




Self-gating stochastic-resonance-based autoencoder for unsupervised learningYuhao Ren,¹ Fabing Duan ^{1,*} François Chapeau-Blondeau ² and Derek Abbott ³¹*Institute of Complexity Science, Qingdao University, Qingdao 266071, People's Republic of China*²*Laboratoire Angevin de Recherche en Ingénierie des Systèmes (LARIS),
Université d'Angers, 62 Avenue Notre Dame du Lac, 49000 Angers, France*³*Centre for Biomedical Engineering (CBME) and School of Electrical & Electronic Engineering,
The University of Adelaide, Adelaide, South Australia 5005, Australia*

(Received 23 December 2023; revised 15 May 2024; accepted 7 June 2024; published 1 July 2024)

Incorporating additive noise components to an ensemble of McCulloch-Pitts neurons can enhance the information representation of the input, asymptotically approaching the average firing probability for large enough ensembles. We further multiply the input by the average firing probability to control the higher probability of self-gating, thereby forming a unified noise-boosted activation model with learnable noise-related hyperparameters. This gating strategy plays a crucial role in improving the performance of neural networks, as evidenced by the optimization of the autoencoder loss at nonzero optimal-noise-scaling hyperparameters, a phenomenon termed self-gating stochastic resonance. Experiments with designed autoencoders using noise-boosted activation functions demonstrate the potential applications of the self-gating stochastic resonance effect in the field of unsupervised learning.

DOI: [10.1103/PhysRevE.110.014107](https://doi.org/10.1103/PhysRevE.110.014107)**I. INTRODUCTION**

Noise injection, as a powerful method of improving neural network generalization ability, has attracted the attention of researchers for unsupervised learning [1–5]. It is widely recognized that *dropout* [6] can be viewed as injecting multiplicative noise into synaptic weights to prevent neural networks from overfitting. This is because a random zero denotes weight pruning while a random one preserves it. Consequently, the dropout rate, as a hyperparameter, is applied during network training to prevent over-reliance on specific neurons or features and promote a more robust data representation. In contrast to multiplicative noise, some interesting results for enhancing neural network ability benefit from additive noise. In the last few years, in particular, regularizing denoising [1], contractive [2], variational [3], and graph autoencoders [7] through noise injection has become a research focus. It has been demonstrated that training with adding noise into the input [1,7–9], the hidden units [4], or the latent space [10] can yield desired low-dimensional representations and high-fidelity reconstructions of input data.

However, it must be noted that, in the context of nonlinear mappings such as a neural network employing sigmoid or rectified linear unit (ReLU) activations, the equivalence of the noise injection with Tikhonov regularization holds true only in the asymptotic regime of small injected noise [1,11–13]. For large injected noise level, the rigorous expansion of infinitesimal parameters becomes impracticable, thereby invalidating the equivalence of noise injection to a smoothing regularization term [1]. In addition, both theoretical and experimental

results clearly show the lack of equivalence between the two approaches under the conditions of the injected noise scaled by the nonlinear activation function and the weights [11,14].

Nonlinearity of a neural network has its origin in activation functions that endow the network with the ability to process data and learn features [5,15–18]. Now, the widely used activation function is the ReLU $\max(x, 0)$ [16] in training deep neural networks for image-related tasks, because it allows a network to easily obtain sparse representations [17]. Supporting the biological plausibility, the activation functions modeling vertebrate retinal neurons, similar to the ReLU, also exhibit the inactivation below zero and yield the unbounded response above zero [19,20]. Nevertheless, the nonsmoothness and the monotonicity of ReLU do not accord with the pointwise nonlinearity of vertebrate retinal neuron [19,20], and hence, certain smooth and nonmonotonic activation functions, e.g., parametric ReLU [21], the Gaussian error linear unit (GELU) [22], the sigmoid linear unit (SiLU) [22–24], the active or not (ACON) unit [25], the parametric rectified linear unit (PRLU) [26], and the nonbistable rectified linear unit [27], have been proposed. The learning of hyperparameters associated with these activations [28] has also been introduced to mimic the adaptation of visual neurons [19,20]. Moreover, these proposed units with adjustable characteristics can improve the neural network performance with only a small increase in the complexity of the network architecture [19–25,28]. As a result, the exploration of learnable activation functions has emerged as a prominent research direction in the domain of machine learning.

Naturally, some interesting questions arise. In contrast to the ReLU activation function, these learnable activation functions [19–25,28] typically exhibit negative responses for negative inputs near zero, but with saturation for very small

*Contact author: fabingduan@qdu.edu.cn

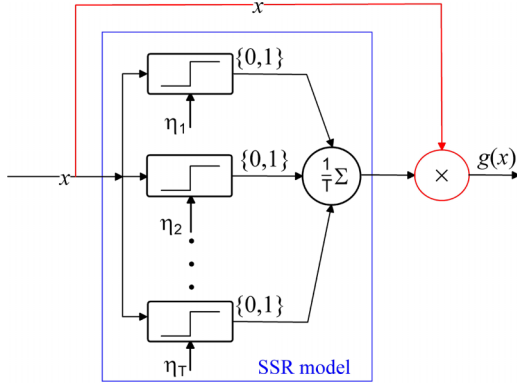


FIG. 1. Noise-boosted activation functions based on the SSR model.

negative inputs. This observation motivates us to investigate the following questions: does the negative region of these functions within the subthreshold regime hold any physical significance, and how does it relate to the firing rate of neurons? What intricate interconnections are established by learning among these nonlinear activations [22–26,28,29]? Is it possible to consolidate these diverse activations into a unified neuron model also capable of eliciting novel ones? How can we elucidate the physical significance and evaluate the role of learnable hyperparameters in this neuron model? To address these critical questions, this paper proposes a unified noise-boosted activation model with learnable noise-related hyperparameters. This model incorporates elements from existing activation functions while introducing learnable parameters that can be optimized during the training process. We investigate the impact of these learnable parameters on the network performance and also explore the potential biological relevance of the resulting activation dynamics.

