, pp. 3–11. Springer, 2018.

CONTENTS

A PROOF STRATEGY

A.1 PROOF STRATEGY: CONVNETS APPROXIMATE THE BELIEF PROPAGATION ALGORITHM

Lemma 17 introduces a decomposition of $\mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \mu_\star)$ into two components, approximation error and generalization error:

$$\mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \mu_\star) \le \underbrace{\inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\boldsymbol{W}}, \mu_\star)}_{\text{approximation error}} + 2 \cdot \underbrace{\sup_{\boldsymbol{W}\in\mathcal{W}} \left|\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\right|}_{\text{generalization error}}.$$

The bound of generalization error follows a standard approach: employing a chaining argument, the error is controlled by $\tilde{O}(\sqrt{d_{\mathrm{p}}/n})$, where d_{p} represents the number of parameters in the ConvNet class.

In the following, we describe our strategy to control the approximation error: we first present the belief propagation and message passing algorithm for computing the Bayes classifier $\mu_\star$, and then demonstrate that ConvNets are capable of effectively approximating this message passing algorithm.

The belief propagation and message passing algorithm. The belief propagation algorithm operates on input $\boldsymbol{x} \in [S]^d$ and iteratively calculates the beliefs $\{\nu_v^{(\ell)} \in \Delta([S])\}_{\ell\in\{0,\dots,L\}, v\in\mathcal{V}^{(\ell)}}$ as follows:

$$\begin{aligned}
&\nu_v^{(L)}(x_v^{(L)}) = 1\{x_v^{(L)} = x_v\},\\
&\nu_v^{(\ell)}(x_v^{(\ell)}) \propto \textstyle\sum_{x_{\mathcal{C}(v)}^{(\ell+1)}} \prod_{v'\in\mathcal{C}(v)} \Big(\psi_{\iota(v')}^{(\ell+1)}(x_v^{(\ell)}, x_{v'}^{(\ell+1)})\nu_{v'}^{(\ell+1)}(x_{v'}^{(\ell+1)})\Big), \quad \ell = L-1,\dots,0,\\
&\mu_{\mathrm{BP}}(y|\boldsymbol{x}) = \nu_{\mathrm{r}}^{(0)}(y).
\end{aligned} \tag{BP-CLS}$$

Classical results in graphical models verify that the belief propagation algorithm accurately computes the Bayes classifier in this tree graph.

Lemma 1 (BP calculates the Bayes classifier exactly (Pearl, 1982; Wainwright et al., 2008; Mezard & Montanari, 2009))**.** *When applying the belief propagation algorithm (BP-CLS) starting with $\boldsymbol{x} \in [S]^d$, it holds that $\mu_\star(\cdot|\boldsymbol{x}) = \mu_{\mathrm{BP}}(\cdot|\boldsymbol{x})$.*

The belief propagation algorithm can be streamlined into a message passing algorithm, starting with the initialization $h_v^{(L)} = x_v$ for each node v in the highest layer $\mathcal{V}^{(L)}$. The operations are defined as follows:

$$\begin{aligned}
q_v^{(\ell)} &= f_{\iota(v)}^{(\ell)}(h_v^{(\ell)}) \in \mathbb{R}^S, && \ell\in[L], \quad v\in\mathcal{V}^{(\ell)},\\
h_v^{(\ell-1)} &= \mathrm{normalize}\Big(\textstyle\sum_{v'\in\mathcal{C}(v)} q_{v'}^{(\ell)}\Big) \in \mathbb{R}^S, && \ell\in[L], \quad v\in\mathcal{V}^{(\ell-1)},\\
\mu_{\mathrm{MP}}(y|\boldsymbol{x}) &= \mathrm{softmax}(h_{\mathrm{r}}^{(0)}).
\end{aligned} \tag{MP-CLS}$$

The functions $f_\iota^{(L)} : [S] \to \mathbb{R}^S$, $\{f_\iota^{(\ell)} : \mathbb{R}^S \to \mathbb{R}^S\}_{\ell\in[L-1]}$ are defined as:

$$\begin{aligned}
f_\iota^{(L)}(x)_s &= \log \psi_\iota^{(L)}(s,x), && x\in[S], s\in[S],\\
f_\iota^{(\ell)}(h)_s &= \log \textstyle\sum_{a\in[S]} \psi_\iota^{(\ell)}(s,a)e^{h_a}, && h\in\mathbb{R}^S, s\in[S], \ell\in[L-1].
\end{aligned} \tag{22}$$

We note that the normalization operator in (MP-CLS) is non-essential and could be dropped; however, we include it to ensure the formulation closely mirrors (ConvNet), offering a technical benefit.

The subsequent proposition affirms that message passing is essentially equivalent to belief propagation:

Proposition 3 (BP reduces to MP)**.** *Consider the belief propagation algorithm and the message passing algorithm, both starting from $\boldsymbol{x} \in [S]^d$, as in Eq. (BP-CLS) and (MP-CLS). Then we have $\nu_v^{(\ell)}(\cdot) = \mathrm{softmax}(h_v^{(\ell)})$ for all $0 \le \ell \le L-1$ and $v \in \mathcal{V}^{(\ell)}$. In particular, we have $\mu_{\mathrm{BP}}(\cdot|\boldsymbol{x}) = \mu_{\mathrm{MP}}(\cdot|\boldsymbol{x}) = \mu_\star(\cdot|\boldsymbol{x})$.*

The proof of Proposition 3 is presented in Section C.

Approximating message passing with ConvNets. By comparing the message passing algorithm (MP-CLS) alongside the ConvNet (ConvNet), the primary distinction lies in the nonlinear functions used: $f_\iota^{(\ell)}$ versus $\mathrm{NN}_\iota^{(\ell)}$. Given the expression $f_\iota^{(\ell)}(h)_s = \log \sum_{a\in[S]} \psi_\iota^{(\ell)}(s,a)e^{h_a}$, it becomes evident that approximating the logarithmic and exponential functions using one-hidden-layer networks enables $f_\iota^{(\ell)}(h)$ to be effectively approximated by a two-hidden-layer neural network. This leads to the following theorem:

Theorem 4 (ConvNets approximation of Bayes classifier)**.** *Let Assumption 1 and 2 hold. For any $\delta > 0$, take*

$$D = 4\lceil S^2K^2d\cdot 3^L/\delta\rceil, \qquad B = \mathrm{Poly}(d, S, K, 3^L, 1/\delta).$$

Then there exists $\boldsymbol{W} \in \mathcal{W}_{d,\underline{m},L,S,D,B}$ as in Eq. (9), such that defining $\mu_{\mathrm{NN}}^{\boldsymbol{W}}$ as in Eq. (ConvNet), we have

$$\max_{y\in[S],\boldsymbol{x}\in[S]^d} \left|\log \mu_\star(y|\boldsymbol{x}) - \log \mu_{\mathrm{NN}}^{\boldsymbol{W}}(y|\boldsymbol{x})\right| \le \delta.$$

The proof of Theorem 4 is detailed in Section D.

A.2 Proof strategy: U-Nets approximate the belief propagation algorithm

The proof strategy for the denoising task closely aligns with that of the classification task as detailed in Section A.1.

Lemma 22 introduces a decomposition of $\mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \boldsymbol{m}_\star)$ into two components, approximation error and generalization error:

$$\mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \boldsymbol{m}_\star) \le \underbrace{\inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}, \boldsymbol{m}_\star)}_{\text{approximation error}} + 2\cdot \underbrace{\sup_{\boldsymbol{W}\in\mathcal{W}} \left|\widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}})\right|}_{\text{generalization error}}.$$

The bound of generalization error follows a standard approach: employing a chaining argument, the error is controlled by $\tilde{O}(\sqrt{d_{\mathrm{p}}/n})$, where d_{p} represents the number of parameters in the U-Net class.

In the following, we describe our strategy to control the approximation error: we first present the belief propagation and message passing algorithm for computing the Bayes denoiser $\boldsymbol{m}_\star$, and then demonstrate that U-Nets are capable of effectively approximating this message passing algorithm.

The belief propagation and message passing algorithm. The belief propagation algorithm operates on input $\boldsymbol{z}\in\mathbb{R}^d$ and iteratively calculates the beliefs $\{\nu_{\downarrow,v}^{(\ell)}, \nu_{\uparrow,v}^{(\ell)} \in \Delta([S])\}_{\ell\in\{0,\dots,L\}, v\in\mathcal{V}^{(\ell)}}$ as follows:

$$\begin{aligned}
&\nu_{\downarrow,v}^{(L)}(x_v^{(L)}) = \psi^{(L+1)}(x_v^{(L)}, z_v),\\
&\nu_{\downarrow,v}^{(\ell)}(x_v^{(\ell)}) \propto \textstyle\sum_{x_{\mathcal{C}(v)}^{(\ell+1)}} \prod_{v'\in\mathcal{C}(v)} \Big(\psi_{\iota(v')}^{(\ell+1)}(x_v^{(\ell)}, x_{v'}^{(\ell+1)})\nu_{\downarrow,v'}^{(\ell+1)}(x_{v'}^{(\ell+1)})\Big), && \ell = L-1,\dots,0,\\
&\nu_{\uparrow,\mathrm{r}}^{(0)}(x_{\mathrm{r}}^{(0)}) \propto 1,\\
&\nu_{\uparrow,v}^{(\ell)}(x_v^{(\ell)}) \propto \textstyle\sum_{x_{\mathrm{pa}(v)}^{(\ell-1)}, x_{\mathcal{N}(v)}^{(\ell)}} \psi^{(\ell)}(x_{\mathrm{pa}(v)}^{(\ell-1)}, x_{\mathcal{C}(\mathrm{pa}(v))}^{(\ell)})\nu_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)}(x_{\mathrm{pa}(v)}^{(\ell-1)}) \textstyle\prod_{v'\in\mathcal{N}(v)} \nu_{\downarrow,v'}^{(\ell)}(x_{v'}^{(\ell)}), && \ell = 1,\dots,L,\\
&\nu_v^{(L)}(x_v^{(L)}) \propto \nu_{\uparrow,L}(x_v^{(L)})\psi^{(L+1)}(x_v^{(L)}, z_v),\\
&\boldsymbol{m}_{\mathrm{BP}}(\boldsymbol{z})_v = \textstyle\sum_{x_v^{(L)}} x_v^{(L)}\nu_v^{(L)}(x_v^{(L)}), && \text{(BP-DNS)}
\end{aligned}$$

where $\psi^{(L+1)}(x_v^{(L)}, z_v) := \exp\{-(x_v^{(L)} - z_v)^2/2\}$. Classical results in graphical models verify that the belief propagation algorithm accurately computes the Bayes denoiser in this tree graph.

Lemma 2 (BP calculates the Bayes denoiser exactly (Pearl, 1982; Wainwright et al., 2008; Mezard & Montanari, 2009))**.** *When applying the belief propagation algorithm (BP-DNS) starting with $\boldsymbol{z}\in\mathbb{R}^d$, it holds that $\mu_\star(x_v|\boldsymbol{z}) = \nu_v^{(L)}(x_v)$ for $v\in\mathcal{V}^{(L)}$, so that $\boldsymbol{m}_{\mathrm{BP}}(\boldsymbol{z}) = \boldsymbol{m}(\boldsymbol{z})$.*

We remark that the downward-upward structure of belief propagation in generative hierarchical models has been pointed out in the literature (Sclocchi et al., 2024).

The belief propagation algorithm can be streamlined into a message passing algorithm, starting with the initialization $h_{\downarrow,v}^{(L)} = (-(x - z_v)^2/2)_{x\in[S]} \in \mathbb{R}^S$ for each node v in the highest layer $\mathcal{V}^{(L)}$. The operations are defined as follows:

$$\begin{aligned}
q_{\downarrow,v}^{(\ell)} &= f_{\downarrow,\iota(v)}^{(\ell)}(\text{normalize}(h_{\downarrow,v}^{(\ell)})) \in \mathbb{R}^S, && \ell \in [L], \quad v \in \mathcal{V}^{(\ell)}, \\
h_{\downarrow,v}^{(\ell-1)} &= \textstyle\sum_{v'\in\mathcal{C}(v)} q_{\downarrow,v'}^{(\ell)} \in \mathbb{R}^S, && \ell \in [L], \quad v \in \mathcal{V}^{(\ell-1)}, \\
u_{\uparrow,v}^{(\ell)} &= b_{\uparrow,\text{pa}(v)}^{(\ell-1)} \in \mathbb{R}^S, \quad (\text{with } b_{\uparrow,\text{r}}^{(0)} = h_{\downarrow,\text{r}}^{(0)}) && \ell \in [L], \quad v \in \mathcal{V}^{(\ell)}, \qquad \text{(MP-DNS)} \\
b_{\uparrow,v}^{(\ell)} &= f_{\uparrow,\iota(v)}^{(\ell)}(\text{normalize}(u_{\uparrow,v}^{(\ell)} - q_{\downarrow,v}^{(\ell)})) + h_{\downarrow,v}^{(\ell)} \in \mathbb{R}^S, && \ell \in [L], \quad v \in \mathcal{V}^{(\ell)}, \\
\boldsymbol{m}_{\text{MP}}(\boldsymbol{z})_v &= \textstyle\sum_{s\in[S]} s \cdot \text{softmax}(b_{\uparrow,v}^{(L)})_s, && v \in \mathcal{V}^{(L)}.
\end{aligned}$$

The functions $\{f_{\downarrow,\iota}^{(\ell)}, f_{\uparrow,\iota}^{(\ell)} : \mathbb{R}^S \to \mathbb{R}^S\}_{\ell\in[L]}$ are defined as

$$\begin{aligned}
f_{\downarrow,\iota}^{(\ell)}(h)_s &= \log \textstyle\sum_{a\in[S]} \psi_\iota^{(\ell)}(s,a) e^{h_a}, \qquad h \in \mathbb{R}^S, s \in [S], \ell \in [L], \\
f_{\uparrow,\iota}^{(\ell)}(h)_s &= \log \textstyle\sum_{a\in[S]} \psi_\iota^{(\ell)}(a,s) e^{h_a}, \qquad h \in \mathbb{R}^S, s \in [S], \ell \in [L].
\end{aligned} \tag{23}$$

We note that the normalization operator in (MP-DNS) is non-essential and could be dropped; however, we include it to ensure the formulation closely mirrors (UNet), offering a technical benefit.

