On Explainability of A Simple Classifier for AR(1) Source

Cem Benar and Ali N. Akansu
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
University Heights, Newark, NJ, 07102 USA
Email: cb427@njit.edu

***Abstract*—The heuristic reasoning and experiments based design approach have been the pillars of studies on artificial neural networks. The explainable network performance is required for most applications. We focus on a simple classifier network for the two-class case of AR(1) data sources. We trace the input statistics through the network and quantify changes to explain relationship between accuracy performance, optimized parameters and activation function types employed for the given architecture. We present test accuracy results for various network configurations with different dimension and activation types. AR(1) source model for a two-class case is utilized to generate training and test data sets of the experiments due to its ease of use for analytical study. We quantify the relationships with well known metrics among signal (class) statistics, network architecture, activation function type and accuracy for several correlation coefficient pairs of the two AR(1) sources utilized in this paper. It is observed from the experiments that the analyses of data, input-output relationships of hidden and output layer nodes for the given architecture provide invaluable insights and guidance to judiciously design a neural network and to explain its performance based on characteristics of the building blocks.**

***Index Terms*—Explainable neural network, activation function, pdf shaping, node compression ratio, layer compression ratio, AR(1) source.**

I. Introduction

Data intensive and computationally driven scientific discovery creates challenges for researchers to explain and to develop theoretical frameworks with repeatable outcomes. The recent progress in machine learning (ML) and artificial neural networks (ANN) is a good example of this paradigm shift. This study attempts to better understand and explain the interactions between the basic building blocks and performance of a simple learning network.

A typical network has its predefined architecture where input, hidden and output layers are interconnected together and its weight coefficients (parameters) are optimized based on an objective function related to the application of concern. We focus on classification problem in this study. The data driven numerical optimization of the parameters is performed during the *training* of the network model. Then, this set of optimized parameters is used to *test* and measure the network performance [1].

Since the neural network design methodology is heavily data driven and numerical in nature, the network performance is calculated for the widely accepted reference data sets. In this study, we are creating the *training* and the *test* data sets from auto-regressive order one, AR(1), source model that is commonly used as a coarse approximation to natural signals like images and speech due to its ease of analytical treatment [2], [3], [4].

Fig. 1. A single node of neural network. Activation function $g(.)$ maps input x to output y.

A simple classifier network with *one hidden layer* for a *two class problem* is introduced in Section II. We focus on and trace signals at various points of such a network and assess them by using well known metrics in signal processing and information theory. We highlight the impact of the *activation* (transfer) function on *input-output relationship* of a node in this section. AR(1) source model and its characteristics are given in Section III. Accuracy performance of simple neural network with respect to various architectures, activation function types, and auto-correlation coefficients, $\rho - pair$, of the input signal classes are presented and explained in Section IV. The remarks on our findings and conclusions are given in the last section of the paper.

II. A Simple Classifier Network

A single node of neural network is displayed in Fig. 1 with the input x. The activation function $g(.)$ of the node is nonlinear and the output is expressed as

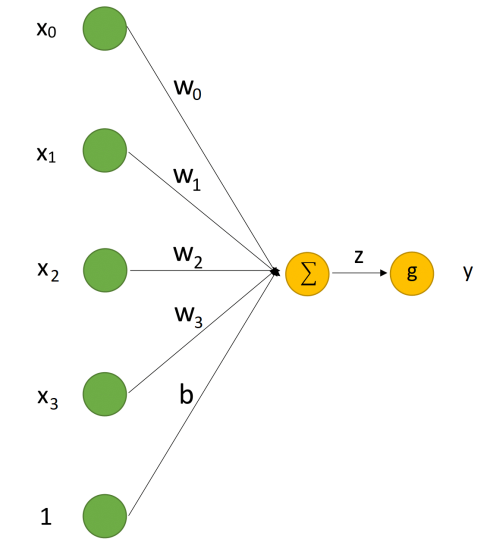
$$y = g(x) \tag{1}$$

Let us assume that the input and output *pdfs* of the node are $f_x(x)$ and $f_y(y)$, respectively, and $g(.)$ is a monotonic function. Then, the output *pdf* is calculated from the input as [5].

