# APPENDIX

## A RELATED WORKS

Positional encoding became a popular topic among the machine learning community after the seminal work on Transformers by Viswani *et al.* [2] Since the attention mechanism used in the Transformers is position-insensitive, they employed a sinusoidal signal to encode the positions before feeding them to the higher blocks. A contemporary work by Gehring *et al.* [1] also proposed a convolutional seq2seq model, adapting a positional encoding mechanism. Since then, using positional encoding in language models became a common trend [18; 19; 20; 21; 22]. Notably, Wang *et al.* [23] extended the embedding space from real numbers to complex values. Another critical aspect of their work is replacing the pre-defined encoding mechanism with a learnable one. There have also been other exciting attempts to improve positional encoding, such as extending the sequential positional encoding to tree-based positional encoding [24], untying the correlations between words and positions while embedding coordinates [25], and modeling positional encoding using dynamical systems [26].

In parallel, positional encoding is also gaining attention in computer vision, specifically with coordinate-MLPs. Coordinate-MLPs provide an efficient method to encode objects such as images [27; 28] and 3D scenes [29; 30; 31] as their weights. Remarkably, Mildenhall *et al.* [7] and Zhong *et al.* [8] found that encoding coordinates with sinusoidal signals allow coordinate-MLPs to learn high frequency content better. One of the earliest roots of this approach can perhaps be traced to the work by Rahimi *et al.* [32], where they used random Fourier features to approximate an arbitrary stationary kernel function by applying Bochner's theorem. More recently, Tancik *et al.* [16], leveraging the NTK theory [33; 34; 35; 36; 37], recently added theoretical rigor to this particular practice by showing that such embeddings enable tuning the spectrum of the NTK of the corresponding MLP. In contrast, the goal of this paper is to show that one does not have to be limited to the Fourier embedding for positional encoding. We demonstrate that alternative functions can be used for positional encoding while gaining similar or better performance compared to Fourier embedding.

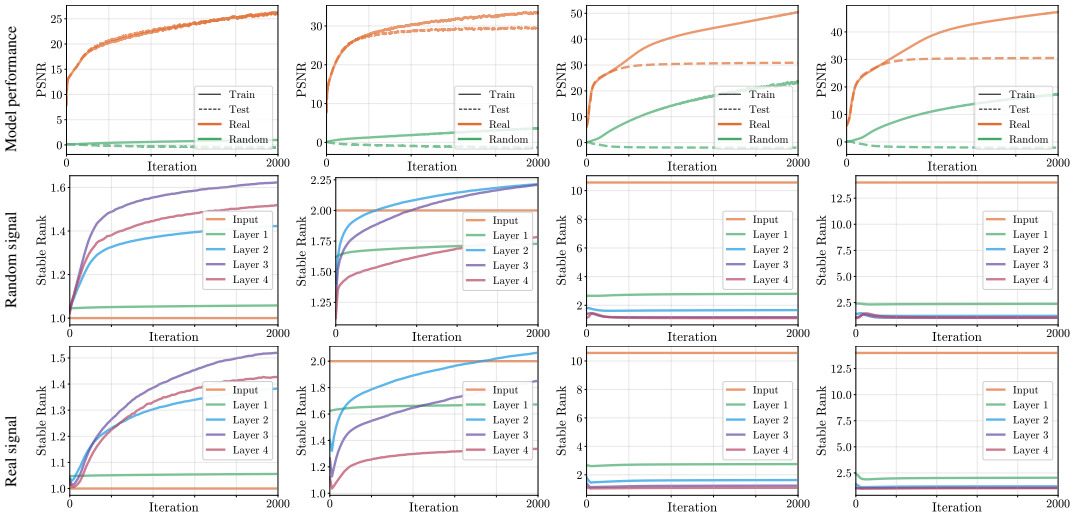
## B THE ROLE OF STABLE RANK IN TRAINING AN MLP

A bulk of our derivations stem from the fact that the stable rank plays a key role in determining the performance of the positional embedding. Therefore, it is compelling to observe the connection of the stable rank to the performance and behavior of a network during training.