The subsequent proposition affirms that message passing is essentially equivalent to belief propagation:

Proposition 5 (BP reduces to MP)**.** *Consider the belief propagation algorithm and the message passing algorithm, both with input $\boldsymbol{z} \in \mathbb{R}^d$, as in Eq. (BP-DNS) and (MP-DNS). Then we have $\nu_{\downarrow,v}^{(\ell)}(\cdot) = \text{softmax}(h_{\downarrow,v}^{(\ell)})$ and $\nu_{\uparrow,v}^{(\ell)}(\cdot) = \text{softmax}(b_{\uparrow,v}^{(\ell)} - h_{\downarrow,v}^{(\ell)})$, and $\nu_v^{(L)}(\cdot) = \text{softmax}(b_{\uparrow,v}^{(L)})$. In particular, $\boldsymbol{m}_{\text{MP}}(\boldsymbol{z}) = \boldsymbol{m}_{\text{BP}}(\boldsymbol{z}) = \boldsymbol{m}(\boldsymbol{z})$.*

The proof of Proposition 5 is presented in Section F.

Approximating message passing with ConvNets. By comparing the message passing algorithm (MP-DNS) alongside the U-Net (UNet), the primary distinction lies in the nonlinear functions used: $f_{\diamond,\iota}^{(\ell)}$ versus $\text{NN}_{\diamond,\iota}^{(\ell)}$. Notably, $f_{\diamond,\iota}^{(\ell)}$ entails a log-sum-exponential structure. This structure suggests that approximating the logarithmic and exponential functions with one-hidden-layer neural networks can allow $f_{\diamond,\iota}^{(\ell)}(h)$ to be effectively approximated by a two-hidden-layer neural network. This leads to the following theorem:

Theorem 6 (U-Nets approximation of Bayes denoiser)**.** *Let Assumption 1 and 2 hold. For any $\delta > 0$, take*

$$D = 4\lceil S^3K^2d \cdot 18^L/\delta \rceil, \qquad B = \text{Poly}(d, S, K, 18^L, 1/\delta).$$

Then there exists $\boldsymbol{W} \in \mathcal{W}_{d,\underline{m},L,S,D,B}$ as in Eq. (18), such that defining $\boldsymbol{m}_{\text{NN}}^{\boldsymbol{W}}$ as in Eq. (UNet), we have

$$\sup_{\boldsymbol{z}\in\mathbb{R}^d} \|\boldsymbol{m}(\boldsymbol{z}) - \boldsymbol{m}_{\text{NN}}(\boldsymbol{z})\|_\infty \le \delta.$$

The proof of Theorem 6 is detailed in Section G.

B TECHNICAL PRELIMINARIES

We here present a bound on the supremum of sub-Gaussian processes, whose proof was based on the chaining argument.

Lemma 3 (Proposition A.4 of (Bai et al., 2024))**.** *Suppose that $\{X_w\}_{w\in\Theta}$ is a zero-mean random process given by*

$$X_w \equiv \frac{1}{n}\sum_{i=1}^n f(z_i; w) - \mathbb{E}_z[f(z; w)],$$

where $z_1, \cdots, z_n$ are i.i.d samples from a distribution $\mathbb{P}_z$ such that the following assumption holds:

(a) The index set Θ is equipped with a distance ρ and diameter B_{p}. Further, assume that for some constant A_{p}, for any ball Θ' of radius r in Θ, the covering number admits upper bound $\log N(\Delta; \Theta', \rho) \le d_{\mathrm{p}} \log(2A_{\mathrm{p}} r/\Delta)$ for all $0 < \Delta \le 2r$.

(b) For any fixed $w \in \Theta$ and z sampled from $\mathbb{P}_z$, the random variable $f(z; w) - \mathbb{E}_z[f(z; w)]$ is a σ-sub-Gaussian random variable ($\mathbb{E}[e^{\lambda[f(z;w) - \mathbb{E}_{z'}[f(z';w)]]}] \le e^{\lambda^2\sigma^2/2}$ for any $\lambda \in \mathbb{R}$).

(c) For any $w, w' \in \Theta$ and z sampled from $\mathbb{P}_z$, the random variable $f(z; w) - f(z; w')$ is a $\sigma'\rho(w, w')$-sub-Gaussian random variable ($\mathbb{E}[e^{\lambda[f(z;w) - f(z;w')]}] \le e^{\lambda^2(\sigma')^2\rho^2(w,w')/2}$ for any $\lambda \in \mathbb{R}$).

Then with probability at least $1 - \eta$, it holds that

$$\sup_{w \in \Theta} |X_w| \le C\sigma \sqrt{\frac{d_{\mathrm{p}} \cdot \log(2A_{\mathrm{p}}(1 + B_{\mathrm{p}}\sigma'/\sigma)) + \log(1/\eta)}{n}},$$

where C is a universal constant.

We next present a simple inequality used in the proof of Theorem 1.

Lemma 4 (From log ratio bound to square distance bound)**.** *Let p and q be two probability measures on $\Delta([S])$ such that*

$$\max_{y \in [S]} \Big| \log p(y) - \log q(y) \Big| \le \delta.$$

Then we have

$$\textstyle\sum_{s=1}^{S} (p(s) - q(s))^2 \le (e^{\delta} - 1)^2.$$

Proof of Lemma 4. The lemma is by the fact that $|p(y) - q(y)| \le (\exp\{|\log p(y) - \log q(y)|\} - 1) \cdot p(y)$. □

C Proof of Proposition 3

Proof of Proposition 3. By Eq. (MP-CLS) and (22), defining $\nu_v^{(\ell)}(\cdot) = \mathrm{softmax}(h_v^{(\ell)})$, then for $\ell \le L - 2$, we get

$$\begin{aligned}
\nu_v^{(\ell)}(x_v^{(\ell)}) &\propto \prod_{v' \in \mathcal{C}(v)} \Big(\sum_{a \in [S]} \psi_{\iota(v')}^{(\ell+1)}(x_v^{(\ell)}, a) e^{(h_v^{(\ell+1)})_a} \Big) \\
&\propto \sum_{x_{\mathcal{C}(v)}^{(\ell+1)}} \prod_{v' \in \mathcal{C}(v)} \Big(\psi_{\iota(v')}^{(\ell+1)}(x_v^{(\ell)}, x_{v'}^{(\ell+1)}) \nu_{v'}^{(\ell+1)}(x_{v'}^{(\ell+1)}) \Big).
\end{aligned}$$

This coincides with the update rule in Eq. (BP-CLS). For $\ell = L - 1$, we get

$$\nu_v^{(L-1)}(x_v^{(L-1)}) \propto \prod_{v' \in \mathcal{C}(v)} \Big(\sum_{a \in [S]} \psi_{\iota(v')}^{(L)}(x_v^{(L-1)}, a) 1\{a = x_{v'}\} \Big) \propto \prod_{v' \in \mathcal{C}(v)} \psi_{\iota(v')}^{(L)}(x_v^{(L-1)}, x_{v'}).$$

This again coincides with the update rule in Eq. (BP-CLS). This finishes the proof of Proposition 3. □

D Proof of Theorem 4

Proof of Theorem 4. By Lemma 13, take $M_1^{(\ell)} = \lceil 2SKd3^L/\delta \rceil + 1$ and $M_2^{(\ell)} = \lceil 2SK^2 d3^L/\delta \rceil + 1$. Then there exists $\{(a_j, w_j, b_j)\}_{j \in [M_1^{(\ell)}]}$ and $\{(\bar{a}_j, \bar{w}_j, \bar{b}_j)\}_{j \in [M_2^{(\ell)}]}$ with

$$\sup_j |a_j| \le 2SK, \quad \sup_j |w_j| \le 1, \quad \sup_j |b_j| \le SK, \quad \sup_j |\bar{a}_j| \le 2, \quad \sup_j |\bar{w}_j| \le 1, \quad \sup_j |\bar{b}_j| \le \log(4 \cdot 3^L SdK^2/\delta),$$

such that defining

$$\log_{\delta\star}(x) = \textstyle\sum_{j=1}^{M_1^{(\ell)}} a_j \cdot \mathrm{ReLU}(w_j x + b_j), \quad \exp_{\delta\star}(x) = \sum_{j=1}^{M_2^{(\ell)}} \bar{a}_j \cdot \mathrm{ReLU}(\bar{w}_j x + \bar{b}_j),$$

and defining $f_\iota^{(\ell)}(h), \bar{f}_\iota^{(\ell)}(h) \in \mathbb{R}^S$ by

$$f_\iota^{(\ell)}(h)_i = \log \sum_{j=1}^S \psi_\iota^{(\ell)}(i,j) \exp(h_j), \quad \bar{f}_\iota^{(\ell)}(h)_i = \log_{\delta\star} \sum_{j=1}^S \psi_\iota^{(\ell)}(i,j) \exp_{\delta\star}(h_j), \quad \forall i \in [S],$$

we have

$$\sup_{\max_i h_i = 0} \|f_\iota^{(\ell)}(h) - \bar{f}_\iota^{(\ell)}(h)\|_\infty \le \delta/(d3^L). \tag{24}$$

In addition, by Lemma 15, take $M_1^{(L)} = \lceil 3^L dK/\delta \rceil + 1$ and $M_2^{(L)} = 3$. Then there exists $\{(a_j^{(L)}, w_j^{(L)}, b_j^{(L)})\}_{j \in [M_1^{(L)}]}$ and $\{(\bar{a}_j^{(L)}, \bar{w}_j^{(L)}, \bar{b}_j^{(L)})\}_{j \in [M_2^{(L)}]}$ with

$$\sup_j |a_j^{(L)}| \le 2K, \quad \sup_j |w_j^{(L)}| \le 1, \quad \sup_j |b_j^{(L)}| \le SK, \quad \sup_j |\bar{a}_j^{(L)}| \le 4, \quad \sup_j |\bar{w}_j^{(L)}| \le 1, \quad \sup_j |\bar{b}_j^{(L)}| \le 1,$$

such that defining

$$\log_\delta(x) = \sum_{j=1}^{M_1^{(L)}} a_j^{(L)} \cdot \mathrm{ReLU}(w_j^{(L)} x + b_j^{(L)}), \quad \mathrm{Ind}(x) = \sum_{j=1}^{M_2^{(L)}} \bar{a}_j^{(L)} \cdot \mathrm{ReLU}(\bar{w}_j^{(L)} x + \bar{b}_j^{(L)}),$$

and defining $f_\iota^{(L)}(h), \bar{f}_\iota^{(L)}(h) \in \mathbb{R}^S$ by

$$f_\iota^{(L)}(x)_i = \log \sum_{j=1}^S \psi_\iota^{(L)}(i,j) 1(x=j), \quad \bar{f}_\iota^{(L)}(x)_i = \log_\delta \sum_{j=1}^S \psi_\iota^{(L)}(i,j) \mathrm{Ind}(x-j), \quad \forall i \in [S],$$

we have

$$\sup_{x \in [S]} \|f_\iota^{(L)}(x) - \bar{f}_\iota^{(L)}(x)\|_\infty \le \delta/(d3^L). \tag{25}$$

By Eq. (24) and (25) and Lemma 5, taking $h_{\mathrm{r}}^{(0)} \in \mathbb{R}^S$ to be as defined in Eq. (MP-CLS) and $\bar{h}_{\mathrm{r}}^{(0)} \in \mathbb{R}^S$ to be as defined in Eq. (A-MP-CLS) with $\{\bar{f}_\iota^{(\ell)}\}_{\ell \in [L], \iota \in [m^{(\ell)}]}$ as defined above, we have

$$\|h_{\mathrm{r}}^{(0)} - \bar{h}_{\mathrm{r}}^{(0)}\|_\infty \le [\delta/(d3^L)] \times \prod_{1 \le \ell \le L} (2m^{(\ell)} + 1) \le \delta.$$

As a consequence, we just need to show that the approximate version of message passing algorithm as in Eq. (A-MP-CLS) could be cast as a neural network.

