Universal Consistency of Deep Convolutional Neural Networks

Shao-Bo Lin, Kaidong Wang, Yao Wang, and Ding-Xuan Zhou

Abstract

Compared with avid research activities of deep convolutional neural networks (DCNNs) in practice, the study of theoretical behaviors of DCNNs lags heavily behind. In particular, the universal consistency of DCNNs remains open. In this paper, we prove that implementing empirical risk minimization on DCNNs with expansive convolution (with zero-padding) is strongly universally consistent. Motivated by the universal consistency, we conduct a series of experiments to show that without any fully connected layers, DCNNs with expansive convolution perform not worse than the widely used deep neural networks with hybrid structure containing contracting (without zero-padding) convolutional layers and several fully connected layers.

Index Terms

Deep learning, convolutional neural networks, universal consistency

I. Introduction

The great success of deep learning [8] in practice stimulates avid research activities to understand the magic behind it. The reasons for success can be attributed to the depth of networks [17], [27], massiveness of data [20], [6], fast developed optimization algorithms [24], [1] and more importantly, architectures [5], [15] that reduce the number of free parameters of networks while maintaining their excellent performances in feature extraction and function representations. Deep convolutional neural networks (DCNNs) that equip deep neural networks with convolutional structures are one of the most popular networks used in image processing [18], game theory [28], signal processing [16], among many others.

We are interested in DCNNs induced by one-dimensional convolution, one channel and the rectifier linear unit (ReLU) activation function. As in [31], there are not any fully connected layers in DCNNs considered in this paper. The convolution of two functions f and h on $\mathbb{R}$ is defined by

$$h \otimes f(x) = \int_{-\infty}^{\infty} f(x')h(x-x')dx', \qquad x \in \mathbb{R}.$$

Discretely, let $\vec{w} = (w_j)_{j=-\infty}^{\infty}$ be a filter of of length s, i.e. $w_j^k \neq 0$ only for $0 \leq j \leq s$. Two widely used types of 1-D convolution of $\vec{w}$ with a vector $\vec{v} = (v_1, \dots, v_D)^T$, regarded as a sequence on $\mathbb{Z}$ supported in $\{1, \dots, D\}$, are the expansive convolution (also called convolution with zero-padding) denoted by $\vec{w} * \vec{v}$ and contracting convolution (or convolution without zero-padding) $\vec{w} \star \vec{v}$ with j-th components

$$(\vec{w} * \vec{v})_j = \sum_{\ell=1}^{D} w_{j-\ell} v_\ell, \qquad j = 1, \dots, D+s, \tag{1}$$

and

$$(\vec{w} \star \vec{v})_j = \sum_{\ell=j-s}^{j} w_{j-\ell} v_\ell, \qquad j = s+1, \dots, D. \tag{2}$$

From (1) and (2), it is easy to derive [31] that there are $D \times (D+s)$ sparse Toeplitz type matrix $\widetilde{W}$ and $D \times (D-s)$ one $\widetilde{W'}$ such that

$$\vec{w} * \vec{v} = \widetilde{W}\vec{v}, \qquad \text{and} \quad \vec{w} \star \vec{v} = \widetilde{W}'\vec{v}, \qquad \forall \vec{v} \in \mathbb{R}^D. \tag{3}$$

Let $\sigma(t) = \max\{0, t\}$ be ReLU and $L \in \mathbb{N}$ be the number of hidden layers. Given a set of filters $\{\vec{w}_k\}_{k=1}^L$, a set of bias vectors $\{\vec{b}_k\}_{k=1}^L$ of compatible sizes and a vector $\vec{a}_L$, the DCNN can be defined by