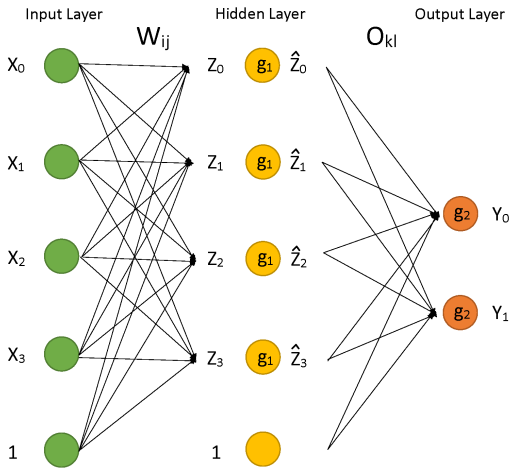
$$f_y(y) = \frac{f_x(x)}{|\frac{\partial y}{\partial x}|} \tag{2}$$

Note that we can calculate the (differential) entropy of a continuous information source with the *pdf* $f_x(x)$ as [6].

$$E(x) = -\int_{-\infty}^{+\infty} f_x(x)\, ln[f_x(x)]dx \tag{3}$$



(a) Perceptron Model.



(b) 4-4-2 Multilayer Perceptron (MLP).

Fig. 2. a) Perceptron model with inputs $\{x_k\}$, weight coefficients $\{w_k\}$, $0 \leq k \leq 3$ with bias term b and activation function $g(.)$. b) The single hidden layer, 4-4-2 classifier network for the two-class case.

The information entropy of a discrete source with N distinct symbols $\{x_i\}$ and symbol probabilities $\{p_i\}$ is calculated in *bits* as [6]

$$E_x = -\sum_{i=1}^{N} p_i log_2 p_i \tag{4}$$

We are interested in the *lossless entropy compression* properties of activation functions (with one-to-one invertible mapping properties) used in the hidden and output layers of the network with respect to input data statistics, architecture and the nature of classification problem as a set of *unique* features. Therefore, the *pdf* $f_x(x)$ of node input is calculated from the *pdfs* $\{f_{x_i}(x_i)\}$ of feeding node outputs of the previous layer weighted with coefficients $\{w_i\}$ and bias parameter b. The *Central Limit Theorem* (CLT) states that when statistically independent random variables $\{X_i,\ i = 1, 2, ..., N\}$ of any *pdf* type are added together, the resulting X tends to become a Gaussian random variable [5]. Note that the weight coefficients $\{w_i\}$ of a node input are real numbers, in general, with positive or negative signs. The *pdf* of node input is tracked, its input and output entropies are measured and traced *tomographically* for all hidden and output nodes of a neural network. This analysis sheds additional light to explain the relationship between network accuracy and its building blocks. This point is revisited in Section IV.

Fig. 2a displays the perceptron model with four inputs, $\{x_i,\ i = 0, 1, 2, 3\}$, weights coefficients $\{w_i,\ i = 0, 1, 2, 3\}$ and bias coefficient b where the input of the activation function $g(.)$ is calculated as $x = \sum_{i=0}^{3} w_i x_i + b$. A single hidden layer neural network with four input, four hidden layer and two output nodes, 4-4-2 classifier network, is displayed in Fig. 2b. The sets of weight coefficients $\{w_{i,j},\ i, j = 0, 1, 2, 3\}$ and $\{o_{k,l},\ k = 0, 1, 2, 3\, l = 0, 1\}$ for the hidden and output layers (with the preselected two sets of activation functions) of this 4-4-2 classifier network, respectively, are optimized based on stochastic gradient descent algorithm through the training step to build the model. Then, its accuracy is tested for various two-class data types.