Indeed, by Lemma 14, there exist two-hidden-layer neural networks (for $\ell \in [L-1]$ and $\iota \in [m^{(\ell)}]$)

$$\mathrm{NN}_{W_{1,\iota}^{(\ell)}, W_{2,\iota}^{(\ell)}, W_{3,\iota}^{(\ell)}}(h) = W_{1,\iota}^{(\ell)} \cdot \mathrm{ReLU}(W_{2,\iota}^{(\ell)} \cdot \mathrm{ReLU}(W_{3,\iota}^{(\ell)} \cdot [h; 1])),$$

with $W_{1,\iota}^{(\ell)} \in \mathbb{R}^{S \times SM_1^{(\ell)}}$, $W_{2,\iota}^{(\ell)} \in \mathbb{R}^{SM_1^{(\ell)} \times (SM_2^{(\ell)}+1)}$, $W_{3,\iota}^{(\ell)} \in \mathbb{R}^{(SM_2^{(\ell)}+1) \times (S+1)}$, and

$$\|W_{1,\iota}^{(\ell)}\|_{\max} \le 2SK, \quad \|W_{2,\iota}^{(\ell)}\|_{\max} \le \mathrm{Poly}(SKM_1^{(\ell)} M_2^{(\ell)}), \quad \|W_{3,\iota}^{(\ell)}\|_{\max} \le \log(4SK^2/\delta).$$

such that

$$\mathrm{NN}_{W_{1,\iota}^{(\ell)}, W_{2,\iota}^{(\ell)}, W_{3,\iota}^{(\ell)}}(h) = \bar{f}_\iota^{(\ell)}(h), \quad \forall h \in \mathbb{R}^S \text{ such that } \max_j h_j = 0.$$

Furthermore, by Lemma 16, there exist two-hidden-layer neural networks (for $\iota \in [m^{(L)}]$)

$$\mathrm{NN}_{W_{1,\iota}^{(L)}, W_{2,\iota}^{(L)}, W_{3,\iota}^{(L)}}(x) = W_{1,\iota}^{(L)} \cdot \mathrm{ReLU}(W_{2,\iota}^{(L)} \cdot \mathrm{ReLU}(W_{3,\iota}^{(L)} \cdot [x; 1])),$$

with $W_{1,\iota}^{(L)} \in \mathbb{R}^{S \times SM_1^{(L)}}$, $W_{2,\iota}^{(L)} \in \mathbb{R}^{SM_1^{(L)} \times (SM_2^{(L)}+1)}$, $W_{3,\iota}^{(L)} \in \mathbb{R}^{(SM_2^{(L)}+1) \times 2}$,

$$\|W_{1,\iota}^{(L)}\|_{\max} \le 2K, \quad \|W_{2,\iota}^{(L)}\|_{\max} \le \mathrm{Poly}(SKM_1^{(L)} M_2^{(L)}), \quad \|W_{3,\iota}^{(L)}\|_{\max} \le 1.$$

such that

$$\mathrm{NN}_{W_{1,\iota}^{(L)}, W_{2,\iota}^{(L)}, W_{3,\iota}^{(L)}}(x) = \bar{f}_\iota^{(L)}(x), \quad \forall x \in [S].$$

This proves that the approximate version of message passing as in Eq. (A-MP-CLS) can be cast into the convolutional neural network as in Eq. (ConvNet) with proper choice of dimension

$$D \ge \max_{\ell \in [L]} \{SM_1^{(\ell)}, SM_2^{(\ell)} + 1\} = S \times \Big(\lceil 2SK^2 d3^L/\delta \rceil + 1 \Big) + 1,$$

and norm of the weights. This finishes the proof of Theorem 4. □

D.1 AUXILLARY LEMMAS

Lemma 5 (Error propagation of the approximate version of message passing in classification)**.** *Assume we have functions $f_\iota^{(\ell)}$ and $\bar{f}_\iota^{(\ell)}$ such that*

$$\begin{aligned} &\|f_\iota^{(L)}(x) - \bar{f}_\iota^{(L)}(x)\|_\infty \le \delta, \quad \forall x \in [S], \\ &\|f_\iota^{(\ell)}(h) - \bar{f}_\iota^{(\ell)}(h)\|_\infty \le \delta, \quad \forall h \in \mathbb{R}^S \text{ such that } \max_{j\in[S]} h_j = 0, \ \ \forall \ell \le L-1. \end{aligned} \tag{26}$$

Furthermore, consider the following approximate version of message passing algorithm with initialization $\bar{h}_v^{(L)} = x_v$ for $v \in \mathcal{V}^{(L)}$:

$$\begin{aligned} \bar{q}_v^{(\ell)} &= \bar{f}_{\iota(v)}^{(\ell)}(\bar{h}_v^{(\ell)}) \in \mathbb{R}^S, && \ell \in [L-1], \quad v \in \mathcal{V}^{(\ell)}, \\ \bar{h}_v^{(\ell-1)} &= \mathrm{normalize}\Big(\textstyle\sum_{v'\in\mathcal{C}(v)} \bar{q}_{v'}^{(\ell)}\Big) \in \mathbb{R}^S, && \ell \in [L-1], \quad v \in \mathcal{V}^{(\ell-1)}. \end{aligned} \tag{A-MP-CLS}$$

Taking $h_{\mathrm{r}}^{(0)} \in \mathbb{R}^S$ to be as defined in Eq. (MP-CLS) and $\bar{h}_{\mathrm{r}}^{(0)} \in \mathbb{R}^S$ to be as defined in Eq. (A-MP-CLS), we have

$$\|h_{\mathrm{r}}^{(0)} - \bar{h}_{\mathrm{r}}^{(0)}\|_\infty \le \delta \times \prod_{1\le\ell\le L} (2m^{(\ell)} + 1).$$

Proof of Lemma 5. We prove this lemma by induction, aiming to show that for any $\ell \in [L-1]$ we have

$$\|\bar{h}_v^{(\ell)} - h_v^{(\ell)}\|_\infty \le 2m^{(\ell+1)} \textstyle\prod_{k=\ell+2}^{L}(2m^{(k)}+1)\delta, \quad \forall v \in \mathcal{V}^{(\ell)}. \tag{27}$$

To prove the formula for $\ell = L-1$, since $h_v^{(L)} = \bar{h}_v^{(L)}$, by Eq. (26), we get

$$\|\bar{q}_v^{(L)} - q_v^{(L)}\|_\infty \le \delta, \quad \forall v \in \mathcal{V}^{(L)}.$$

By Lemma 7, we get

$$\|h_v^{(L-1)} - \bar{h}_v^{(L-1)}\|_\infty \le 2\Big\| \textstyle\sum_{v'\in\mathcal{C}(v)} (\bar{q}_{v'}^{(L)} - q_{v'}^{(L)})\Big\|_\infty \le 2m^{(L)}\delta, \quad \forall v \in \mathcal{V}^{(L-1)}.$$

This proves the formula (27) for $\ell = L-1$.

Assuming that (27) holds at the layer ℓ, by the update formula, we have

$$\begin{aligned} &\|\bar{q}_v^{(\ell)} - q_v^{(\ell)}\|_\infty = \|\bar{f}_{\iota(v)}^{(\ell)}(\bar{h}_v^{(\ell)}) - f_{\iota(v)}^{(\ell)}(h_v^{(\ell)})\|_\infty \\ &\le \|\bar{f}_{\iota(v)}^{(\ell)}(\bar{h}_v^{(\ell)}) - f_{\iota(v)}^{(\ell)}(\bar{h}_v^{(\ell)})\|_\infty + \|f_{\iota(v)}^{(\ell)}(h_v^{(\ell)}) - f_{\iota(v)}^{(\ell)}(\bar{h}_v^{(\ell)})\|_\infty \\ &\le \delta + 2m^{(\ell+1)} \textstyle\prod_{k=\ell+2}^{L}(2m^{(k)}+1)\delta \le \prod_{k=\ell+1}^{L}(2m^{(k)}+1)\delta, \end{aligned}$$

where the middle inequality is by the assumption of $f_\iota^{(\ell)}$ and by Lemma 6. By Lemma 7, we get

$$\|\bar{h}_v^{(\ell-1)} - h_v^{(\ell-1)}\|_\infty \le 2m^{(\ell)} \textstyle\prod_{k=\ell+1}^{L}(2m^{(k)}+1)\delta, \quad \forall v \in \mathcal{V}^{(\ell)}.$$

This proves Lemma 5 by the induction argument. □

Lemma 6 (Non-expansiveness of log-sum-exponential)**.** *For $h \in \mathbb{R}^S$ and $\Psi \in \mathbb{R}^{S\times S}$, define $f(h) \in \mathbb{R}^S$ by*

$$f(h)_i = \log \textstyle\sum_{j=1}^{S} \Psi_{ij} \exp(h_j), \quad \forall i \in [S].$$

Then for $h_1, h_2 \in \mathbb{R}^S$, we have

$$\|f(h_1) - f(h_2)\|_\infty \le \|h_1 - h_2\|_\infty.$$

Proof of Lemma 6. Fix $i \in [S]$. We have

$$\nabla_h f(h)_i = \Big(\frac{\Psi_{ij}\exp(h_j)}{\sum_{k\in[S]}\Psi_{ik}\exp(h_k)}\Big)_{j\in[S]},$$

so that $\|\nabla_h f(h)_i\|_1 = 1$. By intermediate value theorem, we have

$$|f(h_1)_i - f(h_2)_i| = |\nabla_h f(\xi)_i^{\mathbb{T}}(h_1 - h_2)| \le \|\nabla_h f(\xi)_i^{\mathbb{T}}\|_1 \|h_1 - h_2\|_\infty = \|h_1 - h_2\|_\infty.$$

This proves Lemma 6. □

Lemma 7 (Lipschitzness of the normalization operator)**.** *For $h \in \mathbb{R}^S$, define* $\mathrm{normalize}(h) \in \mathbb{R}^S$ *by*

$$\mathrm{normalize}(h)_i = h_i - \max_j h_j, \quad \forall i \in [S].$$

Then for $h_1, h_2 \in \mathbb{R}^S$, we have

$$\|\mathrm{normalize}(h_1) - \mathrm{normalize}(h_2)\|_\infty \le 2\|h_1 - h_2\|_\infty.$$

Proof of Lemma 7. Note we have the following inequality

$$|\max_j h_{1,j} - \max_j h_{2,j}| \le \|h_1 - h_2\|_\infty,$$

so that

$$|\mathrm{normalize}(h_1)_i - \mathrm{normalize}(h_2)_i| \le \|h_1 - h_2\|_\infty + |\max_j h_{1,j} - \max_j h_{2,j}| \le 2\|h_1 - h_2\|_\infty.$$

This completes the proof of Lemma 7. □

Lemma 8 (ReLU approximation of the exponential function)**.** *For any $\delta > 0$, take $M = \lceil 1/\delta \rceil + 1 \in \mathbb{N}$. Then there exists $\{(a_j, w_j, b_j)\}_{j\in[M]}$ with*

$$\sup_j |a_j| \le 2, \quad \sup_j |w_j| \le 1, \quad \sup_j |b_j| \le \log M, \tag{28}$$

such that defining $\exp_\delta : \mathbb{R} \to \mathbb{R}$ by

$$\exp_\delta(x) = \textstyle\sum_{j=1}^M a_j \cdot \mathrm{ReLU}(w_j x + b_j),$$

we have $\exp_\delta$ is non-decreasing on $(-\infty, 0]$, and

$$\sup_{x\in(-\infty,0]} |\exp(x) - \exp_\delta(x)| \le \delta, \quad \exp_\delta(0) = 1.$$

Proof of Lemma 8. Define $e_j = j/(M-1)$, $b_j = -\log(e_j)$ for $j \in [M-1]$, $a_1 = (e_2 - e_1)/(b_2 - b_1)$ and $a_j = (e_{j+1} - e_j)/(b_{j+1} - b_j) - (e_j - e_{j-1})/(b_j - b_{j-1})$ for $2 \le j \le M-2$. Furthermore, define

$$\exp_\delta(x) = \sum_{j=1}^{M-2} a_j \mathrm{ReLU}(x + b_j) + \mathrm{ReLU}(-x + e_1) - \mathrm{ReLU}(-x).$$

Then we have $\exp_\delta(-b_j) = e_j$ for $j \in [M-1]$, and $\exp_\delta$ is piece-wise linear and non-decreasing on $(-\infty, 0]$. Note that we also have $\exp(-b_j) = e_j$ for $j \in [M-1]$, and $\exp$ is increasing on $(-\infty, 0]$. This proves that $\sup_{x\in(-\infty,0]} |\exp(x) - \exp_\delta(x)| \le \delta$. Furthermore, it is easy to see that $\exp_\delta(0) = 1$ and $\exp_\delta$ is non-decreasing on $(-\infty, 0]$. Finally, since $\exp$ is 1-Lipschitz, it is easy to see that $\sup_{j\in[M]} |a_j| \le 2$. It is also easy to see the other parts of Eq. (28) are satisfied, and this proves Lemma 8. □