II. ACTIVATION MODEL

In order to address the aforementioned queries, we first investigate the interconnections among these activations from a perspective of the suprathreshold stochastic resonance (SSR) model [30] illustrated in Fig. 1, which consists of an array of McCulloch-Pitts [31] or bipolar neurons [19,20]

$$h(x) = \frac{1}{2}[1 + \text{sgn}(x)] \quad (1)$$

subjected to a common input x and mutually independent noise components η_t for $t = 1, 2, \dots, T$. Here, $\text{sgn}(x)$ denotes the signum function. Facilitated by the nonzero optimal noise components η_t , the output of the SSR model

$$\bar{g}(x) = \frac{1}{T} \sum_{t=1}^T h(x + \eta_t) \quad (2)$$

effectively aggregates information from each neuron [30]. Assume that the injected noise components η_t have the common probability density function (PDF) $f_\eta(\eta)$, then each neuron yields a response of unity with probability

$$p(x) = \int_{-x}^{\infty} f_\eta(\eta) d\eta = 1 - F_\eta(-x), \quad (3)$$

where $F_\eta(u) = \int_{-\infty}^u f_\eta(\eta) d\eta$ denotes the cumulative distribution function (CDF). For a sufficiently large number T of neurons [30,32], the output $\bar{g}(x)$ of the SSR model approaches close to the average firing probability $p(x)$, i.e., $\lim_{T \rightarrow \infty} \bar{g}(x) = p(x)$. Within the framework of McCulloch-Pitts neurons, the firing probability $p(x)$ is emphasized to persist even for negative inputs ($x < 0$) under the influence of background noise $\eta(t)$. This can be attributed to the possibility of the summation $x + \eta_t$ exceeding zero, which triggers a firing response of unity from the neuron.

Gates play a critical role in enabling long short-term memory (LSTM) networks to excel at processing sequential data [33]. Within each gate, a nonlinear activation function modulates the information flow through a subsequent multiplication operation. The activation function generates outputs ranging from 0 to 1, effectively representing the likelihood of the gate being open or closed. Values close to unity indicate an open gate, allowing uninhibited information flow. Conversely, values close to zero signify a closed gate, thereby impeding information transmission [33]. Motivated by the gating mechanism of the LSTM network [33], we conceptualize the SSR model as a gating operator acting on the input, as shown in Fig. 1. Alternatively, this gating operator deactivates and yields a zero output with probability $1 - p(x)$, while transmitting the input with probability $p(x)$. Therefore, we obtain a noise-boosted activation function expressed as

$$g(x) = xp(x) = x [1 - F_\eta(-x)]. \quad (4)$$

Interestingly, without the injection of noise η , the aggregation $\sum_{t=1}^T h(x + \eta_t)/T = h(x)$ simplifies to a single McCulloch-Pitts neuron, and $g(x) = xh(x) = \max(x, 0)$ represents the ReLU [16]. By injecting diverse noise types into the SSR model of Fig. 1, the existing activations in Refs. [22–24] can be deduced from Eq. (4). For instance, when introducing logistic noise η with its PDF $f_\eta(x) = e^{-x/\sigma}/[\sigma(1 + e^{-x/\sigma})^2]$ and CDF $F_\eta(x) = 1/(1 + e^{-x/\sigma})$ for the scale parameter $\sigma > 0$, Eq. (4) evolves into the SiLU [22–24] given by

$$g(x, \sigma) = \frac{x}{1 + e^{-\frac{x}{\sigma}}}. \quad (5)$$

When considering injected noise with a Gaussian PDF $f_\eta(x) = \exp(-x^2/2\sigma^2)/\sqrt{2\pi\sigma^2}$, the noise-boosted activation function of Eq. (4) becomes the GELU [22] defined as

$$g(x, \sigma) = x\Phi(x/\sigma) \quad (6)$$

and can also be derived from Eq. (4). Here, $\Phi(x) = \frac{1}{2} + \frac{1}{2}\text{erf}(x/\sqrt{2})$ denotes the CDF of a standard Gaussian random variable ($\sigma = 1$). Moreover, novel activation functions can be derived from Eq. (4) according to various noise distributions. For example, considering the exponential noise PDF $f_\eta(x) = (e^{-x/\sigma})/\sigma$ on the support interval $x \in [0, \infty)$, we obtain the exponential linear unit (ExLU) as

$$g(x, \sigma) = \begin{cases} x, & x \geq 0, \\ xe^{x/\sigma}, & x < 0. \end{cases} \quad (7)$$

Similarly, for the Rayleigh noise PDF $f_\eta(x) = x \exp(-x^2/2\sigma^2)/\sigma^2$ over the interval $x \in [0, \infty)$, the corresponding activation function is referred to as the

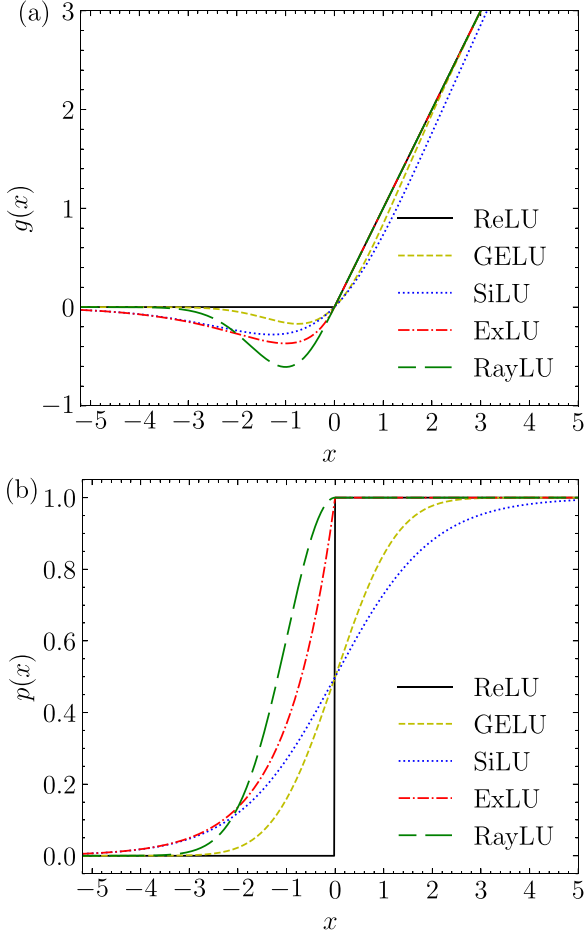


FIG. 2. (a) Noise-boosted activations of ReLU, GELU, SiLU, ExLU, and RayLU with (b) the corresponding average firing probabilities $p(x)$. Here, the noise-scale parameter $\sigma = 1$ for GELU, SiLU, ExLU, and RayLU.