A. Activation Functions

One needs to know the joint and marginal probabilities of variables in a random vector process for its proper statistical representation [5]. It is a cumbersome if not an impossible task even for relatively low dimensional data types. On the other hand, there have been a plethora of research activities with mostly heuristic and data driven methods yielding promising performance to extract higher order statistical characteristics of different signal (data) types in various application domains. One of the commonly agreed weaknesses of the state-of-the-art learning networks is their inability "to extract and organize the discriminating information from the data" [7]. The nonlinear activation function (one-to-one *invertible* mapping) of a network node has been hypothesized to exploit higher order statistics to extract some latent information among the signal classes under study. The activation function is very similar to the *companding* operator used in the early years of telephony. Indeed, it may be considered as the special version of *pdf-optimized* non-uniform quantizer (a *lossless compression* method when the number of bins goes to infinity and the output range is reduced to 0 to 1 or -1 to +1). It is a mature subject in information theory [2], [4], [6], [8], [9]. The design and selection of optimal activation (transfer) function and its merit in a neural network is an active research topic and beyond the scope of this paper [10], [11].

We use Sigmoid, Hyperbolic Tangent (Tanh) and ReLU functions in the paper for performance analysis of a simple classifier network. These functions are defined in Table I. Note that the first two provide one-to-one *unique* (lossless) mappings between the input and output while ReLU does not have that feature. We quantify the impact of activation functions in nodes of the network with *entropy* and *correlation* measurements and to relate them with accuracy presented in Section IV.

Table I: Sigmoid, Tanh, and ReLU activation functions.

Sigmoid	*Tanh*	*ReLU*
$\frac{1}{1+e^{-x}}$	$\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$	$\max(0,x)$

III. Auto-Regressive Order One Source Model

Auto-regressive discrete process with order one, AR(1), provides coarse approximation to natural signals like images and widely used in signal processing research. AR(1) signal source is modeled as [4]

$$x\left(n\right)=\rho x\left(n-1\right)+\xi\left(n\right) \tag{5}$$

$\xi(n)$ is Gaussian with zero-mean and variance σ_{ξ}^{2}, $E\left\{\xi(n)\xi(n+k)\right\}=\sigma_{\xi}^{2}\delta_{n-k}$ where $-1<\rho<1$. The correlation coefficient is defined as

$$\begin{aligned}\rho&=R_{xx}\left(1\right)/R_{xx}\left(0\right)\\&=\frac{E\left\{x\left(n\right)x\left(n+1\right)\right\}}{E\left\{x\left(n\right)x\left(n\right)\right\}}\end{aligned} \tag{6}$$

The signal variance is expressed as $\sigma_{x}^{2}=\frac{\sigma_{\xi}^{2}}{(1-\rho^{2})}$. The auto-correlation sequence of AR(1) process is written as

$$R_{xx}(k)=E\left\{x(n)x(n+k)\right\}=\sigma_{x}^{2}\rho^{|k|};\ k=0,\pm1,\pm2,\ldots \tag{7}$$

Note that AR(1) signal becomes Gaussian white noise when $\rho=0$.

We used AR(1) model to generate the training and test data of various signal classes for several classification problem scenarios as explained in Section IV. We prefer to use this simple model due to its ease of use in analytical formulations and modeling. We generated two sets of *training* and *test data* in one-byte per sample resolution from AR(1) source model for two different classes defined by their first order correlation coefficients ρ_0 and ρ_1, respectively. Labels for each class were created and the data sets and labels were randomly shuffled. Each dataset is normalized to zero mean and unit variance per dimension.