Lemma 9 (ReLU approximation of the logarithm function)**.** *For any $A > 0$, $\delta > 0$, take $M = \lceil 2A/\delta \rceil + 1 \in \mathbb{N}$. Then there exists $\{(a_j, w_j, b_j)\}_{j\in[M]}$ with*

$$\sup_j |a_j| \le 2A, \quad \sup_j |w_j| \le 1, \quad \sup_j |b_j| \le A, \tag{29}$$

such that defining $\log_\delta : \mathbb{R} \to \mathbb{R}$ by

$$\log_\delta(x) = \textstyle\sum_{j=1}^M a_j \cdot \mathrm{ReLU}(w_j x + b_j),$$

we have $\log_\delta$ is non-decreasing on $[1/A, A]$, and

$$\sup_{x\in[1/A,A]} |\log(x) - \log_\delta(x)| \le \delta.$$

Proof of Lemma 9. The proof of Lemma 9 is similar to Lemma 8. □

Lemma 10 (ReLU approximation of indicator function)**.** *Define*

$$\mathrm{Ind}(x) = 2\mathrm{ReLU}(x - 1/2) + 2\mathrm{ReLU}(x + 1/2) - 4\mathrm{ReLU}(x),$$

we have

$$1(x = j) = \mathrm{Ind}(x - j), \quad \forall j, x \in \mathbb{Z}.$$

Proof of Lemma 10. The lemma holds by direct calculation. □

Lemma 11 (Log-sum-exponential approximation)**.** *Assume* $\log_{\delta_1} : \mathbb{R} \to \mathbb{R}$ *and* $\exp_{\delta_2} : \mathbb{R} \to \mathbb{R}$ *are such that,*

$$\sup_{x \in [1/K, SK]} |\log(x) - \log_{\delta_1}(x)| \le \delta_1, \quad \sup_{x \in (-\infty, 0]} |\exp(x) - \exp_{\delta_2}(x)| \le \delta_2,$$
$$\exp_{\delta_2}(0) = 1, \quad \exp_{\delta_2} \text{ is non-decreasing on } (-\infty, 0].$$

Assume that $1/K \le \min_{ij} \Psi_{ij} \le \max_{ij} \Psi_{ij} \le K$. *Define* $f(h), f_{\delta_1,\delta_2}(h) \in \mathbb{R}^S$ *by*

$$f(h)_i = \log \textstyle\sum_{j=1}^S \Psi_{ij} \exp(h_j), \quad f_{\delta_1,\delta_2}(h)_i = \log_{\delta_1} \textstyle\sum_{j=1}^S \Psi_{ij} \exp_{\delta_2}(h_j), \quad \forall i \in [S].$$

Then we have

$$\sup_{\max_i h_i = 0} \|f(h) - f_{\delta_1,\delta_2}(h)\|_\infty \le \delta_1 + SK^2\delta_2.$$

Proof of Lemma 11. We have

$$|f(h)_i - f_{\delta_1,\delta_2}(h)_i| = |\log\langle \Psi_{i:}, \exp(h)\rangle - \log_{\delta_1}\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle|$$
$$\le |\log\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle - \log_{\delta_1}\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle| + |\log\langle \Psi_{i:}, \exp(h)\rangle - \log\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle|.$$

For the first term, since $\exp_{\delta_2}(h) \le 1$ for all $h \le 0$, we have $\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle \le S \max_{ij} \Psi_{ij} \le SK$. Furthermore, since $\max_i h_i = 0$ and $\exp_{\delta_2}(0) = 1$, we have $\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle \ge \min_{ij} \Psi_{ij} \ge 1/K$. As a consequence, by assumption, we have

$$|\log\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle - \log_{\delta_1}\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle| \le \delta_1.$$

For the second term, since both $\langle \Psi_{i:}, \exp(h)\rangle$ and $\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle$ are within $[1/K, SK]$ on which log function has Lipschitz constant K, we have

$$|\log\langle \Psi_{i:}, \exp(h)\rangle - \log\langle \Psi_{i:}, \exp_{\delta_2}(h)\rangle| \le SK \max_{ij} \Psi_{ij} \cdot \|\exp(h) - \exp_{\delta_2}(h)\|_\infty \le SK^2\delta_2.$$

This finishes the proof of Lemma 11. □

Lemma 12 (Log-Psi approximation)**.** *Assume* $\log_\delta : \mathbb{R} \to \mathbb{R}$ *is such that,*

$$\sup_{x \in [1/K, K]} |\log(x) - \log_\delta(x)| \le \delta.$$

Assume that $1/K \le \min_{ij} \Psi_{ij} \le \max_{ij} \Psi_{ij} \le K$. *For* $x \in [S]$, *define* $f(x), f_\delta(x) \in \mathbb{R}^S$ *by*

$$f(x)_i = \log \textstyle\sum_{j=1}^S \Psi_{ij} 1(x = j), \quad f_\delta(x)_i = \log_\delta \textstyle\sum_{j=1}^S \Psi_{ij} 1(x = j), \quad \forall i \in [S].$$

Then we have

$$\sup_{x \in [S]} \|f(x) - f_\delta(x)\|_\infty \le \delta.$$

Proof of Lemma 12. For any fixed $x \in [S]$ and $i \in [S]$, we have

$$|f(x)_i - f_\delta(x)_i| = |\log \Psi_{ix} - \log_\delta \Psi_{ix}| \le \delta,$$

where the last inequality is by assumption. This proves Lemma 12. □

Lemma 13 (ReLU approximation of log-sum-exponential)**.** *Assume that* $1/K \le \min_{ij} \Psi_{ij} \le \max_{ij} \Psi_{ij} \le K$*. For any* $\delta > 0$*, take* $M_1 = \lceil 2SK/\delta \rceil + 1$ *and* $M_2 = \lceil 2SK^2/\delta \rceil + 1$*. Then there exists* $\{(a_j, w_j, b_j)\}_{j\in[M_1]}$ *and* $\{(\bar{a}_j, \bar{w}_j, \bar{b}_j)\}_{j\in[M_2]}$ *with*

$$\sup_j |a_j| \le 2SK, \quad \sup_j |w_j| \le 1, \quad \sup_j |b_j| \le SK, \quad \sup_j |\bar{a}_j| \le 2, \quad \sup_j |\bar{w}_j| \le 1, \quad \sup_j |\bar{b}_j| \le \log(4SK^2/\delta),$$

such that defining

$$\log_{\delta\star}(x) = \textstyle\sum_{j=1}^{M_1} a_j \cdot \mathrm{ReLU}(w_j x + b_j), \quad \exp_{\delta\star}(x) = \sum_{j=1}^{M_2} \bar{a}_j \cdot \mathrm{ReLU}(\bar{w}_j x + \bar{b}_j),$$

and defining $f(h), f_\delta(h) \in \mathbb{R}^S$ *by*

$$f(h)_i = \log \textstyle\sum_{j=1}^{S} \Psi_{ij} \exp(h_j), \quad f_\delta(h)_i = \log_{\delta\star} \sum_{j=1}^{S} \Psi_{ij} \exp_{\delta\star}(h_j), \quad \forall i \in [S],$$

we have

$$\sup_{\max_i h_i = 0} \|f(h) - f_\delta(h)\|_\infty \le \delta.$$

Proof of Lemma 13. Lemma 9 implies that taking $M_1 = \lceil 2SK/\delta \rceil + 1$, there exists $\{(a_j, w_j, b_j)\}_{j\in[M_1]}$ with

$$\sup_j |a_j| \le 2SK, \quad \sup_j |w_j| \le 1, \quad \sup_j |b_j| \le SK,$$

such that defining

$$\log_{\delta\star}(x) = \textstyle\sum_{j=1}^{M_1} a_j \cdot \mathrm{ReLU}(w_j x + b_j),$$

we have

$$\sup_{1/(SK) \le x \le SK} |\log(x) - \log_{\delta\star}(x)| \le \delta/2.$$

Lemma 8 implies that taking $M_2 = \lceil 2SK^2/\delta \rceil + 1$, there exists $\{(\bar{a}_j, \bar{w}_j, \bar{b}_j)\}_{j\in[M_2]}$ with

$$\sup_j |\bar{a}_j| \le 2, \quad \sup_j |\bar{w}_j| \le 1, \quad \sup_j |\bar{b}_j| \le \log(4SK^2/\delta),$$

such that defining

$$\exp_{\delta\star}(x) = \textstyle\sum_{j=1}^{M_2} \bar{a}_j \cdot \mathrm{ReLU}(\bar{w}_j x + \bar{b}_j),$$

we have $\exp_{\delta\star}$ is non-decreasing on $(-\infty, 0]$, and

$$\sup_{x\in(-\infty,0]} |\exp(x) - \exp_{\delta\star}(x)| \le \delta/(2SK^2), \quad \exp_{\delta\star}(0) = 1.$$

As a consequence, the condition of Lemma 11 is satisfied with $\delta_1 = \delta/2$ and $\delta_2 = \delta/(2SK^2)$, so that we have

$$\sup_{\max_i h_i = 0} \|f(h) - f_\delta(h)\|_\infty \le \delta_1 + SK^2\delta_2 = \delta.$$

This finishes the proof of Lemma 13. □

Lemma 14 (Existence of ReLU network approximating log-sum-exponential)**.** *Let* f_δ *be the function as defined in Lemma 13. Then there exists a two-hidden-layer neural network*

$$\mathrm{NN}_{W_1,W_2,W_3}(h) = W_1 \cdot \mathrm{ReLU}(W_2 \cdot \mathrm{ReLU}(W_3 \cdot [h; 1])),$$

with $W_1 \in \mathbb{R}^{S\times SM_1}$, $W_2 \in \mathbb{R}^{SM_1\times(SM_2+1)}$, $W_3 \in \mathbb{R}^{(SM_2+1)\times(S+1)}$*, and*

$$\|W_1\|_{\max} \le 2SK, \quad \|W_2\|_{\max} \le \mathrm{Poly}(SKM_1M_2), \quad \|W_3\|_{\max} \le \log(4SK^2/\delta).$$

such that

$$\mathrm{NN}_{W_1,W_2,W_3}(h) = f_\delta(h), \quad \forall h \in \mathbb{R}^S \text{ such that } \max_j h_j = 0.$$

Proof of Lemma 14. Define

$$\overline{W}_2 = \begin{bmatrix} \bar{a}_{1:M_2}^{\mathbb{T}} & 0 & \cdots & 0 & 0 \\ 0 & \bar{a}_{1:M_2}^{\mathbb{T}} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \bar{a}_{1:M_2}^{\mathbb{T}} & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \in \mathbb{R}^{(S+1)\times(SM_2+1)},$$

$$W_3 = \begin{bmatrix} \bar{w}_{1:M_2} & 0 & \cdots & 0 & \bar{b}_{1:M_2} \\ 0 & \bar{w}_{1:M_2} & \cdots & 0 & \bar{b}_{1:M_2} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \bar{w}_{1:M_2} & \bar{b}_{1:M_2} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \in \mathbb{R}^{(SM_2+1)\times(S+1)},$$

then we have

$$[\exp_{\delta\star}(h); 1] = \overline{W}_2 \cdot \mathrm{ReLU}(W_3 \cdot [h; 1]).$$

Define

$$W_1 = \begin{bmatrix} a_{1:M_1}^{\mathbb{T}} & 0 & \cdots & 0 \\ 0 & a_{1:M_1}^{\mathbb{T}} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & a_{1:M_1}^{\mathbb{T}} \end{bmatrix} \in \mathbb{R}^{S\times SM_1},$$

$$\tilde{W}_2 = \begin{bmatrix} w_{1:M_1} & 0 & \cdots & 0 & b_{1:M_1} \\ 0 & w_{1:M_1} & \cdots & 0 & b_{1:M_1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & w_{1:M_1} & \bar{b}_{1:M_1} \end{bmatrix} \in \mathbb{R}^{SM_1\times(S+1)},$$

then we have

$$\log_{\delta\star}(h) = W_1 \cdot \mathrm{ReLU}(\tilde{W}_2 \cdot [h; 1]).$$

As a consequence, we have

$$f_\star(h) = \log_{\delta\star} \Psi \exp_{\delta\star}(h) = W_1 \cdot \mathrm{ReLU}(\tilde{W}_2 \cdot \mathrm{diag}(\Psi, 1) \cdot \overline{W}_2 \cdot \mathrm{ReLU}(W_3 \cdot [h; 1])) = W_1 \cdot \mathrm{ReLU}(W_2 \cdot \mathrm{ReLU}(W_3 \cdot [h; 1])),$$

where we define $W_2 = \tilde{W}_2 \cdot \mathrm{diag}(\Psi, 1) \cdot \overline{W}_2$. It is also direct to upper bound $\|W_1\|_{\max}$, $\|W_2\|_{\max}$, and $\|W_3\|_{\max}$. This finishes the proof of Lemma 14. □