Rayleigh linear unit (RayLU), defined as

$$g(x, \sigma) = \begin{cases} x, & x \geq 0, \\ xe^{-\frac{x^2}{2\sigma^2}}, & x < 0. \end{cases} \quad (8)$$

As illustrated in Fig. 2(a), the activations derived from Eq. (4), namely, GELU, SiLU, ExLU, and RayLU, are smooth and thus differentiable across all input values, distinguishing them from ReLU. Significantly, these noise-boosted activations exhibit a nonmonotonic behavior in the negative input region attributed to the nonzero hyperparameter σ , converging to ReLU as the hyperparameter σ approaches zero. When the hyperparameter $\sigma > 0$, the noise-boosted activations of GELU, SiLU, ExLU, and RayLU explicitly manifest a nonzero nonlinear transformation within the negative domain of x . This alteration arises from the integration of noise, thereby endowing even negative x inputs below the threshold with a firing probability, as indicated in Fig. 2(b). We propose that the interpretation of continuous-valued neuronal output depends on the stage of information processing. At the postsynaptic level, it is typically construed as an average firing rate, reflecting the overall activity of the neuron. However, at the presynaptic stage, where neurons communicate with each

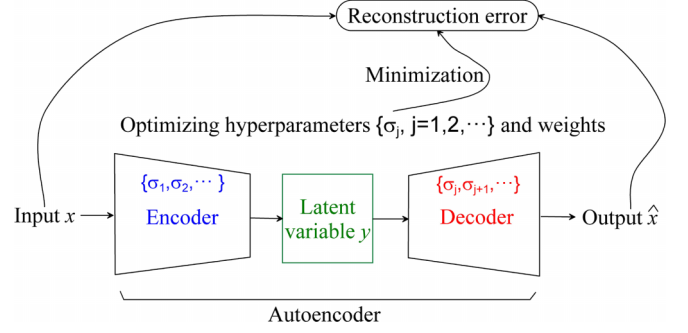


FIG. 3. Diagram of the designed autoencoder with the hyperparameter set $\{\sigma_j, j = 1, 2, \dots\}$ in hidden layers, as introduced in Eq. (4).

other, the continuous activity can be more accurately interpreted as an average membrane potential, or depolarization. In this context, the significance of negative activity becomes readily apparent. Negative values in our model represent hyperpolarization of the presynaptic membrane, which serves to inhibit the firing probability of the postsynaptic neuron.

III. MAIN RESULTS OF THE AUTOENCODER BASED ON NOISE-BOOSTED ACTIVATIONS

Next, we integrate these noise-boosted activation functions into the autoencoder for tasks such as dimensional reduction or denoising and explore the physical significance of learnable hyperparameters. As depicted in Fig. 3, the input x originating from a potentially high-dimensional space \mathcal{X} is transformed by the encoder Γ into the latent variable $y = \Gamma(X)$ within a low-dimensional space \mathcal{Y} . The decoder \mathcal{D} generates the output $\hat{x} = \mathcal{D}(y)$ to reconstruct the input x from the latent variable y . Here, the training criterion for autoencoders involves minimizing the reconstruction error $\mathcal{L} = \sum_{x \in \mathcal{X}} \|x - \hat{x}\|_2$.

It is worth noting that the noise-boosted activation functions described in Eq. (4) introduce hyperparameters in both the encoder and the decoder, thereby forming a set of hyperparameters $\{\sigma_j, j = 1, 2, \dots\}$ to be optimized. It is noteworthy that the introduction of the noise-scale hyperparameter σ is limited to the corresponding hidden layer, resulting in a relatively modest increase in the design complexity of the proposed autoencoder. Therefore, we can adaptively learn hyperparameters through the minibatch gradient descent (GD) approach [13, 15]:

$$\sigma_j^t = \sigma_j^{t-1} - \alpha^t \sum_{x \in \mathcal{B}} \left. \frac{\partial \mathcal{L}(x; \sigma_j)}{\partial \sigma_j} \right|_{\sigma_j = \sigma_j^{t-1}}, \quad (9)$$

where \mathcal{B} is a minibatch sampled from the space \mathcal{X} , and $\alpha^t > 0$ represents the learning rate at the t th training epoch.

A. Motivated experiment of the GELU autoencoder and stochastic resonance

Henceforth, we demonstrate the manifestation of stochastic resonance phenomenon within the proposed autoencoder, incorporating noise-boosted activation functions of Eq. (6). An illustrative fully connected autoencoder with five layers is constructed (source codes are provided in Ref. [34]),