IV. Performance

A. Input-Output Relationship of a Node

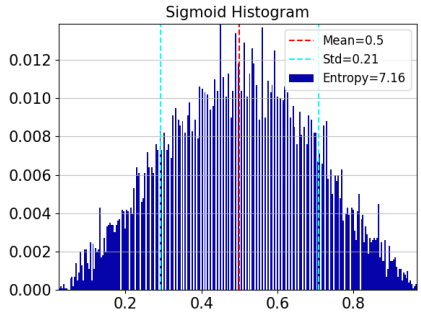
The mathematical relationship between the input and output of a node displayed in Fig. 1 is expressed in Eqs. (1) and (2) with respect to their *pdfs* and the activation function $g(.)$. The input and output *information entropies* of a node, E_{in} and E_{out}, respectively, with Sigmoid, Tanh and ReLU activation functions for discrete AR(1) inputs of various ρ values are tabulated in Table II.

The output histograms of a node with the three activation functions and information entropies for discrete AR(1) input with $\rho=0.9$ are displayed in Fig. 3.

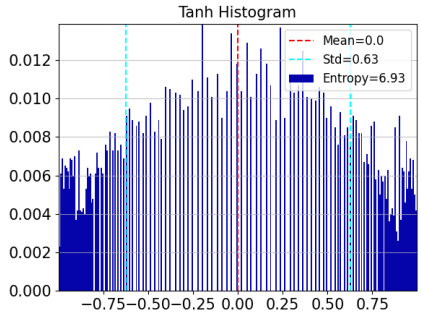
Similarly, input and output information entropies of four hidden layer nodes with Sigmoid activations in the 4-4-2 network shown in Fig. 2b are tabulated in Table III for discrete AR(1) inputs of various $\rho_0-\rho_1$ pairs. The input and output histograms of the two hidden layer nodes (N_0^H and N_3^H) and

Table II: Input and output *information entropies* (E_{in}, E_{out}) of a node with different activation functions for discrete AR(1) inputs of various correlation coefficients ρ.

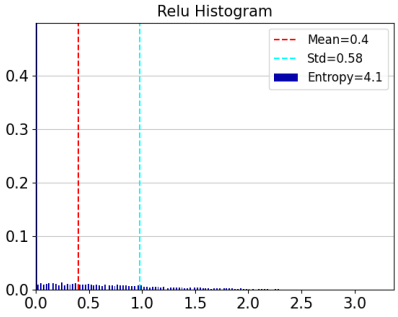
ρ		*Sigmoid*	*Tanh*	*ReLU*
0	E_{in}	7.18	7.18	7.18
	E_{out}	7.14	6.92	4.02
0.5	E_{in}	7.04	7.04	7.04
	E_{out}	7.01	6.82	4.02
0.9	E_{in}	7.2	7.2	7.2
	E_{out}	7.16	6.93	4.1



(a) Sigmoid



(b) Tanh



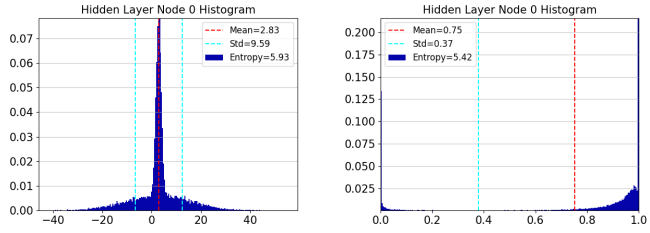
(c) ReLU

Fig. 3. The output histograms of a node with the three activation functions and information entropies for discrete AR(1) input with $\rho=0.9$.

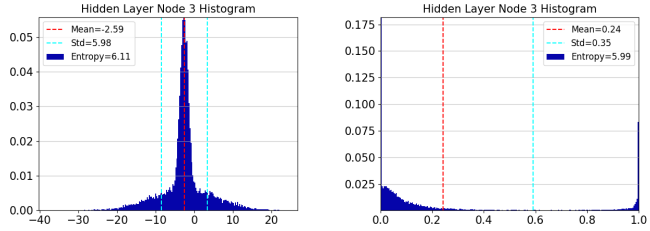
two output layer nodes (N_0^O and N_1^O) in the same network with Sigmoid activation for AR(1) input data classes of $\rho_0 = 0$ and $\rho_1 = 0.99$ are displayed in Figs. 4 and 5, respectively.