Lemma 15 (ReLU approximation of log-Psi)**.** *Assume that* $1/K \le \min_{ij} \Psi_{ij} \le \max_{ij} \Psi_{ij} \le K$. *For any* $\delta > 0$*, take* $M_1 = \lceil K/\delta \rceil + 1$ *and* $M_2 = 3$*. Then there exists* $\{(a_j, w_j, b_j)\}_{j\in[M_1]}$ *and* $\{(\bar{a}_j, \bar{w}_j, \bar{b}_j)\}_{j\in[M_2]}$ *with*

$$\sup_j |a_j| \le 2K, \quad \sup_j |w_j| \le 1, \quad \sup_j |b_j| \le SK, \quad \sup_j |\bar{a}_j| \le 4, \quad \sup_j |\bar{w}_j| \le 1, \quad \sup_j |\bar{b}_j| \le 1,$$

such that defining

$$\log_\delta(x) = \textstyle\sum_{j=1}^{M_1} a_j \cdot \mathrm{ReLU}(w_j x + b_j), \quad \mathrm{Ind}(x) = \textstyle\sum_{j=1}^{M_2} \bar{a}_j \cdot \mathrm{ReLU}(w_j x + b_j),$$

and defining $f(x), f_\delta(x) \in \mathbb{R}^S$ *by*

$$f(x)_i = \log \textstyle\sum_{j=1}^{S} \Psi_{ij} 1(x = j), \quad f_\delta(x)_i = \log_\delta \textstyle\sum_{j=1}^{S} \Psi_{ij} \mathrm{Ind}(x - j), \quad \forall i \in [S],$$

we have

$$\sup_{x\in[S]} \|f(x) - f_\delta(x)\|_\infty \le \delta.$$

Proof of Lemma 15. The proof of the lemma is similar to the proof of Lemma 13, using a combination of Lemma 10, 9, and 12. □

Lemma 16 (Existence of ReLU network approximating log-Psi)**.** *Let* f_δ *be the function as defined in Lemma 15. Then there exists a two-hidden-layer neural network*

$$\mathrm{NN}_{W_1,W_2,W_3}(x) = W_1 \cdot \mathrm{ReLU}(W_2 \cdot \mathrm{ReLU}(W_3 \cdot [x; 1])),$$

with $W_1 \in \mathbb{R}^{S\times SM_1}$, $W_2 \in \mathbb{R}^{SM_1\times(SM_2+1)}$, $W_3 \in \mathbb{R}^{(SM_2+1)\times(S+1)}$, *and*

$$\|W_1\|_{\max} \le 2K, \quad \|W_2\|_{\max} \le \mathrm{Poly}(SKM_1M_2), \quad \|W_3\|_{\max} \le 1.$$

such that

$$\mathrm{NN}_{W_1,W_2,W_3}(x) = f_\delta(x), \quad \forall x \in [S].$$

Proof of Lemma 16. The proof of Lemma 16 is similar to the proof of Lemma 14. □

E PROOF OF THEOREM 1

Proof of Theorem 1. By Lemma 17, we have the error decomposition

$$\mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \mu_\star) \le \inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\boldsymbol{W}}, \mu_\star) + 2\cdot \sup_{\boldsymbol{W}\in\mathcal{W}} \Big|\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\Big|.$$

To control the first term (the approximation error), by Theorem 4, there exists $\boldsymbol{W} \in \mathcal{W}_{d,\underline{m},L,S,D,B}$ as in Eq. (9) with norm bound $B = \mathrm{Poly}(d, S, K, 3^L, D)$, such that defining $\mu_{\mathrm{NN}}^{\boldsymbol{W}}$ as in Eq. (ConvNet), we have

$$\max_{y\in[S],\boldsymbol{x}\in[S]^d} \Big|\log \mu_\star(y|\boldsymbol{x}) - \log \mu_{\mathrm{NN}}^{\boldsymbol{W}}(y|\boldsymbol{x})\Big| \le C\cdot \frac{S^2K^2d\cdot 3^L}{D}.$$

Furthermore, by Lemma 4, when $D \ge S^2K^2d\cdot 3^L$, we have

$$\inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\boldsymbol{W}}, \mu_\star) \le C\Big(e^{\frac{S^2K^2d\cdot 3^L}{D}} - 1\Big)^2 \le C\frac{S^4K^4d^2\cdot 3^{2L}}{D^2}.$$

To control the second term (the generalization error), by Proposition 7, with probability at least $1-\eta$, we have

$$\sup_{\boldsymbol{W}\in\mathcal{W}_{d,\underline{m},L,S,D,B}} \Big|\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\Big| \le C\cdot\sqrt{\frac{LD(D+2S+1)\|\underline{m}\|_1\log(d\|\underline{m}\|_1 DSB\cdot 3^L) + \log(1/\eta)}{n}}.$$

Combining the above two equations proves Theorem 1. □

E.1 ERROR DECOMPOSITION

Lemma 17. *Consider the setting of Theorem 1. We have decomposition*

$$\mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \mu_\star) \le \inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\boldsymbol{W}}, \mu_\star) + 2\cdot \sup_{\boldsymbol{W}\in\mathcal{W}} \Big|\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\Big|.$$

Proof of Lemma 17. We have that for any conditional distribution $\mu_1(\cdot|\cdot)$, there is decomposition

$$\mathsf{D}_2^2(\mu_1,\mu_\star) = \mathbb{E}_{\boldsymbol{x}\sim\mu_\star}\Big[\sum_{s=1}^S \Big(\mu_1(s|\boldsymbol{x}) - \mu_\star(s|\boldsymbol{x})\Big)^2\Big]$$

$$= \mathbb{E}_{(\boldsymbol{x},y)\sim\mu_\star}\Big[\sum_{s=1}^S (\mu_1(s|\boldsymbol{x}) - 1\{y=s\})^2\Big] - \mathbb{E}_{(\boldsymbol{x},y)\sim\mu_\star}\Big[\sum_{s=1}^S (\mu_\star(s|\boldsymbol{x}) - 1\{y=s\})^2\Big] = \mathsf{R}(\mu_1) - \mathsf{R}(\mu_\star).$$

Define

$$\boldsymbol{W}_\star = \arg\min_{\boldsymbol{W}\in\mathcal{W}} \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) = \arg\min_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\boldsymbol{W}}, \mu_\star).$$

Then we have

$$\begin{aligned}
&\mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \mu_\star) = \mathsf{R}(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) - \mathsf{R}(\mu_\star)\\
&= \mathsf{R}(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) - \widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) + \widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) - \widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}_\star}) + \widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}_\star}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}_\star}) + \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}_\star}) - \mathsf{R}(\mu_\star)\\
&\le 2\cdot \sup_{\boldsymbol{W}\in\mathcal{W}} \Big|\mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\Big| + \mathsf{D}_2^2(\mu_{\mathrm{NN}}^{\boldsymbol{W}_\star}, \mu_\star)
\end{aligned}$$

This proves Lemma 17. □

E.2 RESULTS ON GENERALIZATION

Proposition 7 (Generalization error of the classification problem)**.** *Let $\mathcal{W}_{d,\underline{m},L,S,D,B}$ be the set defined as in Eq. (9). Then, with probability at least $1-\eta$, we have*

$$\sup_{\boldsymbol{W}\in\mathcal{W}_{d,\underline{m},L,S,D,B}} \Big|\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\Big| \le C\cdot\sqrt{\frac{LD(D+2S+1)\|\underline{m}\|_1\log(d\|\underline{m}\|_1 DSB\cdot 3^L) + \log(1/\eta)}{n}}.$$

Proof of Proposition 7. In Lemma 3, we can take $z = (y, \boldsymbol{x})$, $w = \boldsymbol{W}$, $\Theta = \mathcal{W}_{d,\underline{m},L,S,D,B}$, $\rho(w, w') = \|\boldsymbol{W} - \boldsymbol{W}'\|$, and $f(z_i; w) = \mathrm{loss}(y, \mu_{\mathrm{NN}}^{\boldsymbol{W}}(\cdot|\boldsymbol{x}))$. Therefore, to show Proposition 7, we just need to apply Lemma 3 by checking (a), (b), (c).

Check (a). We note that the index set $\Theta := \mathcal{W}_{d,\underline{m},L,S,D,B}$ equipped with $\rho(w, w') := \|\boldsymbol{W} - \boldsymbol{W}'\|$ has diameter $B_{\mathrm{p}} := 2B$. Further note that $\mathcal{W}_{d,\underline{m},L,S,D,B}$ has a dimension bounded by $d_{\mathrm{p}} := D(D + 2S + 1)\|\underline{m}\|_1$. According to Example 5.8 of (Wainwright, 2019), it holds that $\log N(\Delta; \mathcal{W}_{d,\underline{m},L,S,D,B}, \|\cdot\|) \le d_{\mathrm{p}} \cdot \log(1 + 2r/\Delta)$ for any $0 < \Delta \le 2r$. This verifies (a).

Check (b). Since $f(z_i; w) = \mathrm{loss}(y, \mu_{\mathrm{NN}}^{\boldsymbol{W}}(\cdot|\boldsymbol{x}))$ is 2-bounded. As a consequence, $f(z, w) - \mathbb{E}_z[f(z, w)]$ is a sub-Gaussian random variable with the sub-Gaussian parameter to be a universal constant.

Check (c). Lemma 20 implies that

$$|f(z; w_1) - f(z; w_2)| \le \sigma' \cdot \|\boldsymbol{W}_1 - \boldsymbol{W}_2\|, \quad \sigma' := 12\|\underline{m}\|_1(3B^3)^L \cdot d \cdot S^{3/2} \cdot (S + D).$$

As a consequence, $f(z; w_1) - f(z; w_2)$ is $\sigma'\rho(w_1, w_2) = \sigma'\|\boldsymbol{W}_1 - \boldsymbol{W}_2\|$ sub-Gaussian.

Therefore, we apply Lemma 3 to conclude the proof of Proposition 7. □

E.3 Auxillary lemmas

Lemma 18 (Norm bound in the chain rule in classification settings)**.** *Consider the ConvNet as in Eq. (ConvNet). Assume that $\|\boldsymbol{W}\| \le B$. Then for any ℓ, v, ι, and $\star \in [3]$, we have*

$$\begin{aligned}
\|\mathsf{h}_v^{(L)}\|_2 &\le S^{3/2}, \\
\|\mathsf{q}_v^{(\ell)}\|_2 &\le B^3 \cdot (\|\mathsf{h}_v^{(\ell)}\|_2 + 1), \\
\|\mathsf{h}_v^{(\ell-1)}\|_2 &\le 2m^{(\ell)} \cdot \max_{v' \in \mathcal{C}(v)} \|\mathsf{q}_{v'}^{(\ell)}\|_2, \\
\max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(k)}} \mathsf{q}_{v,i}^{(\ell)}\|_{\mathrm{op}} &\le B^3 \cdot \max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(k)}} \mathsf{h}_{v,i}^{(\ell)}\|_{\mathrm{op}}, && \forall k \ge \ell + 1, \\
\max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(\ell)}} \mathsf{q}_{v,i}^{(\ell)}\|_{\mathrm{op}} &\le B^2 \cdot \|\mathsf{h}_v^{(\ell)}\|_2, \\
\max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(k)}} \mathsf{h}_{v,i}^{(\ell-1)}\|_{\mathrm{op}} &\le 2m^{(\ell)} \cdot \max_{v' \in \mathcal{C}(v)} \max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(k)}} \mathsf{q}_{v',i}^{(\ell)}\|_{\mathrm{op}}, && \forall k \ge \ell, \\
\max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(\ell)}} \mathrm{softmax}(\mathsf{h}_{\mathrm{r}}^{(0)})_i\|_{\mathrm{op}} &\le \max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(\ell)}} \mathsf{h}_{\mathrm{r},i}^{(0)}\|_{\mathrm{op}}.
\end{aligned}$$

Proof of Lemma 18. The proof of the lemma uses the chain rule, the 1-Lipschitzness of ReLU, the 2-Lipschitzness of normalize, and the 1-Lipschitzness of softmax. □

Lemma 19. *Consider the ConvNet as in Eq. (ConvNet). Assume that $\|\boldsymbol{W}\| \le B$. Then we have*

$$\max_{\boldsymbol{x}\in[S]^d} \max_{\star\in[3]} \max_{\ell\in[L]} \max_{\iota\in[m^{(\ell)}]} \max_{i\in[S]} \|\nabla_{W_{\star,\iota}^{(\ell)}} \mathrm{softmax}(\mathsf{h}_{\mathrm{r}}^{(0)})_i\|_{\mathrm{op}} \le (3B^3)^L \cdot d \cdot S^{3/2}.$$

Proof of Lemma 19. This lemma is implied by Lemma 18 and an induction argument. □

Lemma 20. *Consider the ConvNet as in Eq. (ConvNet). Assume that $\|\boldsymbol{W}\| \le B$. Then we have*

$$\max_{y,\boldsymbol{x}} \left|\mu_{\mathrm{NN}}^{\boldsymbol{W}}(y|\boldsymbol{x}) - \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(y|\boldsymbol{x})\right| \le 3\|\underline{m}\|_1(3B^3)^L \cdot d \cdot S^{3/2} \cdot (S + D) \cdot \|\boldsymbol{W} - \overline{\boldsymbol{W}}\|.$$

Therefore, we have

$$\left|\mathrm{loss}(y, \mu_{\mathrm{NN}}^{\boldsymbol{W}}(\cdot|\boldsymbol{x})) - \mathrm{loss}(y, \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(\cdot|\boldsymbol{x}))\right| \le 12\|\underline{m}\|_1(3B^3)^L \cdot d \cdot S^{3/2} \cdot (S + D) \cdot \|\boldsymbol{W} - \overline{\boldsymbol{W}}\|.$$

Proof of Lemma 20. The first inequality is by the fact that

$$\begin{aligned}
&\max_{y,\boldsymbol{x}} \left|\mu_{\mathrm{NN}}^{\boldsymbol{W}}(y|\boldsymbol{x}) - \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(y|\boldsymbol{x})\right| \\
&\le \sum_{\star\in[3]} \sum_{\ell\in[L]} \sum_{\iota\in[m^{(\ell)}]} \min\{\mathrm{nrow}(W_{\star,\iota}^{(\ell)}), \mathrm{ncol}(W_{\star,\iota}^{(\ell)})\} \|\nabla_{W_{\star,\iota}^{(\ell)}} \mathrm{softmax}(\mathsf{h}_{\mathrm{r}}^{(0)})_i\|_{\mathrm{op}} \|W_{\star,\iota}^{(\ell)} - \overline{W}_{\star,\iota}^{(\ell)}\|_{\mathrm{op}},
\end{aligned}$$

where we have used the inequality that $\mathrm{trace}(A^{\mathbb{T}}B) \le \{\mathrm{nrow}(A), \mathrm{ncol}(A)\}\|A\|_{\mathrm{op}}\|B\|_{\mathrm{op}}$.