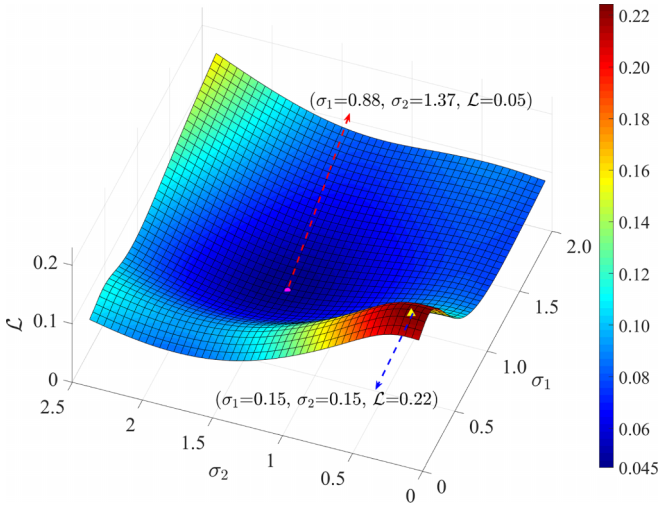


FIG. 4. Mean-squared-error surface of the GELU autoencoder as a function of hyperparameters σ_1 and σ_2 on the Iris dataset.

comprising the encoder Γ ($4 \times 10 \times 2$) and the decoder \mathcal{D} ($2 \times 10 \times 4$). In the architecture of this devised autoencoder, the input x of the Iris dataset [35] comprises four features and undergoes transformation through ten GELU activation functions within the encoder Γ , yielding a two-dimensional latent variable y . Conversely, the decoder \mathcal{D} maps the latent variable y to the reconstructed output \bar{x} using a layer of ten GELU neurons. It is noteworthy that both the encoder and the decoder utilize a single layer comprised of GELU activation functions with a common hyperparameter: σ_1 for the encoder Γ and σ_2 for the decoder \mathcal{D} . This configuration results in a parsimonious model with only two additional hyperparameters σ_1 and σ_2 , which will be optimized concurrently with the weight coefficients of linear layers in the autoencoder by the gradient descent methodology as indicated in Eq. (9).

After 200 training epochs on the Iris dataset consisting of 150 data points, the designed autoencoder achieves a minimum reconstruction error of $\mathcal{L} = 0.05$ at the converged hyperparameter values $\sigma_1 = 0.88$ and $\sigma_2 = 1.37$. Subsequently, with the autoencoder weights fixed at their optimal values, the performance landscape of the reconstruction error \mathcal{L} is plotted in Fig. 4 as a function of σ_1 and σ_2 within their respective ranges of $[0, 2] \times [0, 2.5]$. It is observed in Fig. 4 that the reconstruction error \mathcal{L} attains a local optimum of 0.05 at the optimal coordinate pair $(\sigma_1 = 0.88, \sigma_2 = 1.37)$. This observation manifests the phenomenon of stochastic resonance with nonzero optimal levels (σ_1, σ_2) minimizing the reconstruction error values of \mathcal{L} . It is noteworthy in Eq. (6) that an excessively small value of the hyperparameter σ corresponds to the ReLU activation, while an excessively large value of σ corresponds to the linear activation function. Therefore, an optimal nonzero hyperparameter σ in Eq. (6) signifies a nonlinear transfer function exhibiting curvature at all input values of x with increasing the noise scale σ . This experiment demonstrates the physical significance of the proposed noise-boosted activation model of Eq. (4) within the context of the stochastic resonance mechanism. Given the interpretation of the hyperparameter σ in the noise-boosted activation function of Fig. 1, a nonzero value of σ demonstrates the

beneficial action of a noise-driven probability gate for the input x . Hence, we call this phenomenon depicted in Fig. 4 the self-gating stochastic resonance effect observed during autoencoder learning.

B. Experiments on the MNIST dataset

However, there is a possibility for the noise parameter σ to converge to a value outside its defined domain or to fail to converge. As depicted in Fig. 4, upon initialization with $(\sigma_1, \sigma_2) = (0.15, 0.15)$, the optimization procedure, employing gradient descent methodology, may lead to convergence issues, wherein the hyperparameter pair (σ_1, σ_2) tends towards negative values or suffers the problem of exploding gradients. For instance, the gradient of the ExLU function defined by Eq. (7) with respect to the scale parameter σ can be expressed as

$$\frac{\partial g(x, \sigma)}{\partial \sigma} = -\frac{x^2}{\sigma^2} e^{\frac{x}{\sigma}}, \quad (10)$$

approaching infinity as σ approaches zero and the input $x > 0$.

For such situations, we can utilize the tree-structured Parzen estimator (TPE) approach [36] to optimize the hyperparameter σ by the acquisition function of the expected improvement:

$$\mu_{\text{EI}}(\sigma) = \mathbb{E}_{\mathcal{L}}[\max\{\mathcal{L}_{\min} - \mathcal{L}(\sigma), 0\}], \quad (11)$$

where the loss \mathcal{L} is assumed to be a Gaussian random variable $\mathcal{L} \sim \mathcal{N}[\hat{c}(\sigma), \widehat{\text{var}}(\sigma)]$ and $\mathbb{E}_{\mathcal{L}}$ means the expectation operator with respect to this Gaussian distribution of \mathcal{L} . Here, the mean $\hat{c}(\sigma)$ represents the model prediction, and the variance $\widehat{\text{var}}(\sigma)$ indicates the posterior uncertainty [36]. The continuous hyperparameter σ is assumed to have a uniform prior over the interval $[0, a]$. It is important to note that in the subsequent experiments, the upper bound is set to $a = 10$, and the hyperparameter optimization software utilized is OPTUNA [37].

We perform an experiment on the MNIST dataset employing a fully connected vanilla autoencoder. The architecture comprises an encoder Γ with dimensions $784 \times 2000 \times 256$, a ten-dimensional latent variable y , and a decoder \mathcal{D} with dimensions $256 \times 2000 \times 784$ (source codes are provided in Ref. [34]). Here, the sigmoid function employed in the final layer of \mathcal{D} confines the output to the interval $[0, 1]$. Three layers of Γ and two layers of \mathcal{D} pass through ReLU or noise-boosted activation functions defined in Eq. (4). We randomly select 50 000 images as the training data and 10 000 images as the test data and calculate the reconstruction error \mathcal{L} between the original image and the reconstructed one. In addition, aiming to assess the representation capability of the low-dimensional latent variable y , we also utilize the support vector machine classifier for classification. It is important to note that the labels of the MNIST dataset are utilized for classification, but not for training designed autoencoders.