In this experiment, we consider *real* signals and *random* signals as the targets. For the real signals, we pick $10$ random rows of the Pepper image. Similarly, we obtain $10$ random signals from a Gaussian distribution. We train a 4-layer MLP with ReLU activation for 2000 epochs with Adam optimizer. The learning rate for original coordinates is $1e^{-3}$, and for others, it is $1e^{-4}$. The averaged results over each of the $10$ rows are reported in Fig. 10. As illustrated, when no encoding mechanism is used, the network attempts to increase the stable rank in each layer, before mapping down to the target, to obtain better performance during the training process. This leads to a considerably higher training time. This trend is common to both real and random signals, although for the random signal, stable rank increases more rapidly. However, since the random signal has no redundancy, the network cannot achieve the desired stable rank, leading to poor performance. In the case of sinusoidal embedder, performance is slightly better since the rank of the input is slightly higher. In contrast, with the RFF and the Gaussian embedder the inputs have sufficient rank, and the network does not have to put effort into increasing the rank. This results in more rapid convergence and better performance. However, the random signals still demand a very high stable rank, causing the networks to demonstrate sub-par performance. Further, observe that the memorization of the training data becomes better as the stable rank of the embedding scheme increase, as predicted in theory.

## C THEORETICAL RESULTS

**Proposition 1.** *Consider a set of coordinates* $\mathbf{x} = [x_1, x_2, \cdots, x_N]^T$*, corresponding outputs* $\mathbf{y} = [y_1, y_2, \cdots, y_N]^T$*, and a* $d$ *dimensional embedding* $\Psi : \mathbb{R} \to \mathbb{R}^d$*. Assuming perfect convergence, the*




(a) No embedder (b) Sinusoidal embedder (c) Gaussian embedder (d) RFF

Figure 10: Connection of the stable rank to the performance and behavior of an MLP. As depicted, if the stable rank of the inputs is too low, the network attempts to increase the stable rank of the hidden layers, causing poor performance and longer training time. As the stable rank of the positional embedding increases, the network demonstrates a lower stable rank at hidden layers, leading to better performance and rapid convergence.

*necessary and sufficient condition for a linear model to perfect memorize of the mapping between* $\mathbf{x}$ *and* $\mathbf{y}$ *is for* $\mathbf{X} = [\Psi(x_1)\ \Psi(x_2)\ \ldots\ \Psi(x_N)]$ *to have full rank.*

**Proof:** Let us refer to the row vectors of $\mathbf{X}$ as $[\mathbf{p}_1, \ldots, \mathbf{p}_d]^T$. In order to perfectly reconstruct $\mathbf{y}$ using a linear learner with weights $\mathbf{w} = [w_1, w_2, \ldots, w_d]$ as

$$\mathbf{y} = \sum_{i=1}^{d} w_i \mathbf{p}_i + b, \tag{7}$$

one needs $\mathbf{X}$ to be of rank $N$ (since $\mathbf{y}$ needs to completely span $\{\mathbf{p}_i\}_{i=1}^{d}$). If $d > N$ then there is no unique solution to $\{\mathbf{w}, b\}$ without some regularization. In the unlikely scenario that the row vectors of $\mathbf{X}$ have zero mean, then $\mathbf{X}$ needs to be of rank $N - 1$ since the bias term $b$ can account for that missing linear basis. □

**Proposition 2.** *Let the Gaussian embedder be denoted as* $\psi(t, x) = \exp(-\frac{\|t-x\|^2}{2\sigma^2})$. *With a sufficient embedding dimension, the stable rank of the embedding matrix obtained using the Gaussian embedder is* $\min(N, \frac{1}{2\sqrt{\pi}\sigma})$ *where* $N$ *is the number of embedded coordinates. Under the same conditions, the embedded distance between two coordinates* $x_1$ *and* $x_2$ *is* $\mathrm{D}(x_1, x_2) = \exp(-\frac{\|x_1-x_2\|^2}{4\sigma^2})$.

**Proof:** Let us define the Gaussian embedder as $\psi(t, x) = \exp(-\frac{\|t-x\|^2}{2\sigma^2})$, where $\sigma$ is the standard deviation. Given $d$ samples points $[t_1, \ldots, t_d]$ and $N$ input coordinates $[x_1, \ldots, x_N]$, the elements of the embedding matrix are

$$\Psi_{i,j} = \psi(t_i, x_j). \tag{8}$$

To make sure the stable rank is saturated, we assume that $d$ and $N$ is large enough. Then, $\Psi$ is approximately a circulant matrix. We know that the singular value decomposition of a circulant matrix $C$, whose first row is $c$, can be written as

$$C = \frac{1}{n} F_n^{-1} diag(F_n c) F_n, \tag{9}$$

where $F_n$ is the Fourier transform matrix. This means the singular values of a circulant matrix is the Fourier transform of first row. When $N$ is large enough, we can approximate the first row of $\Psi$ as a

continuous signal, which is $\psi(x, t=0) = \exp(-\frac{\|x\|^2}{2\sigma^2})$, so the singular values are

$$s(\xi) = \mathcal{F}(\psi(x; t=0)) = \sqrt{2\pi}\sigma \exp(-2\sigma^2\|\pi\xi\|^2). \tag{10}$$

Therefore, we can calculate stable rank directly from the definition,

$$\text{Stable Rank}(\Psi) = \sum_{i=1}^{N} \frac{s_i^2}{s_1^2} = \int_{-\infty}^{+\infty} \frac{s(\xi)}{s(0)}^2 d\xi = \int_{-\infty}^{+\infty} \exp(-4\sigma^2\|\pi\xi\|^2) d\xi = \frac{1}{2\sqrt{\pi}\sigma}. \tag{11}$$

Considering the general case, where $N$ might not be large enough, the stable rank will be $\min(N, \frac{1}{2\sqrt{\pi}\sigma})$.