Table III: Input and output entropies (E_{in}, E_{out}) of four hidden layer nodes in the 4-4-2 *trained* network with Sigmoid activations for $\rho_0 - \rho_1$ pair correlation coefficients in the two-class case.

ρ_0	ρ_1		N_0^H	N_1^H	N_2^H	N_3^H
0	0.99	E_{in}	5.93	6.06	6.06	6.11
		E_{out}	5.42	5.76	5.71	5.99
0.4	0.99	E_{in}	6.08	6.05	7.04	7.04
		E_{out}	5.43	5.31	2.93	5.4
0.7	0.99	E_{in}	6.33	6.54	6.32	6.44
		E_{out}	6.02	6.08	6.19	6.09
0.85	0.95	E_{in}	6.82	6.8	7.1	6.67
		E_{out}	6.27	5.98	5.26	6.27
0.75	0.85	E_{in}	6.92	6.96	6.92	6.95
		E_{out}	5.67	5.8	5.91	6.27



(a) $N_0^H Input$ (b) $N_0^H Output$



(c) $N_3^H Input$ (d) $N_3^H Output$

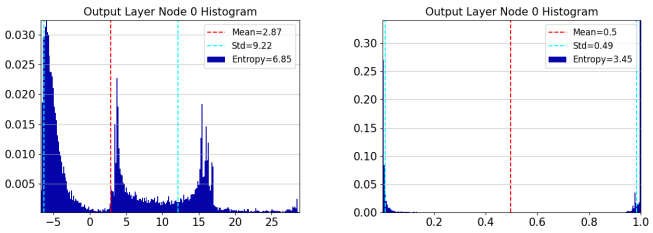
Fig. 4. Input and output histograms of N_0^H and N_3^H in the 4-4-2 trained classifier network with Sigmoid activations for the two-class experiment with correlation coefficients of $\rho_0 = 0$ and $\rho_1 = 0.99$.

It is observed from Table III that the nonlinear and *invertible* mapping of the Sigmoid function results in *lossless* entropy compression. Figs. 4 and 5 clearly show the impact of activation functions on the *pdf shapes* at the node outputs. The *bimodal* nature of the node output pdfs might be highly related to accuracy of the classifier. This point requires further study.

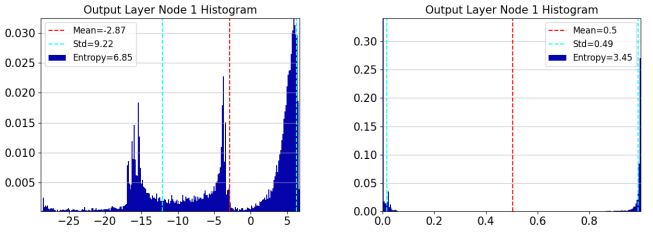
We define the metrics called the *node compression ratio* (NCR),

$$\eta_{E_i} = E_{in}^i / E_{out}^i \tag{8}$$

for node i, and the *layer compression ratio* (LCR)



(a) $N_0^O Input$ (b) $N_0^O Output$



(c) $N_1^O Input$ (d) $N_1^O Output$

Fig. 5. Input and output histograms of N_0^O and N_1^O in the 4-4-2 trained classifier network with Sigmoid activations for the two-class experiment with correlation coefficients of $\rho_0 = 0$ and $\rho_1 = 0.99$.

$$\eta_E = \frac{1}{N} \sum_{i=0}^{N-1} \eta_{E_i} \tag{9}$$

for the given hidden (η_E^H) or output (η_E^O) layer. We repeated the same experiment for 16-16-2 and 64-64-2 classifier networks for various $\rho_0 - \rho_1$ pairs with Sigmoid and Tanh activations. The *layer compression ratios* of the hidden and output layers along with the accuracy measurements are tabulated in Tables IV and V. It is seen from the tables that the output *layer compression ratio* η_E^O gets larger as the number of input and hidden nodes (N, K) of the network increases. It is also observed that η_E^O and accuracy are correlated. In contrast, the hidden *layer compression ratio* η_E^H and accuracy are inversely correlated. The results show that the accuracy is related to the level of class correlations as well as their differences as explained below.