As presented in Table I, the autoencoders utilizing SiLU, ExLU, GELU, and RayLU activation functions exhibit lower reconstruction errors \mathcal{L} and higher accuracies compared to the autoencoder using ReLU activation functions. By compressing the high-dimensional data into a ten-dimensional latent variable y , Table I demonstrates that, utilizing the minibatch GD learning rule of Eq. (9), the SiLU, ExLU, GELU, and

TABLE I. Results of the designed autoencoder on the MNIST dataset.

Approach $g(x)$	Minibatch GD		OPTUNA	
	$\mathcal{L} (10^{-2})$	Accuracy (%)	$\mathcal{L} (10^{-2})$	Accuracy (%)
ReLU	1.80	91.74	–	–
SiLU	1.05	94.24	1.06	94.13
ExLU	1.05	93.84	1.10	94.93
GELU	1.06	94.29	1.08	94.61
RayLU	1.09	93.69	1.14	95.20

RayLU autoencoders exhibit superior representational abilities in comparison to the ReLU autoencoder, resulting in classification accuracies that exceed the ReLU autoencoder by 2.50%, 2.10%, 2.55%, and 1.95%, respectively. Moreover, the ExLU autoencoder exhibits a significant decrease in the reconstruction error \mathcal{L} , achieving a value of 1.06 compared to the 1.79 obtained with the ReLU autoencoder. This corresponds to a 41% reduction in \mathcal{L} , indicating a superior ability of the ExLU architecture to capture the underlying data representation.

We argue that the enhanced representational capacity of the autoencoder stems from the five learnable hyperparameters σ_j for $j = 1, 2, \dots, 5$. In Fig. 5, the learning curve of the reconstruction error \mathcal{L} for the designed GELU autoencoder and the corresponding hyperparameters σ_j for $j = 1, 2, \dots, 5$ are presented. It is evident that σ_2 exhibits a constant growth, while the remaining hyperparameters σ_j start from the same initial value and converge to distinct but nonzero optima. Interestingly, the value of σ_2 being significantly larger than unity does not impact the convergence of \mathcal{L} and results in $g(x)$ behaving as a linear activation function in the proximity of zero.

To circumvent the issue of nonconvergence, we employ the TPE Bayesian optimization approach [36] described in Eq. (11) to optimize the hyperparameters of the designed autoencoders, and we also present the corresponding experimental results in Table I. For the designed GELU autoencoder, the level of the added noise converges to the optimal values σ_j for $j = 1, 2, \dots, 5$ that are determined to be 2.22, 3.49, 9.64, 1.22, and 1.38, respectively. As shown in Fig. 6, the level curves of the reconstruction error \mathcal{L} for the GELU autoencoder illustrate the convergence process of hyperparameters σ_1 and σ_2 . For simplicity, other level curves of the reconstruction error \mathcal{L} versus hyperparameters σ_j are not shown here. Furthermore, the RayLU autoencoder attains the highest classification accuracy of 95.20% among all considered autoencoder models, surpassing the ReLU one by 3.46%. This outcome once again highlights the useful contribution from the generalization of the unified model in Eq. (4) and from the tunable level of added noise with the hyperparameter.

Figure 6 also reveals that the reconstruction error \mathcal{L} exhibits an increasing trend for both high and low values of hyperparameters σ_1 and σ_2 . Hence, the presence of nonzero converged noise hyperparameters σ_j in Fig. 6 provides evidence for the occurrence of the self-gating stochastic resonance effect in the designed autoencoder optimized by the TPE Bayesian approach. This phenomenon is characterized

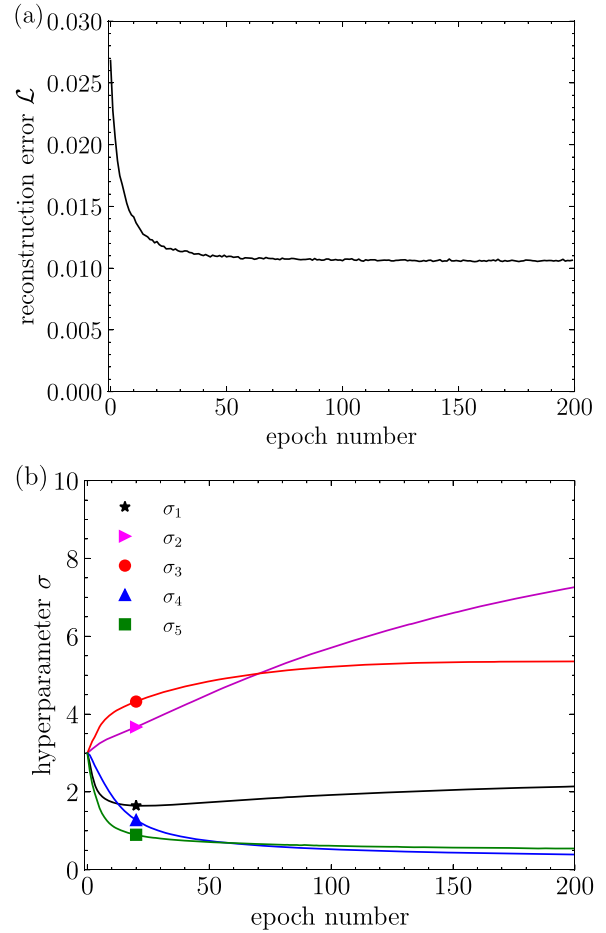


FIG. 5. Plots of (a) the reconstruction error \mathcal{L} and (b) the hyperparameters σ versus the epoch numbers for training the fully connected GELU autoencoder on the MNIST dataset.

by the deliberate introduction of nonzero optimized noise levels into the activation model of Eq. (4), strategically employed to enhance the autoencoder performance.