To prove the second equation, we have

$$\Big|\mathrm{loss}(y, \mu_{\mathrm{NN}}^{\boldsymbol{W}}(\cdot|\boldsymbol{x})) - \mathrm{loss}(y, \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(\cdot|\boldsymbol{x}))\Big| = \Big|\sum_{s=1}^{S}\Big(1\{y=s\} - \mu_{\mathrm{NN}}^{\boldsymbol{W}}(s|\boldsymbol{x})\Big)^2 - \sum_{s=1}^{S}\Big(1\{y=s\} - \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(s|\boldsymbol{x})\Big)^2\Big|$$

$$\le \sum_{s=1}^{S}\Big|1\{y=s\} - \mu_{\mathrm{NN}}^{\boldsymbol{W}}(s|\boldsymbol{x}) + 1\{y=s\} - \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(s|\boldsymbol{x})\Big| \cdot \Big|\mu_{\mathrm{NN}}^{\boldsymbol{W}}(s|\boldsymbol{x}) - \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(s|\boldsymbol{x})\Big| \le 4 \cdot \max_{s}\Big|\mu_{\mathrm{NN}}^{\boldsymbol{W}}(s|\boldsymbol{x}) - \mu_{\mathrm{NN}}^{\overline{\boldsymbol{W}}}(s|\boldsymbol{x})\Big|.$$

This completes the proof of Lemma 20. □

F Proof of Proposition 5

Proof of Proposition 5. By Eq. (MP-DNS) and (23), defining $\nu_{\downarrow,v}^{(\ell)}(\cdot) = \mathrm{softmax}(h_{\downarrow,v}^{(\ell)})$, then for $\ell \le L-1$, we get

$$\nu_{\downarrow,v}^{(\ell)}(x_v^{(\ell)}) \propto \prod_{v'\in\mathcal{C}(v)}\Big(\sum_{a\in[S]}\psi_{\iota(v')}^{(\ell+1)}(x_v^{(\ell)}, a)e^{(h_{\downarrow,v}^{(\ell+1)})_a}\Big)$$
$$\propto \sum_{x_{\mathcal{C}(v)}^{(\ell+1)}}\prod_{v'\in\mathcal{C}(v)}\Big(\psi_{\iota(v')}^{(\ell+1)}(x_v^{(\ell)}, x_{v'}^{(\ell+1)})\nu_{\downarrow,v'}^{(\ell+1)}(x_{v'}^{(\ell+1)})\Big).$$

This coincides with the update rule of $\nu_{\downarrow,v}^{(\ell)}$ as in Eq. (BP-DNS).

Furthermore, defining $\nu_{\uparrow,v}^{(\ell)}(\cdot) = \mathrm{softmax}(b_{\uparrow,v}^{(\ell)} - h_{\downarrow,v}^{(\ell)})$, then for $\ell = 0, 1, \ldots, L$, we get

$$\nu_{\uparrow,v}^{(\ell)}(x_v^{(\ell)}) \propto \sum_{b\in[S]}\psi_{\iota(v)}^{(\ell+1)}(b, x_v^{(\ell)})\nu_{\uparrow,\mathrm{pa}(v)}^{(\ell+1)}(b)\prod_{v'\in\mathcal{N}(v)}\Big(\sum_{a\in[S]}\psi_{\iota(v')}^{(\ell+1)}(b, a)e^{(h_{\downarrow,v}^{(\ell+1)})_a}\Big)$$
$$\propto \sum_{x_{\mathrm{pa}(v)}^{(\ell-1)}, x_{\mathcal{N}(v)}^{(\ell)}}\psi^{(\ell)}(x_{\mathrm{pa}(v)}^{(\ell-1)}, x_{\mathcal{C}(\mathrm{pa}(v))}^{(\ell)})\nu_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)}(x_{\mathrm{pa}(v)}^{(\ell-1)})\prod_{v'\in\mathcal{N}(v)}\nu_{\downarrow,v'}^{(\ell)}(x_{v'}^{(\ell)}).$$

This coincides with the update rule of $\nu_{\uparrow,v}^{(\ell)}$ as in Eq. (BP-DNS).

Finally, defining $\nu_v^{(L)}(\cdot) = \mathrm{softmax}(b_{\uparrow,v}^{(L)})$, we get

$$\nu_v^{(L)}(x_v^{(L)}) \propto \sum_{b\in[S]}\psi_{\iota(v)}^{(L)}(b, x_v^{(L)})\nu_{\uparrow,\mathrm{pa}(v)}^{(L)}(b)\prod_{v'\in\mathcal{N}(v)}\Big(\sum_{a\in[S]}\psi_{\iota(v')}^{(L)}(b, a)e^{(h_{\downarrow,v}^{(L)})_a}\Big) \times \nu_{\downarrow,v}^{(L)}(x_v^{(L)})$$
$$\propto \nu_{\uparrow,v}^{(L)}(x_v^{(L)})\psi^{(L+1)}(x_v^{(L)}, z_v).$$

This coincides with the formula of $\nu_v^{(L)}$ as in Eq. (BP-DNS).

This finishes the proof of Proposition 5. □

G Proof of Theorem 6

Proof of Theorem 6. By Lemma 13, take $M_1 = \lceil 2S^2Kd18^L/\delta\rceil + 1$ and $M_2 = \lceil 2S^2K^2d18^L/\delta\rceil + 1$. Then there exists $\{(a_j, w_j, b_j)\}_{j\in[M_1]}$ and $\{(\bar{a}_j, \bar{w}_j, \bar{b}_j)\}_{j\in[M_2]}$ with

$$\sup_j|a_j| \le 2SK, \quad \sup_j|w_j| \le 1, \quad \sup_j|b_j| \le SK, \quad \sup_j|\bar{a}_j| \le 2, \quad \sup_j|\bar{w}_j| \le 1, \quad \sup_j|\bar{b}_j| \le \log(4{\cdot}18^LS^2dK^2/\delta),$$

such that defining

$$\log_{\delta\star}(x) = \textstyle\sum_{j=1}^{M_1} a_j \cdot \mathrm{ReLU}(w_jx + b_j), \quad \exp_{\delta\star}(x) = \textstyle\sum_{j=1}^{M_2}\bar{a}_j \cdot \mathrm{ReLU}(\bar{w}_jx + \bar{b}_j),$$

and defining $f^{(\ell)}_{\diamond,\iota}(h), \bar{f}^{(\ell)}_{\diamond,\iota}(h) \in \mathbb{R}^S$ for $\diamond \in \{\downarrow, \uparrow\}$ by

$$f^{(\ell)}_{\downarrow,\iota}(h)_i = \log \sum_{j=1}^S \psi^{(\ell)}_\iota(i,j)\exp(h_j), \quad \bar{f}^{(\ell)}_{\downarrow,\iota}(h)_i = \log_{\delta\star} \sum_{j=1}^S \psi^{(\ell)}_\iota(i,j)\exp_{\delta\star}(h_j), \qquad \forall i \in [S],$$
$$f^{(\ell)}_{\uparrow,\iota}(h)_i = \log \sum_{j=1}^S \psi^{(\ell)}_\iota(j,i)\exp(h_j), \quad \bar{f}^{(\ell)}_{\uparrow,\iota}(h)_i = \log_{\delta\star} \sum_{j=1}^S \psi^{(\ell)}_\iota(j,i)\exp_{\delta\star}(h_j), \qquad \forall i \in [S],$$

we have

$$\sup_{\max_i h_i = 0} \|f^{(\ell)}_{\diamond,\iota}(h) - \bar{f}^{(\ell)}_{\diamond,\iota}(h)\|_\infty \le \delta/(d18^L S). \tag{30}$$

By Eq. (30) and Lemma 21, taking $b^{(L)}_{\uparrow,v} \in \mathbb{R}^S$ to be as defined in Eq. (MP-DNS) and $\bar{b}^{(L)}_{\uparrow,v} \in \mathbb{R}^S$ to be as defined in Eq. (A-MP-DNS) with $\{\bar{f}^{(\ell)}_{\diamond,\iota}\}_{\ell\in[L],\iota\in[m^{(\ell)}]}$ as defined above, we have

$$\|b^{(L)}_{\uparrow,v} - \bar{b}^{(L)}_{\uparrow,v}\|_\infty \le [\delta/(d18^L S)] \times 18^L d = \delta/S,$$

which gives

$$\sup_{\boldsymbol{z}\in\mathbb{R}^d} \|\boldsymbol{m}(\boldsymbol{z}) - \boldsymbol{m}_{\mathrm{NN}}(\boldsymbol{z})\|_\infty \le S \cdot \|b^{(L)}_{\uparrow,v} - \bar{b}^{(L)}_{\uparrow,v}\|_\infty \le \delta.$$

As a consequence, we just need to show that the approximate version of message passing algorithm as in Eq. (A-MP-DNS) could be cast as a neural network.

Indeed, by Lemma 14, there exist two-hidden-layer neural networks (for $\diamond \in \{\downarrow,\uparrow\}$, $\ell \in [L]$, and $\iota \in [m^{(\ell)}]$)

$$\mathrm{NN}_{W^{(\ell)}_{1,\diamond,\iota}, W^{(\ell)}_{2,\diamond,\iota}, W^{(\ell)}_{3,\diamond,\iota}}(h) = W^{(\ell)}_{1,\diamond,\iota} \cdot \mathrm{ReLU}(W^{(\ell)}_{2,\diamond,\iota} \cdot \mathrm{ReLU}(W^{(\ell)}_{3,\diamond,\iota} \cdot [h;1])),$$

with $W^{(\ell)}_{1,\diamond,\iota} \in \mathbb{R}^{S\times SM_1}$, $W^{(\ell)}_{2,\diamond,\iota} \in \mathbb{R}^{SM_1\times(SM_2+1)}$, $W^{(\ell)}_{3,\diamond,\iota} \in \mathbb{R}^{(SM_2+1)\times(S+1)}$, and

$$\|W^{(\ell)}_{1,\diamond,\iota}\|_{\max} \le 2SK, \quad \|W^{(\ell)}_{2,\diamond,\iota}\|_{\max} \le \mathrm{Poly}(SKM_1M_2), \quad \|W^{(\ell)}_{3,\diamond,\iota}\|_{\max} \le \log(4S^2d18^LK^2/\delta).$$

such that

$$\mathrm{NN}_{W^{(\ell)}_{1,\diamond,\iota}, W^{(\ell)}_{2,\diamond,\iota}, W^{(\ell)}_{3,\diamond,\iota}}(h) = \bar{f}^{(\ell)}_{\diamond,\iota}(h), \quad \forall h \in \mathbb{R}^S \text{ such that } \max_j h_j = 0.$$

This proves that the approximate version of message passing as in Eq. (A-MP-DNS) coincides with the U-Net as in Eq. (UNet) with proper choice of dimension

$$D \ge \max_{\ell\in[L]}\{SM^{(\ell)}_1, SM^{(\ell)}_2 + 1\} = S \times \left(\lceil 2S^2K^2d18^L/\delta\rceil + 1\right) + 1$$

and norm of the weights. This finishes the proof of Theorem 6. □

G.1 Auxillary lemmas

Lemma 21 (Error propagation of the approximate version of message passing in denoising)**.** *Assume we have functions $\{f^{(\ell)}_{\downarrow,\iota}, f^{(\ell)}_{\uparrow,\iota}\}$ and $\{\bar{f}^{(\ell)}_{\downarrow,\iota}, \bar{f}^{(\ell)}_{\uparrow,\iota}\}$ such that*

$$\|f^{(\ell)}_{\diamond,\iota}(h) - \bar{f}^{(\ell)}_{\diamond,\iota}(h)\|_\infty \le \delta, \quad \forall h \in \mathbb{R}^S \text{ such that } \max_{j\in[S]} h_j = 0, \ \diamond \in \{\downarrow,\uparrow\}. \tag{31}$$