The distance (or similarity) between two embedded coordinates can be obtained via the inner product:

$$\begin{aligned}
\mathrm{D}(x_1, x_2) &= \int_{-\infty}^{+\infty} \psi(t, x_1)\psi(t, x_2) dt \\
&= \int_{-\infty}^{+\infty} e^{-\frac{(t-x_1)^2}{2\sigma^2}} e^{-\frac{(t-x_2)^2}{2\sigma^2}} dt \\
&= \int_{-\infty}^{+\infty} e^{-\frac{(t-x_1)^2+(t-x_2)^2}{2\sigma^2}} dt \\
&= \int_{-\infty}^{+\infty} e^{-\frac{t^2-2x_1t+x_1^2+t^2-2x_2t+x_2^2}{2\sigma^2}} dt \\
&= \int_{-\infty}^{+\infty} e^{-\frac{2t^2-2(x_1+x_2)t+\frac{(x_1+x_2)^2}{2}+\frac{(x_1-x_2)^2}{2}}{2\sigma^2}} dt \\
&= \int_{-\infty}^{+\infty} e^{-\frac{(t-\frac{x_1+x_2}{2})^2}{\sigma^2}} e^{-\frac{(x_1-x_2)^2}{4\sigma^2}} dt \\
&= e^{-\frac{(x_1-x_2)^2}{4\sigma^2}} \int_{-\infty}^{+\infty} e^{-\frac{(t-\frac{x_1+x_2}{2})^2}{\sigma^2}} dt \\
&= \sqrt{\pi}\sigma e^{-\frac{(x_1-x_2)^2}{4\sigma^2}}.
\end{aligned} \tag{12}$$

which is also a Gaussian with a standard deviation of $\sqrt{2}\sigma$. We can empirically define that the distance between two embedded coordinates $x_1$ and $x_2$ is preserved if $D(x_1, x_2) \geq 10^{-k}$, for an interval $x_1 - x_2 \leq l$, where $k$ is a threshold. In the Gaussian embedder, we can analytically obtain a $\sigma$ for an arbitrary $l$ using the relationship $\sigma = \frac{l}{2\sqrt{k \ln 10}}$. □

**Proposition 3.** *Let the RFF embedding be denoted as $\gamma(x) = [\cos 2\pi\mathbf{b}x, \sin 2\pi\mathbf{b}x]$, where $\mathbf{b}$ are sampled from a Gaussian distribution. When the embedding dimension is large enough, the stable rank of RFF will be $\min(N, \sqrt{2\pi}\sigma)$, where $N$ is the numnber of embedded coordinates. Under the same conditions, the embedded distance between two coordinates $x_1$ and $x_2$ is $\mathrm{D}(x_1, x_2) = \sum_j \cos 2\pi b_j(x_1 - x_2)$.*

**Proof:** Given $\frac{d}{2}$ samples for $\mathbf{b}$ as $[b_1, \ldots, b_{\frac{d}{2}}]$ from a Gaussian distribution with a standard deviation $\sigma$ and $N$ input coordinates $[x_1, \ldots, x_N]$, RFF embedding is defined as $\gamma(x) = [\cos 2\pi\mathbf{b}x_i, \sin 2\pi\mathbf{b}x_i]$.

To make sure the stable rank is saturated, we assume that the $d$ and $N$ is large enough. Although RFF embedding matrix is not circulant, it is naturally frequency based so we already know its spectrum, which is its singular value distribution

$$s(\xi) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{\xi^2}{2\sigma^2}). \tag{13}$$

Similarly,

$$\text{Stable Rank}(\gamma) = \sum_{i=1}^{N} \frac{s_i^2}{s_1^2} = \int_{-\infty}^{+\infty} \frac{s(\xi)}{s(0)}^2 d\xi = \int_{-\infty}^{+\infty} \exp(-\frac{\xi^2}{2\sigma^2}) d\xi = \sqrt{2\pi}\sigma, \tag{14}$$

Considering the general case, the stable rank is $\min(N, \sqrt{2\pi}\sigma)$.