C. Experiments of convolution autoencoders on the Olivetti face dataset

Next, we construct a convolution autoencoder with three convolution layers and four layers passing through the ReLU or noise-boosted activation functions in both the decoder \mathcal{D} and the encoder Γ (source codes are provided in Ref. [34]). The latent variable y is with the dimension 30, and the Olivetti face dataset images are employed. In order to avoid the limited original size of the Olivetti face dataset, we augmented this dataset by performing random rotations and crops, resulting in a total 4000 of images. Then, 3200 images were selected from the augmented dataset as the training set, and the remaining 800 images were used for the testing. In addition, besides the reconstruction error \mathcal{L} , we also calculated the peak signal-to-noise ratio (PSNR) during testing to assess the quality of the reconstructed image.

Experimental results of the designed convolution autoencoder on the augmented Olivetti face dataset are listed in Table II. It is seen again that the designed autoencoders

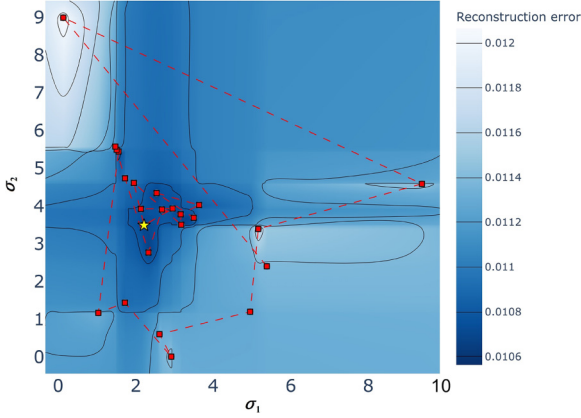


FIG. 6. Level curves of the reconstruction error as a function of hyperparameters σ_1 and σ_2 for the GELU autoencoder, and the dashed line indicates the convergence process of hyperparameters at nonzero optimal levels of the added noise. The yellow colored star represents the final converged coordinate.

with the noise-boosted activation functions of SiLU, ExLU, GELU, and RayLU outperform the ReLU autoencoder in both the measures of the reconstruction error and the PSNR of recovered images. For comparison, Fig. 7 illustratively presents the original images and the reconstructed ones by the ReLU and GELU autoencoders, respectively. We can visually appreciate how the GELU autoencoder optimized through hyperparameters yields reconstructed images with heightened clarity.

D. Experiments of denoising convolutional autoencoders on scanned texts

Finally, we design the denoising convolutional autoencoders to process a dataset of images of scanned text [38]. The detailed autoencoder architecture consists of two convolution layers and two layers passing through ReLU or noise-boosted activation functions in both the decoder and the encoder (refer to source code [34] for implementation details). The comparison results of the reconstructed error \mathcal{L} are provided in Table III, and the reconstructed image samples of ReLU and RayLU autoencoders are illustrated in Fig. 8, respectively. Substantially outperforming the ReLU autoencoder, the RayLU autoencoder achieves remarkable reconstruction accuracy and generates noticeably sharper denoised images.

TABLE II. Results of the designed autoencoder on the Olivetti face dataset.

Approach	Minibatch GD		OPTUNA	
	\mathcal{L} (10^{-3})	PSNR (dB)	\mathcal{L} (10^{-3})	PSNR (dB)
ReLU	3.01	25.21	–	–
SiLU	2.78	25.56	2.79	25.54
ExLU	2.94	25.32	2.89	25.39
GELU	2.85	25.45	2.76	25.58
RayLU	2.94	25.32	2.89	25.39

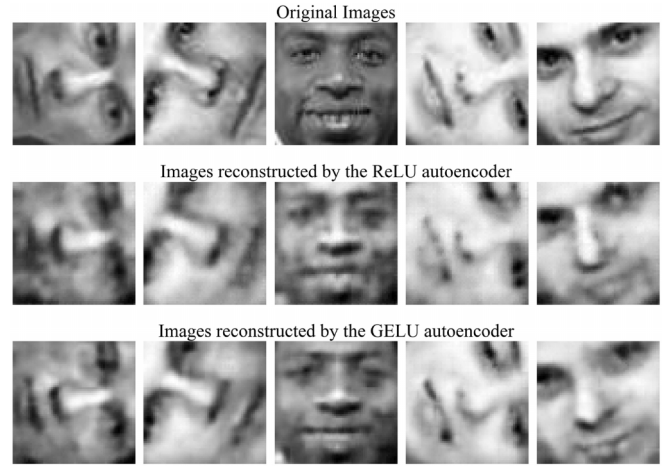


FIG. 7. Original image samples of the Olivetti face dataset and reconstructed image samples of ReLU and GELU autoencoders.

IV. DISCUSSION

To summarize, in this paper we proposed and analyzed a set of noise-boosted activation functions derived from the stochastic resonance model. This framework serves as a bridge connecting stochastic resonance principles with deep learning methodologies within the domain of data science. For considered unsupervised learning tasks, we incorporate these noise-boosted activation functions, such as SiLU, GELU, ExLU, and RayLU, to assess their efficacy in both fully connected and convolutional autoencoders. Experimental results indicate that the devised autoencoders outperform the commonly employed ReLU-based counterparts, particularly in tasks related to image reconstruction.

It is foreseeable that the approach of injecting noise to hidden layer nodes and constructing noise-boosted activation functions can be generalized to various neural network architectures, extending beyond fully connected and convolutional neural networks. Furthermore, innovative noise-boosted activation functions can be obtained through statistical averages in probability-based structures, potentially harnessing their effectiveness in deep learning and exhibiting more intriguing characteristics of the self-gating stochastic resonance effect.

The fundamental characteristic of stochastic resonance phenomena is the existence of an optimal nonzero noise level that amplifies system performance. We integrate the noise parameter into the formulation of the activation function, wherein the nonzero optimized noise level corre-

TABLE III. Results of the designed autoencoder on the scanned text dataset.