Furthermore, consider the following approximate version of message passing algorithm with initialization $h^{(L)}_{\downarrow,v} = (-(x-z_v)^2/2)_{x\in[S]} \in \mathbb{R}^S$ for $v \in \mathcal{V}^{(L)}$, defined as below

$$\begin{aligned}
\bar{q}^{(\ell)}_{\downarrow,v} &= \bar{f}^{(\ell)}_{\downarrow,\iota(v)}(\mathrm{normalize}(\bar{h}^{(\ell)}_{\downarrow,v})) \in \mathbb{R}^S, && \ell\in[L], \quad v\in\mathcal{V}^{(\ell)},\\
\bar{h}^{(\ell-1)}_{\downarrow,v} &= \textstyle\sum_{v'\in\mathcal{C}(v)} \bar{q}^{(\ell)}_{\downarrow,v'} \in \mathbb{R}^S, && \ell\in[L], \quad v\in\mathcal{V}^{(\ell-1)},\\
\bar{u}^{(\ell)}_{\uparrow,v} &= \bar{b}^{(\ell-1)}_{\uparrow,\mathrm{pa}(v)} \in \mathbb{R}^S, \quad (\textit{with } \bar{b}^{(0)}_{\uparrow,\mathrm{r}} = \bar{h}^{(0)}_{\downarrow,\mathrm{r}}) && \ell\in[L], \quad v\in\mathcal{V}^{(\ell)},\\
\bar{b}^{(\ell)}_{\uparrow,v} &= \bar{f}^{(\ell)}_{\uparrow,\iota(v)}(\mathrm{normalize}(\bar{u}^{(\ell)}_{\uparrow,v} - \bar{q}^{(\ell)}_{\downarrow,v})) + \bar{h}^{(\ell)}_{\downarrow,v} \in \mathbb{R}^S, && \ell\in[L], \quad v\in\mathcal{V}^{(\ell)},\\
\bar{\boldsymbol{m}}_{\mathrm{MP}}(\boldsymbol{z})_v &= \textstyle\sum_{s\in[S]} s\cdot\mathrm{softmax}(\bar{b}^{(L)}_{\uparrow,v})_s, && v\in\mathcal{V}^{(L)}.
\end{aligned} \tag{A-MP-DNS}$$

Taking $b^{(L)}_{\uparrow,v} \in \mathbb{R}^S$ to be as defined in Eq. (MP-DNS) and $\bar{b}^{(L)}_{\uparrow,v} \in \mathbb{R}^S$ to be as defined in Eq. (A-MP-DNS), we have

$$\max_{v\in\mathcal{V}^{(L)}} \|b^{(L)}_{\uparrow,v} - \bar{b}^{(L)}_{\uparrow,v}\|_\infty \le \delta \times 18^L \cdot d.$$

Proof of Lemma 21.

Step 1. Downward induction. In the first step, we aim to show that for any $\ell \in [L-1]$ we have

$$\|\bar{h}_{\downarrow,v}^{(\ell)} - h_{\downarrow,v}^{(\ell)}\|_\infty \le m^{(\ell+1)} \prod_{k=\ell+2}^{L} (2m^{(k)}+1)\delta, \quad \forall v \in \mathcal{V}^{(\ell)}. \tag{32}$$

To prove the formula for $\ell = L-1$, since $\mathrm{normalize}(h_{\downarrow,v}^{(L)}) = \mathrm{normalize}(\bar{h}_{\downarrow,v}^{(L)})$, by Eq. (31), we get

$$\|\bar{q}_{\downarrow,v}^{(L)} - q_{\downarrow,v}^{(L)}\|_\infty \le \delta, \quad \forall v \in \mathcal{V}^{(L)}.$$

Hence we get

$$\|h_{\downarrow,v}^{(L-1)} - \bar{h}_{\downarrow,v}^{(L-1)}\|_\infty = \left\| \textstyle\sum_{v' \in \mathcal{C}(v)} (\bar{q}_{\downarrow,v'}^{(L)} - q_{\downarrow,v'}^{(L)}) \right\|_\infty \le m^{(L)}\delta, \quad \forall v \in \mathcal{V}^{(L-1)}.$$

This proves the formula (32) for $\ell = L-1$.

Assuming that (32) holds at the layer ℓ, by the update formula, we have

$$\begin{aligned}
&\|\bar{q}_{\downarrow,v}^{(\ell)} - q_{\downarrow,v}^{(\ell)}\|_\infty = \|\bar{f}_{\downarrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{h}_{\downarrow,v}^{(\ell)})) - f_{\downarrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(h_{\downarrow,v}^{(\ell)}))\|_\infty \\
&\le \|\bar{f}_{\downarrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{h}_{\downarrow,v}^{(\ell)})) - f_{\downarrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{h}_{\downarrow,v}^{(\ell)}))\|_\infty + \|f_{\downarrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(h_{\downarrow,v}^{(\ell)})) - f_{\downarrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{h}_{\downarrow,v}^{(\ell)}))\|_\infty \\
&\le \delta + 2m^{(\ell+1)} \textstyle\prod_{k=\ell+2}^{L} (2m^{(k)}+1)\delta \le \prod_{k=\ell+1}^{L} (2m^{(k)}+1)\delta,
\end{aligned}$$

where the middle inequality is by the assumption of $f_{\downarrow,\iota}^{(\ell)}$ and by Lemma 6 and Lemma 7. Hence we get

$$\|\bar{h}_{\downarrow,v}^{(\ell-1)} - h_{\downarrow,v}^{(\ell-1)}\|_\infty \le m^{(\ell)} \textstyle\prod_{k=\ell+1}^{L} (2m^{(k)}+1)\delta, \quad \forall v \in \mathcal{V}^{(\ell)}.$$

This proves Eq. (32) by the induction argument.

Step 2. Upward induction. The downward induction argument proves that, for $\Gamma = \prod_{k=1}^{L}(2m^{(k)}+1)$, we have

$$\|\bar{q}_{\downarrow,v}^{(\ell)} - q_{\downarrow,v}^{(\ell)}\|_\infty, \|\bar{h}_{\downarrow,v}^{(\ell)} - h_{\downarrow,v}^{(\ell)}\|_\infty \le \Gamma\delta, \quad \forall \ell = 0,1,\dots,L, \ \ \forall v \in \mathcal{V}^{(\ell)}.$$

In this step, we aim to show that for any $\ell = 0,1,\dots,L$, we have

$$\|\bar{b}_{\uparrow,v}^{(\ell)} - b_{\uparrow,v}^{(\ell)}\|_\infty \le 6^\ell \cdot \Gamma \cdot \delta, \quad \forall v \in \mathcal{V}^{(\ell)}. \tag{33}$$

To prove this formula for $\ell = 0$, note that $b_{\uparrow,\mathrm{r}}^{(0)} = h_{\downarrow,\mathrm{r}}^{(0)}$ and $\bar{b}_{\uparrow,\mathrm{r}}^{(0)} = \bar{h}_{\downarrow,\mathrm{r}}^{(0)}$, we have

$$\|\bar{b}_{\uparrow,\mathrm{r}}^{(0)} - b_{\uparrow,\mathrm{r}}^{(0)}\|_\infty = \|\bar{h}_{\downarrow,\mathrm{r}}^{(0)} - h_{\downarrow,\mathrm{r}}^{(0)}\|_\infty \le \Gamma \cdot \delta.$$

This proves the formula (33) for $\ell = 0$.

Assuming that (33) holds at layer $\ell - 1$, by the update formula, we have

$$\begin{aligned}
&\|b_{\uparrow,v}^{(\ell)} - \bar{b}_{\uparrow,v}^{(\ell)}\|_\infty \\
&\le \|f_{\uparrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(b_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - q_{\downarrow,v}^{(\ell)})) - \bar{f}_{\uparrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - \bar{q}_{\downarrow,v}^{(\ell)}))\|_\infty + \|h_{\downarrow,v}^{(\ell)} - \bar{h}_{\downarrow,v}^{(\ell)}\|_\infty \\
&\le \|f_{\uparrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(b_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - q_{\downarrow,v}^{(\ell)})) - f_{\uparrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - \bar{q}_{\downarrow,v}^{(\ell)}))\|_\infty \\
&\quad + \|f_{\uparrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - \bar{q}_{\downarrow,v}^{(\ell)})) - \bar{f}_{\uparrow,\iota(v)}^{(\ell)}(\mathrm{normalize}(\bar{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - \bar{q}_{\downarrow,v}^{(\ell)}))\|_\infty + \|h_{\downarrow,v}^{(\ell)} - \bar{h}_{\downarrow,v}^{(\ell)}\|_\infty \\
&\le \|\mathrm{normalize}(b_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - q_{\downarrow,v}^{(\ell)}) - \mathrm{normalize}(\bar{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)} - \bar{q}_{\downarrow,v}^{(\ell)})\|_\infty + \delta + \Gamma \cdot \delta \\
&\le 4 \cdot 6^{\ell-1} \cdot \Gamma \cdot \delta + \delta + \Gamma \cdot \delta \le 6^\ell \cdot \Gamma \cdot \delta.
\end{aligned}$$

This proves Eq. (33) by the induction argument. This proves the Lemma 21 by observing that $\Gamma \le 3^L \prod_{\ell=1}^{L} m^{(\ell)} = 3^L \cdot d$. □

H PROOF OF THEOREM 2

Proof of Theorem 2. By Lemma 22, we have the error decomposition

$$\mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \boldsymbol{m}) \le \inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}, \boldsymbol{m}) + 2\cdot \sup_{\boldsymbol{W}\in\mathcal{W}} \left|\widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}})\right|.$$

To control the first term (the approximation error), by Theorem 6, there exists $\boldsymbol{W} \in \mathcal{W}_{d,\underline{m},L,S,D,B}$ as in Eq. (18) with norm bound $B = \mathrm{Poly}(d, S, K, 18^L, D)$, such that defining $\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}$ as in Eq. (UNet), we have

$$\sup_{\boldsymbol{z}\in\mathbb{R}^d} \|\boldsymbol{m}(\boldsymbol{z}) - \boldsymbol{m}_{\mathrm{NN}}(\boldsymbol{z})\|_\infty \le C\cdot \frac{S^3K^2d\cdot 18^L}{D}.$$

Therefore, we have

$$\inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}, \boldsymbol{m}) \le \sup_{\boldsymbol{z}} \|\boldsymbol{m}(\boldsymbol{z}) - \boldsymbol{m}_{\mathrm{NN}}(\boldsymbol{z})\|_\infty^2 \le C\cdot \frac{S^6K^4d^2\cdot 18^{2L}}{D^2}.$$

To control the second term (the generalization error), by Proposition 8, with probability at least $1-\eta$, we have

$$\sup_{\boldsymbol{W}\in\mathcal{W}_{d,\underline{m},L,S,D,B}} \left|\widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}})\right| \le C\cdot S^2\cdot \sqrt{\frac{LD(D+2S+1)\|\underline{m}\|_1 \log(d\|\underline{m}\|_1 DSB\cdot 18^L) + \log(1/\eta)}{n}}.$$

Combining the above two equations proves Theorem 2. □

H.1 ERROR DECOMPOSITION

Lemma 22. *Consider the setting of Theorem 2. We have decomposition*

$$\mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \boldsymbol{m}) \le \inf_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}, \boldsymbol{m}) + 2\cdot \sup_{\boldsymbol{W}\in\mathcal{W}} \left|\widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}})\right|.$$

Proof of Lemma 22. We have that for any conditional expectation $\boldsymbol{m}_1(\boldsymbol{z})$, there is decomposition

$$\begin{aligned}
&\mathsf{D}_2^2(\boldsymbol{m}_1, \boldsymbol{m}) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{z})\sim\mu_\star}\left[d^{-1}\|\boldsymbol{m}_1(\boldsymbol{z}) - \boldsymbol{m}(\boldsymbol{z})\|_2^2\right] \\
&= \mathbb{E}_{(\boldsymbol{x},\boldsymbol{z})\sim\mu_\star}\left[d^{-1}\|\boldsymbol{m}_1(\boldsymbol{z}) - \boldsymbol{x}\|_2^2\right] - \mathbb{E}_{(\boldsymbol{x},\boldsymbol{z})\sim\mu_\star}\left[d^{-1}\|\boldsymbol{m}_1(\boldsymbol{z}) - \boldsymbol{x}\|_2^2\right] = \mathsf{R}(\boldsymbol{m}_1) - \mathsf{R}(\boldsymbol{m}).
\end{aligned}$$

Define

$$\boldsymbol{W}_\star = \arg\min_{\boldsymbol{W}\in\mathcal{W}} \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) = \arg\min_{\boldsymbol{W}\in\mathcal{W}} \mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}, \boldsymbol{m}).$$

Then we have

$$\begin{aligned}
&\mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}, \boldsymbol{m}) = \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) - \mathsf{R}(\boldsymbol{m}) \\
&= \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) - \widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) + \widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\widehat{\boldsymbol{W}}}) - \widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}_\star}) + \widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}_\star}) - \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}_\star}) + \mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}_\star}) - \mathsf{R}(\boldsymbol{m}) \\
&\le 2\cdot \sup_{\boldsymbol{W}\in\mathcal{W}} \left|\mathsf{R}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}) - \widehat{\mathsf{R}}(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}})\right| + \mathsf{D}_2^2(\boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}_\star}, \boldsymbol{m})
\end{aligned}$$

This proves Lemma 22. □

H.2 RESULTS ON GENERALIZATION

Proposition 8 (Generalization error of the denoising problem)**.** *Let $\mathcal{W}_{d,\underline{m},L,S,D,B}$ be the set defined as in Eq. (9). Then, with probability at least $1-\eta$, we have*

$$\sup_{\boldsymbol{W}\in\mathcal{W}_{d,\underline{m},L,S,D,B}} \left|\widehat{\mathsf{R}}(\mu_{\mathrm{NN}}^{\boldsymbol{W}}) - \mathsf{R}(\mu_{\mathrm{NN}}^{\boldsymbol{W}})\right| \le C\cdot S^2\cdot \sqrt{\frac{LD(D+2S+1)\|\underline{m}\|_1 \log(d\|\underline{m}\|_1 DSB\cdot 18^L) + \log(1/\eta)}{n}}.$$

Proof of Proposition 8. In Lemma 3, we can take $z = (\boldsymbol{z}, \boldsymbol{x})$, $w = \boldsymbol{W}$, $\Theta = \mathcal{W}_{d,\underline{m},L,S,D,B}$, $\rho(w, w') = \|\boldsymbol{W} - \boldsymbol{W}'\|$, and $f(z_i; w) = \|\boldsymbol{x} - \boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}(\boldsymbol{z})\|_2^2$. Therefore, to show Proposition 8, we just need to apply Lemma 3 by checking (a), (b), (c).