From basic trigonometry, it can be easily deduced that $\mathrm{D}(x_1, x_2) = \sum_j \cos 2\pi b_j (x_1 - x_2)$. When $d$ is extremely large it can be considered as $f(\xi) = \cos 2\pi\xi(x_1 - x_2)$ where $\xi$ is a Gaussian random variable with standard deviation $\sigma$. Then the above sum can be replaced with the integral,

$$\begin{aligned}
\mathrm{D}(x_1, x_2) &= \int_{-\infty}^{+\infty} e^{-\frac{\xi^2}{2\sigma^2}} \cos 2\pi\xi(x_1 - x_2) d\xi \\
&= 2\int_{0}^{+\infty} e^{-\frac{\xi^2}{2\sigma^2}} \cos 2\pi\xi(x_1 - x_2) d\xi \\
&= 2\int_{0}^{+\infty} e^{-\frac{\xi^2}{2\sigma^2}} \frac{1}{2}(e^{i2\pi(x_1-x_2)\xi} + e^{-i2\pi(x_1-x_2)\xi}) d\xi \\
&= \int_{0}^{+\infty} e^{-\frac{\xi^2}{2\sigma^2} + i2\pi(x_1-x_2)\xi} + e^{-\frac{\xi^2}{2\sigma^2} - i2\pi(x_1-x_2)\xi} d\xi.
\end{aligned} \tag{15}$$

Further,

$$\int_0^{+\infty} e^{-ax^2+bx} dx = e^{-\frac{b^2}{4a}} \int_0^{+\infty} e^{-a(x-i\frac{b}{2a})^2} dx = \frac{1}{2}(1 + \mathrm{erfi}(\frac{b}{2\sqrt{a}}))\sqrt{\frac{\pi}{a}} e^{-\frac{b^2}{4a}}. \tag{16}$$

Let $a = \frac{1}{2\sigma^2}$ and $b = \pm 2\pi(x_1 - x_2)$. Then, we have

$$\mathrm{D}(x_1, x_2) = \sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2(x_1-x_2)^2}. \tag{17}$$

□

## D SAMPLING IN HIGHER DIMENSIONS

In cases where there are restrictions with the used the MLP, one can always increase the rank of the embedding matrix by sampling the embedder along multiple directions (as opposed to only two directions). Fig. 11 depicts this. As shown, by sampling along four directions, the Gaussian embedder is able to obtain distance preservation along more directions and on-par results against the RFF embedder. In this experiment, for each embedding dimension and each sampling method in Gaussian embedding, we conduct a linear search for the standard deviation that gives the best test PSNR. For RFF, a constant standard deviation performed equally for all variants. For Gaussian embedder, regardless of the sampling method, the empirical best standard deviation is related to the number of samples along each direction, which is $\sigma = \frac{k}{4d\sqrt{\ln 10}}$ where $d$ is the embedding dimension and $k = 3.5$ from experimental results.

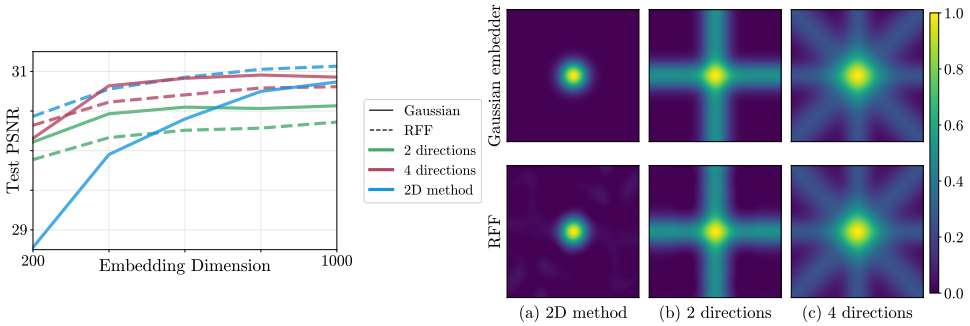



Figure 11: Comparison of the test performance and distance preservation across different sampling methods in Gaussian embedding and RFF, when encoding 2-D signals. Note that in the context of RFF, sampling refers to sampling of the frequencies. *Left*: By sampling along four directions, the Gaussian embedder is able to achieve improved performance. When the embedding dimension is low, sampling in an equidistant grid on the $x - y$ plane (2D method) results in sparse sampling, leading to sub-par performance. However, as the embedding dimension increases, this method demonstrates superior performance. *Right*: Distance preservation depends on the sampling method for both RFF and the Gaussian embedder.