Approach	Minibatch GD \mathcal{L}		Optuna \mathcal{L}
	\mathcal{L}	PSNR (dB)	\mathcal{L}
ReLU	0.13	–	–
SiLU	0.12	–	0.12
ExLU	0.12	–	0.11
GELU	0.12	–	0.12
RayLU	0.11	–	0.11

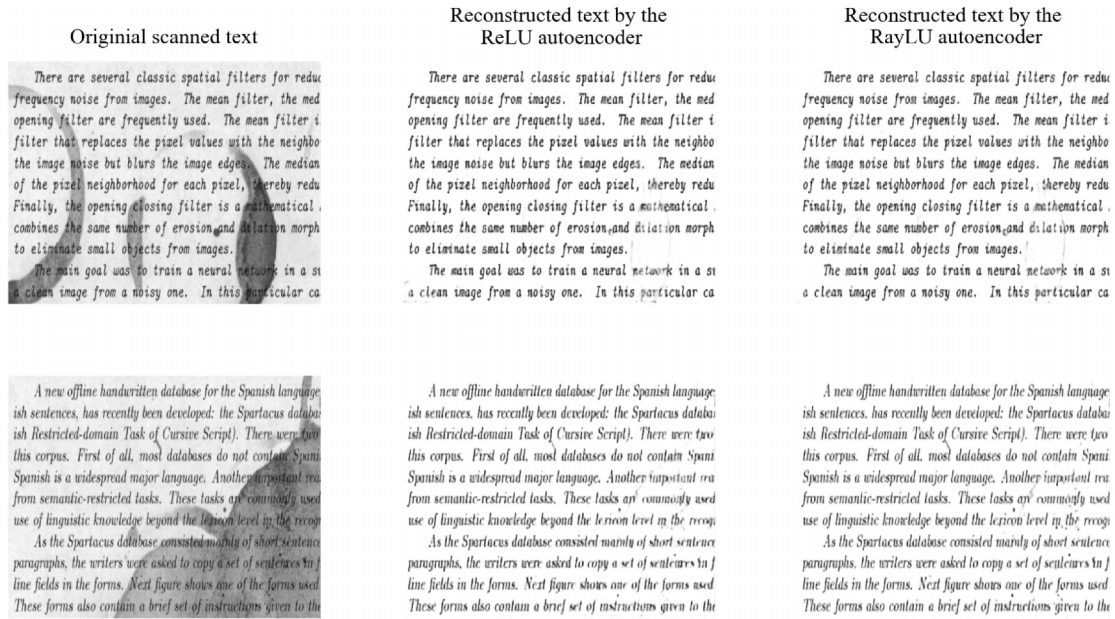


FIG. 8. Reconstructed images by ReLU and RayLU autoencoders for two original image samples of the scanned text.

sponds to a noise-boosted activation function that evolves between the ReLU and linear activation functions. Moreover, this optimized activation function exhibits nonmonotonic transmission characteristics across the entire input domain, particularly within the negative input region. Hence, such noise-boosted activation function models exploit the stochastic resonance mechanism, introducing beneficial adaptations to neural network architectures in the domain of machine learning and enhancing their learning efficacy.

The negative portion of these activation functions within the subthreshold range does not contradict the firing rate of neurons. Within the framework of the noise-boosted activation model, the observed firing probability can be attributed to the introduction of noise that facilitates the activation of neurons even when their inputs fall within the subthreshold regime. Furthermore, negative activity at the presynaptic stage meaningfully represents an inhibitory hyperpolarization of the neuron membrane. This is incorporated into the established neuron model of Eq. (4), where negative activity reduces the firing probability of the postsynaptic neuron. An essential

feature is that the neural transfer functions, with their variability across hyperparameters, are obtained here uniformly via a process of addition of noise that we interpret as a stochastic resonance mechanism. This finding demonstrates that a whole range of useful neural transfer functions, currently exploited in machine-learning applications, can be obtained from simple two-state McCulloch-Pitts neurons engaged in a noise-aided structure where the added noise is appropriately optimized. This capability of control by noise of the response from nonlinear neurons, through a uniform mechanism of stochastic resonance, arguably carries nontrivial physical significance while bridging a gap between neuronal dynamics and machine learning [22–26,28,29].

ACKNOWLEDGMENTS

This work is supported by the Natural Science Foundation of Shandong Province of China (Grant No. ZR2021MF051) and the Taishan Scholar Project of Shandong Province (Grant No. TS20190930), China.

[1] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *J. Mach. Learn. Res.* **11**, 3371 (2010).
 [2] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, Contractive auto-encoders: Explicit invariance during feature extraction, in *Proceedings of the 28th International Conference on Machine Learning, Bellevue, WA*, edited by L. Getoor and T. Scheffer (Omnipress, St Madison, WI, 2011), pp. 833–840.
 [3] A. Camuto and M. Willetts, Variational autoencoders: A harmonic perspective, in *Proceedings of The 25th International*

Conference on Artificial Intelligence and Statistics, Valencia, Spain, edited by G. Camps-Valls, F. J. R. Ruiz, and I. Valera, Proceedings of Machine Learning Research (JMLR, Inc. and Microtome Publishing, Brookline, MA, USA, 2022), Vol. 151, pp. 4595–4611.
 [4] B. Poole, J. Sohl-Dickstein, and S. Ganguli, Analyzing noise in autoencoders and deep networks, [arXiv:1406.1831](https://arxiv.org/abs/1406.1831).
 [5] M. Ferienc, O. Bohdal, T. Hospedales, and M. Rodrigues, Impact of noise on calibration and generalisation of neural networks, in *The Fortieth International Conference on Machine Learning, The Second Workshop on Spurious Correlations, Invariance and Stability*, edited by A. Krause, E. Brunskill,