Check (a). We note that the index set $\Theta := \mathcal{W}_{d,\underline{m},L,S,D,B}$ equipped with $\rho(w, w') := \|\boldsymbol{W} - \boldsymbol{W}'\|$ has diameter $B_{\mathrm{p}} := 2B$. Further note that $\mathcal{W}_{d,\underline{m},L,S,D,B}$ has a dimension bounded by $d_{\mathrm{p}} := 2D(D + 2S + 1)\|\underline{m}\|_1$. According to Example 5.8 of (Wainwright, 2019), it holds that $\log N(\Delta; \mathcal{W}_{d,\underline{m},L,S,D,B}, \|\cdot\|) \le d_{\mathrm{p}} \cdot \log(1 + 2r/\Delta)$ for any $0 < \Delta \le 2r$. This verifies (a).

Check (b). Since $f(z_i; w) = d^{-1}\|\boldsymbol{x} - \boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}(\boldsymbol{z})\|_2^2$ is S^2-bounded. As a consequence, $f(z, w) - \mathbb{E}_z[f(z, w)]$ is a sub-Gaussian random variable with the sub-Gaussian parameter to be $C \cdot S^2$.

Check (c). Lemma 25 implies that

$$|f(z; w_1) - f(z; w_2)| \le L_{\mathrm{p}} \cdot \|\boldsymbol{W}_1 - \boldsymbol{W}_2\|, \quad L_{\mathrm{p}} := 12\|\underline{m}\|_1 18^L B^{6L} \cdot d \cdot S^3 \Big(\sum_{v \in \mathcal{V}^{(L)}} (S + |z_v|) \Big) \cdot (S + D).$$

Since $\boldsymbol{z} \overset{d}{=} \boldsymbol{x} + \boldsymbol{g}$ where $(\boldsymbol{x}, \boldsymbol{g}) \sim \mu_\star \times \mathcal{N}(\boldsymbol{0}, \mathbf{I}_d)$, and $\|\boldsymbol{x}\|_1 \le Sd$, $\|\boldsymbol{g}\|_1$ is Cd-sub-Gaussian. Hence $\|\boldsymbol{z}\|_1$ is CSd-sub-Gausssian, and hence $f(z; w_1) - f(z; w_2)$ is $\sigma' \rho(w_1, w_2)$ sub-Gaussian with

$$\sigma' = C\|\underline{m}\|_1 18^L B^{6L} \cdot d^2 \cdot S^4 \cdot (S + D).$$

Therefore, we apply Lemma 3 to conclude the proof of Proposition 8. □

H.3 Auxillary lemmas

Lemma 23 (Norm bound in the chain rule in denoising settings)**.** *Consider the U-Net as in Eq. (UNet) with modified input* $\mathsf{h}_{\downarrow,v}^{(L)} = (-x^2/2 + xz_v)_{x \in [S]} \in \mathbb{R}^S$ *(Since we will immediately normalize the input, this input is effectively the same as the input* $\mathsf{h}_{\downarrow,v}^{(L)} = (-(x - z_v)^2/2)_{x \in [S]} \in \mathbb{R}^S$*). Assume that* $\|\boldsymbol{W}\| \le B$*. Then for any* ℓ*,* v*,* ι*, and* $\star \in [3]$*, we have*

$$\begin{aligned}
\|\mathsf{h}_{\downarrow,v}^{(L)}\|_2 &\le S^3 + S^2|z_v|, \\
\|\mathsf{q}_{\downarrow,v}^{(\ell)}\|_2 &\le B^3 \cdot (2 \cdot \|\mathsf{h}_{\downarrow,v}^{(\ell)}\|_2 + 1), \\
\|\mathsf{h}_{\downarrow,v}^{(\ell-1)}\|_2 &\le m^{(\ell)} \cdot \max_{v' \in \mathcal{C}(v)} \|\mathsf{q}_{\downarrow,v'}^{(\ell)}\|_2, \\
\|\mathsf{b}_{\uparrow,\mathrm{r}}^{(0)}\|_2 &= \|\mathsf{h}_{\downarrow,\mathrm{r}}^{(0)}\|_2, \\
\|\mathsf{b}_{\uparrow,v}^{(\ell)}\|_2 &\le B^3 \cdot (2\|\mathsf{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)}\|_2 + 2\|\mathsf{q}_{\downarrow,v}^{(\ell)}\|_2 + 1) + \|\mathsf{h}_{\downarrow,v}^{(\ell)}\|_2, \\
\max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{q}_{\downarrow,v,i}^{(\ell)}\|_{\mathrm{op}} &\le 2B^3 \cdot \max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{h}_{\downarrow,v,i}^{(\ell)}\|_{\mathrm{op}}, && \forall k \ge \ell + 1, \\
\max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(\ell)}} \mathsf{q}_{\downarrow,v,i}^{(\ell)}\|_{\mathrm{op}} &\le 2B^2 \cdot \|\mathsf{h}_{\downarrow,v}^{(\ell)}\|_2, \\
\max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{h}_{\downarrow,v,i}^{(\ell-1)}\|_{\mathrm{op}} &\le m^{(\ell)} \cdot \max_{v' \in \mathcal{C}(v)} \max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{q}_{\downarrow,v',i}^{(\ell)}\|_{\mathrm{op}}, && \forall k \ge \ell, \\
\max_{i \in [S]} \|\nabla_{W_{\star,\uparrow,\iota}^{(k)}} \mathsf{b}_{\uparrow,v,i}^{(\ell)}\|_{\mathrm{op}} &\le 2B^3 \cdot \max_{i \in [S]} \|\nabla_{W_{\star,\uparrow,\iota}^{(k)}} \mathsf{b}_{\uparrow,\mathrm{pa}(v),i}^{(\ell-1)}\|_{\mathrm{op}}, && \forall k \ge \ell + 1, \\
\max_{i \in [S]} \|\nabla_{W_{\star,\uparrow,\iota}^{(\ell)}} \mathsf{b}_{\uparrow,v,i}^{(\ell)}\|_{\mathrm{op}} &\le 2B^2 \cdot (\|\mathsf{b}_{\uparrow,\mathrm{pa}(v)}^{(\ell-1)}\|_2 + \|\mathsf{q}_{\downarrow,v}^{(\ell)}\|_2), \\
\max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{b}_{\uparrow,v,i}^{(\ell)}\|_{\mathrm{op}} &\le 2B^3 \cdot \Big(\max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{b}_{\uparrow,\mathrm{pa}(v),i}^{(\ell-1)}\|_{\mathrm{op}} \\
&\qquad + \max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{q}_{\downarrow,v,i}^{(\ell)}\|_{\mathrm{op}} \Big) + \max_{i \in [S]} \|\nabla_{W_{\star,\downarrow,\iota}^{(k)}} \mathsf{h}_{\downarrow,v,i}^{(\ell)}\|_{\mathrm{op}}, && \forall k \in [L], \\
\|\nabla_{W_{\star,\diamond,\iota}^{(k)}} \boldsymbol{m}_{\mathrm{NN}}^{\boldsymbol{W}}(\boldsymbol{z})_v\|_{\mathrm{op}} &\le S \cdot \max_{i \in [S]} \|\nabla_{W_{\star,\diamond,\iota}^{(k)}} \mathsf{b}_{\uparrow,v,i}^{(L)}\|_{\mathrm{op}}, && \forall k \in [L], \diamond \in \{\downarrow, \uparrow\}.
\end{aligned}$$

Proof of Lemma 23. The proof of the lemma uses the chain rule, the 1-Lipschitzness of ReLU, the 2-Lipschitzness of normalize, and the 1-Lipschitzness of softmax. □

Lemma 24. *Consider the U-Net as in Eq. (UNet). Assume that $\|\boldsymbol{W}\| \le B$. Then for any $v \in \mathcal{V}^{(L)}$, we have*

$$\max_{\diamond\in\{\downarrow,\uparrow\}} \max_{\boldsymbol{z}\in\mathbb{R}^d} \max_{\star\in[3]} \max_{\ell\in[L]} \max_{\iota\in[m^{(\ell)}]} \|\nabla_{W^{(\ell)}_{\star,\diamond,\iota}} \boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z})_v\|_{\mathrm{op}} \le 18^L B^{6L} \cdot d \cdot S^3(S + |z_v|).$$

Proof of Lemma 24. This lemma is implied by Lemma 23 and an induction argument. □

Lemma 25. *Consider the ConvNet as in Eq. (UNet). Assume that $\|\boldsymbol{W}\| \le B$. Then for any $v \in \mathcal{V}^{(L)}$, we have*

$$\max_{\boldsymbol{z}} \Big|\boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z})_v - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})_v\Big| \le 6\|\underline{m}\|_1 18^L B^{6L} \cdot d \cdot S^3(S + |z_v|) \cdot (S + D) \cdot \|\boldsymbol{W} - \overline{\boldsymbol{W}}\|.$$

Therefore, we have

$$\max_{\boldsymbol{z}} d^{-1}\Big|\|\boldsymbol{x} - \boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z})\|_2^2 - \|\boldsymbol{x} - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})\|_2^2\Big| \le 12\|\underline{m}\|_1 18^L B^{6L} \cdot d \cdot S^3\Big(\sum_{v\in\mathcal{V}^{(L)}} (S + |z_v|)\Big) \cdot (S + D) \cdot \|\boldsymbol{W} - \overline{\boldsymbol{W}}\|.$$

Proof of Lemma 25. The first inequality is by the fact that

$$\begin{aligned}
&\max_{\boldsymbol{z}} \Big|\boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z})_v - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})_v\Big| \\
\le &\sum_{\diamond\in\{\downarrow,\uparrow\}} \sum_{\star\in[3]} \sum_{\ell\in[L]} \sum_{\iota\in[m^{(\ell)}]} \min\{\mathrm{nrow}(W^{(\ell)}_{\star,\diamond,\iota}), \mathrm{ncol}(W^{(\ell)}_{\star,\diamond,\iota})\} \|\nabla_{W^{(\ell)}_{\star,\diamond,\iota}} \boldsymbol{m}^{\tilde{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})_v\|_{\mathrm{op}} \|W^{(\ell)}_{\star,\diamond,\iota} - \overline{W}^{(\ell)}_{\star,\diamond,\iota}\|_{\mathrm{op}},
\end{aligned}$$

where we have used the inequality that $\mathrm{trace}(A^{\mathbb{T}}B) \le \{\mathrm{nrow}(A), \mathrm{ncol}(A)\}\|A\|_{\mathrm{op}}\|B\|_{\mathrm{op}}$.

To prove the second inequality, we have

$$\begin{aligned}
&\max_{\boldsymbol{z}} \Big|\|\boldsymbol{x} - \boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z})\|_2^2 - \|\boldsymbol{x} - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})\|_2^2\Big| \\
\le &\max_{\boldsymbol{z}} \|2\boldsymbol{x} - \boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z}) - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})\|_\infty \|\boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z}) - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})\|_1 \\
\le &2d \max_{\boldsymbol{z}} \|\boldsymbol{m}^{\boldsymbol{W}}_{\mathrm{NN}}(\boldsymbol{z}) - \boldsymbol{m}^{\overline{\boldsymbol{W}}}_{\mathrm{NN}}(\boldsymbol{z})\|_1.
\end{aligned}$$

This completes the proof of Lemma 25. □