B. Accuracy of Simple Classifier Network

We calculate the accuracy of the 4-4-2 neural network depicted in Fig. 2b for a two-class AR(1) data classification problem with Sigmoid activation functions in the hidden nodes. Note that we also used Sigmoid in the output nodes for all test scenarios presented in the paper. The classes are defined by two different correlation coefficients ρ_1 and ρ_2 of AR(1) source model. We used the same ρ_1 and ρ_2 to generate the training and test data sets of an experiment. Each class has 10,000 training and 2,000 test data vectors. Each data vector is N-dimensional.

The accuracy results of the 4-4-2, 16-16-2 and 64-64-2 topologies with Sigmoid activation function for the training and test data sets are tabulated in Table VI.

Table IV: Compression ratios of Sigmoid function in the hidden and output layers (η_E^H, η_E^O) and test accuracies for AR(1) data sets of various $\rho_1 - \rho_2$ pairs in the 4-4-2, 16-16-2, and 64-64-2 networks. N and K are the number of nodes in the input and hidden layers, respectively.

(a) The results of the 4-4-2 network.

ρ_0	ρ_1	N	K	η_E^H	η_E^O	Test Acc
0	0.99	4	4	1.07	1.71	97.55
0.3	0.99	4	4	1.05	1.47	96.75
0.6	0.99	4	4	1.06	1.28	95
0.9	0.99	4	4	1.13	0.96	84.22
0.85	0.95	4	4	1.18	0.88	69.78
0.75	0.85	4	4	1.17	0.92	60.38
0.65	0.75	4	4	1.26	0.96	57.38

(b) The results of the 16-16-2 network.

ρ_0	ρ_1	N	K	η_E^H	η_E^O	Test Acc
0	0.99	16	16	1.05	3.25	99.5
0.3	0.99	16	16	1.07	3.35	99.88
0.6	0.99	16	16	1.12	3.32	99.85
0.9	0.99	16	16	1.38	2.62	98.65
0.85	0.95	16	16	1.63	1.06	86.42
0.75	0.85	16	16	1.71	0.88	68.4
0.65	0.75	16	16	1.66	0.89	63.3

(c) The results of the 64-64-2 network.

ρ_0	ρ_1	N	K	η_E^H	η_E^O	Test Acc
0	0.99	64	64	1.25	4.13	99.9
0.3	0.99	64	64	1.23	4.24	99.98
0.6	0.99	64	64	1.29	3.97	99.9
0.9	0.99	64	64	1.26	3.56	99.7
0.85	0.95	64	64	1.57	2.65	92.95
0.75	0.85	64	64	2.08	1.79	71.68
0.65	0.75	64	64	2.27	1.55	57.22

Table V: Compression ratio of Tanh function in the hidden layer (η_E^H) and compression ratio of Sigmoid function in the output layer (η_E^O) and test accuracies for AR(1) data sets of various $\rho_1 - \rho_2$ pairs in the 4-4-2, 16-16-2, and 64-64-2 networks. N and K are the number of nodes in the input and hidden layers, respectively.

(a) The results of the 4-4-2 network.

ρ_0	ρ_1	N	K	η_E^H	η_E^O	Test Acc
0	0.99	4	4	1.32	1.6	97.52
0.3	0.99	4	4	1.42	2.24	97.65
0.6	0.99	4	4	1.44	1.19	94.68
0.9	0.99	4	4	1.52	0.95	84.12
0.85	0.95	4	4	1.57	0.9	69.3
0.75	0.85	4	4	1.5	0.94	61.9
0.65	0.75	4	4	1.28	0.98	57.3

(b) The results of the 16-16-2 network.