- K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett (ICML, Honolulu, Hawaii, USA, 2023).
- [6] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.* **15**, 1929 (2014).
- [7] Y. Wang, B. Xu, M. Kwak, and X. Zeng, A noise injection strategy for graph autoencoder training, *Neural Comput. Appl.* **33**, 4807 (2021).
- [8] S. Kariyappa, O. Dia, and M. K. Qureshi, Enabling inference privacy with adaptive noise injection, [arXiv:2104.02261](https://arxiv.org/abs/2104.02261).
- [9] M. Sabri and T. Kurita, Effect of additive noise for multi-layered perceptron with autoencoders, *IEICE Trans. Inf. Syst.* **E100.D**, 1494 (2017).
- [10] M. Lazzara, M. Chevalier, J. Garay-Garcia, C. Lapeyre, and O. Teste, Improving surrogate model prediction by noise injection into autoencoder latent space, in *IEEE 34th International Conference on Tools with Artificial Intelligence, Macao, China*, edited by N. G. Bourbakis, D. Zhang, M. Reformat, and W. Wang (IEEE, Piscataway, NJ, 2022), pp. 533–538.
- [11] Y. Grandvalet and S. Canu, Noise injection: Theoretical prospects, *Neural Comput.* **9**, 1093 (1997).
- [12] C. M. Bishop, Training with noise is equivalent to Tikhonov regularization, *Neural Comput.* **7**, 108 (1995).
- [13] S. Bai, F. Duan, F. Chapeau-Blondeau, and D. Abbott, Generalization of stochastic-resonance-based threshold networks with Tikhonov regularization, *Phys. Rev. E* **106**, L012101 (2022).
- [14] A. Orvieto, A. Raj, H. Kersting, and F. Bach, Explicit regularization in overparametrized models via noise injection, in *26th International Conference on Artificial Intelligence and Statistics*, edited by F. Ruiz and T. Geffner (Valencia, Spain, 2023), pp. 7265–7287.
- [15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, *Nature (London)* **323**, 533 (1986).
- [16] V. Nair and G. E. Hinton, Rectified linear units improve restricted Boltzmann machines, in *27th International Conference on International Conference on Machine Learning*, edited by J. Fürnkranz and T. Joachims (San Juan, Puerto Rico, 2010), pp. 807–814.
- [17] X. Glorot, A. Bordes, and Y. Bengio, Deep sparse rectifier neural networks, in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, edited by G. Gordon and D. Dunson (Ft. Lauderdale, FL, 2011), pp. 315–323.
- [18] M. B. Ghori, Y. Kang, and Y. Chen, Emergence of stochastic resonance in a two-compartment hippocampal pyramidal neuron model, *J. Comput. Neurosci.* **50**, 217 (2022).
- [19] S. A. Baccus and M. Meister, Fast and slow contrast adaptation in retinal circuitry, *Neuron* **36**, 909 (2002).
- [20] E. Real, H. Asari, T. Gollisch, and M. Meister, Neural circuit inference from function to structure, *Curr. Biol.* **27**, 189 (2017).
- [21] K. He, X. Zhang, S. Ren, and J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in *IEEE International Conference on Computer Vision, Santiago, Chile*, edited by R. Bajcsy, G. Hager, and Y. Ma (IEEE, Piscataway, NJ, 2015), pp. 1026–1034.
- [22] D. Hendrycks and K. Gimpel, Gaussian error linear units (GELUs), [arXiv:1606.08415](https://arxiv.org/abs/1606.08415).
- [23] S. Elfving, E. Uchibe, and K. Doya, Sigmoid-weighted linear units for neural network function approximation in reinforcement learning, *Neural Networks* **107**, 3 (2018).
- [24] P. Ramachandran, B. Zoph, and Q.V. Le, Searching for activation functions, [arXiv:1710.05941](https://arxiv.org/abs/1710.05941).
- [25] N. Ma, X. Zhang, M. Liu, and J. Sun, Activate or not: Learning customized activation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (IEEE, Nashville, TN, USA, 2021)*, pp. 8032–8042.
- [26] Y. Ying, J. Su, P. Shan, L. Miao, X. Wang, and S. Peng, Rectified exponential units for convolutional neural networks, *IEEE Access* **7**, 101633 (2019).
- [27] Z. Liao, K. Ma, S. Tang, H. Yamahara, M. Seki, and H. Tabata, Nonbistable rectified linear unit-based gain-dissipative Ising spin network with stochastic resonance effect, *J. Comput. Sci.* **62**, 101722 (2022).
- [28] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, A survey on modern trainable activation functions, *Neural Networks* **138**, 14 (2021).
- [29] S. R. Dubey, S.K. Singh, and B.B. Chaudhuri, Activation functions in deep learning: A comprehensive survey and benchmark, *Neurocomputing* **503**, 92 (2022).
- [30] N. G. Stocks, Suprathreshold stochastic resonance in multilevel threshold systems, *Phys. Rev. Lett.* **84**, 2310 (2000).
- [31] W. McCulloch and W. Pitts, A logical calculus of the ideas immanent in nervous activity, *Bull. Math. Biophys.* **5**, 115 (1943).
- [32] M. D. McDonnell, N. G. Stocks, C. E. M. Pearce, and D. Abbott, *Stochastic Resonance: From Suprathreshold Stochastic Resonance to Stochastic Signal Quantization* (Cambridge University, New York, 2008).
- [33] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Comput.* **9**, 1735 (1997).
- [34] <https://github.com/YuuhouRen/AutoencoderSR>.
- [35] R. A. Fisher, The use of multiple measurements in taxonomic problems, *Ann. Eugen.* **7**, 179 (1936).
- [36] J. Bergstra, R. Bardenet, Y. Bengio, and K. Balázs, Algorithms for hyper-parameter optimization, in *Proceedings of the 24th International Conference on Neural Information Processing Systems, Granada, Spain*, edited by J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger (Curran Associates, Inc., Red Hook, New York, 2011), pp. 2546–2554.
- [37] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, Optuna: A next-generation hyperparameter optimization framework, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, edited by A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis (Anchorage, Alaska, 2019), pp. 2623–2631.
- [38] W. Cukierski, Denoising dirty documents, <https://kaggle.com/competitions/denoising-dirty-documents>, 2015.