ρ_0	ρ_1	N	K	η_E^H	η_E^O	Test Acc
0	0.99	16	16	1.36	4.34	99.88
0.3	0.99	16	16	1.51	4	99.98
0.6	0.99	16	16	1.47	3.71	99.9
0.9	0.99	16	16	4.38	2.37	96.12
0.85	0.95	16	16	2.68	1.11	85.6
0.75	0.85	16	16	2.43	0.88	65.97
0.65	0.75	16	16	2.54	0.91	59.4

(c) The results of the 64-64-2 network.

ρ_0	ρ_1	N	K	η_E^H	η_E^O	Test Acc
0	0.99	64	64	1.98	3.87	99.62
0.3	0.99	64	64	1.91	3.7	99.78
0.6	0.99	64	64	1.88	4.15	99.78
0.9	0.99	64	64	2.75	3.68	99.1
0.85	0.95	64	64	2.31	2.35	90.5
0.75	0.85	64	64	3.27	1.51	65.6
0.65	0.75	64	64	3.35	1.42	54.18

We observe from Tables VIa, VIb and VIc that accuracies increase as the difference $\Delta\rho = \rho_1 - \rho_0$ increases. We also observe that the accuracies decrease as ρ_0 and ρ_1 values get smaller even when $\Delta\rho$ is constant. This is due to the smaller signal to noise ratio of AR(1) source model, $SNR = 1/(1-\rho)^2$ for smaller ρ values [4]. It is seen that higher dimension improves accuracy when both data classes are in the higher SNR range for the N values used in this paper. The impact of $\Delta\rho$ and SNR of the two classes on classifier accuracy deserves further analytical study. We had observations with similar trends for the test scenarios where Sigmoid activations are replaced by Tanh or ReLU functions in the hidden layers of the networks.

V. Remarks and Conclusions

The following remarks are made based on the accuracy results and our observations for AR(1) signal source based experiments with two classes presented above.

a. The nonlinear activation function of a node is a one-to-one *invertible mapping* and reshapes the statistics of the input signal with different pdf, correlation and entropy properties at the output. It is a *lossless entropy compression* method. We introduced the *node compression ratio* (NCR) as the metric to quantify the information processing characteristics of hidden ($\eta_{E_i}^H$) and the output layer nodes ($\eta_{E_i}^O$) for the given scenario. Similarly, we defined the *layer compression ratio* (LCR) as the average $\eta_E = \frac{1}{N}\sum_{i=0}^{N-1}\eta_{E_i}$ for the given hidden H_k or output O_l layer. It is observed from Tables IV and V that the *accuracy* and the *layer compression ratio* of the output layer have *correlation*. In contrast, the *layer compression ratio* of the hidden layer has *inverse correlation* with classifier accuracy. We found this relationship quite interesting although it needs further study since it is at the core of the network optimization.

We observed that the impact of each node is different in the overall network performance. It is noted that $\eta_{E_i}^H$ might become quite large when the input histogram and the activation function pair yields significantly reduced output energy for some nodes. Hence, a proper threshold to ignore such extreme $\eta_{E_i}^H$ values in the calculation of η_E^H might be needed. The

Table VI: Accuracies of the 4-4-2, 16-16-2 and 64-64-2 classifier networks for the training and test data sets of various $\rho_0 - \rho_1$ pairs with Sigmoid function. N and K are the number of nodes in the input and hidden layers, respectively.

(a) The accuracies of the 4-4-2 network for various $\rho_0 - \rho_1$ pairs.

ρ_0	ρ_1	N	K	Train Acc	Test Acc
0.1	0.99	4	4	97.32	97.68
0.3	0.99	4	4	96.8	96.75
0.5	0.99	4	4	95.9	95.8
0.7	0.99	4	4	96.4	96.35
0.9	0.99	4	4	84.52	84.22
0.85	0.95	4	4	69.18	69.78
0.65	0.75	4	4	56.54	57.38
0.45	0.55	4	4	54.31	55.75
0.25	0.35	4	4	52.81	54.1
0.05	0.15	4	4	52.21	51.5

(b) The accuracies of the 16-16-2 network for various $\rho_0 - \rho_1$ pairs.

ρ_0	ρ_1	N	K	Train Acc	Test Acc
0.1	0.99	16	16	99.98	99.95
0.3	0.99	16	16	100	99.88
0.5	0.99	16	16	100	99.9
0.7	0.99	16	16	99.99	99.85
0.9	0.99	16	16	99.24	98.65
0.85	0.95	16	16	88.96	86.42
0.65	0.75	16	16	67.15	63.3
0.45	0.55	16	16	61.74	57.12
0.25	0.35	16	16	57.99	54.2
0.05	0.15	16	16	57.94	53.08

(c) The accuracies of the 64-64-2 network for various $\rho_0 - \rho_1$ pairs.

ρ_0	ρ_1	N	K	Train Acc	Test Acc
0.1	0.99	64	64	100	100
0.3	0.99	64	64	100	99.98
0.5	0.99	64	64	100	99.82
0.7	0.99	64	64	100	99.88
0.9	0.99	64	64	99.96	99.7
0.85	0.95	64	64	99.16	92.95
0.65	0.75	64	64	89.62	57.22
0.45	0.55	64	64	87.42	53.78
0.25	0.35	64	64	87.15	50.55
0.05	0.15	64	64	87.48	49.62

impacts of the number of hidden layers (network depth) and nodes (layer dimensions) on the output node characteristics and accuracy of the network are currently being studied by the authors.

b. The accuracy of the classifier is related to the SNR of each class data as well as the difference of their correlation coefficients $\Delta\rho = \rho_1 - \rho_0$. It suggests that we may utilize more sophisticated correlation model than the exponential correlation function of AR(1) source for better representation of real world signals like images.

c. The performance is related to the dimension N and the number of hidden layers. Therefore, the topology of the network has strong impact on the optimization and accuracy [1], [7]. We were not able to include the multiple hidden layer network architecture examples and our observations in this paper due to the availability of limited space.

One needs to study the available data to model its statistical properties prior to designing the network for the given task. The statistical signal models are good tools to methodically assess and improve network performance. The theoretical relationships between classifier accuracy, SNR and $\Delta\rho$ using various network architectures and activation types for various data types are of great interest and actively pursued by many researchers in the field.

REFERENCES

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016, http://www.deeplearningbook.org.

[2] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video.*, Prentice Hall Professional Technical Reference, 1984.

[3] R. J. Clarke, *Transform Coding of Images.*, Academic Press, Inc., 1985.

[4] A. N. Akansu and R. A. Haddad, *Multiresolution Signal Decomposition: Transforms, Subbands, and Wavelets.*, Academic Press, Inc., 1992.

[5] A. Papoulis, *Probability, Random Variables and Stochastic Processes, 2nd Ed.*, McGraw-Hill, 1984.

[6] T. Berger, *Rate-Distortion Theory.*, John Wiley and Sons, Inc., 2003.

[7] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[8] J. Max, "Quantizing for minimum distortion," *Information Theory, IRE Trans. on*, vol. 6, no. 1, pp. 7–12, Mar. 1960.

[9] S. Lloyd, "Least squares quantization in PCM," *Information Theory, IEEE Trans. on*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[10] A. Apicella, F. Donnarumma, F. Isgro, and R. Prevete, "A survey on modern trainable activation functions," *Neural Networks (Elsevier)*, vol. 138, pp. 14–32, June 2021.

[11] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "A comprehensive survey and performance analysis of activation functions in deep learning," *arXiv preprint arXiv:2109.14545*, 2021.