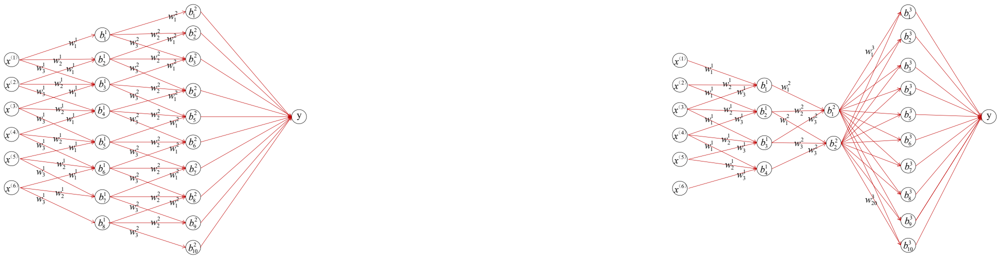
$$h_L(x) = \hat{a}_L \cdot \vec{h}_L(x), \tag{4}$$

where $\vec{h}_0(x) = x$,

$$\vec{h}_k(x) = \sigma(\vec{w}_k \odot \vec{h}_{k-1}(x) + \vec{b}_k), \qquad k = 1, \dots, L, \tag{5}$$

σ acts on vectors componentwise and $\odot$ denotes either $*$ in (1) or $\star$ in (2). Due to (3), DCNNs can be regarded as special deep fully connected neural networks with specified sparse comvolutional structures imposed to weight matrices. Figure 1 shows structures of the mentioned two types of DCNNs.

S. B. Lin, K. Wang and Y. Wang are with the Center for Intelligent Decision-Making and Machine Learning, School of Management, Xi'an Jiaotong University, Xi'an 710049, P R China. D. X. Zhou is with School of Data Science and Department of Mathematics, City University of Hong Kong, Hong Kong. The corresponding author is Y. Wang (email: yao.s.wang@gmail.com).



(a) expansive DCNN

(b) contracted DCNN

Fig. 1: The left shows the expansive DCNN with $s = 2$ and $L = 2$, while the right exhibits a hybrid deep net with the contracted DCNN with $s = 2$ and $L = 2$ and a fully connected neural network of width 10.

Practically, the contracting DCNN (cDCNN) is more commonly used than the expansive DCNN (eDCNN) by utilizing the convolutional layers to extract features of data. However, it can be found in [13, Theorem 1] (see also [12]) that to guarantee the universal approximation property of deep ReLU nets, there exists at least one hidden layer whose width is larger than $d + 1$. Noting further that the maximum width of hidden layers in cDCNN is d, smaller than $d + 1$, cDCNN is thus not a universal approximant. Though this bottleneck of cDCNN can be broken through by equipping the network with additional fully connected layers, it brings new challenges on determining the depth and width of fully connected layers. Differently, due to the zero-padding mechanism, the widths of all hidden layers in eDCNN are not smaller than $d+1$. Under this circumstance, the universal approximation property of eDCNNs has recently been verified in [31], showing that an eDCNN without any fully connected layers can approximate any continuous functions to an arbitrary accuracy, provided there are sufficiently many layers.

From universal approximation to universal consistency in learning, the price for approximation, measured by the capacity of the family of approximants, should be taken into account. In particular, it can be found in [22] that there exists a set of deep nets with two hidden layers and a bounded sigmoid function possessing universal approximation property but is not a universal consistent learner, since the capacity (measured by the pseudo-dimension) of this set is infinite. A similar problem seems to exist for eDCNN, since the constructed network in [31] for universal approximation involves unbounded weights. Generally speaking, unboundedness of parameters leads to extremely large capacity of deep nets, especially for deep nets with sigmoid activation functions [21], [22], [2], which makes the universal approximation property established in [31] be not applicable for the learning purpose at the first glance (see the discussion in [26, Sec. 2] for example). Surprisingly , due to the piecewise linear property of ReLU, a tight pseudo-dimension estimate of deep ReLU nets without any restrictions on free parameters has been derived in [4]. With this, we use a classical relation [14], [25] between the pseudo-dimension and covering number and then succeed in deriving the universal consistency for implementing empirical risk minimization (ERM) on eDCNNs.

Since we are concerned with the universal consistency of eDCNN, seems to be no hint on advantages of eDCNN in our theory, as it has already been demonstrated in [11] and [2] that both shallow nets and deep nets with fixed widths are universally consistent, respectively. However, as discussed in [30], [31], [32], [23], a main advantage of eDCNNs is their good approximation capability in tackling high dimensional data. To be detailed, to approximate a function $f \in C^r([0,1]^d))$, r-th smooth functions defined on $[0,1]^d$ with $r \approx d/2$, within accuracy ε, deep fully connected nets (DFCNs) [29] need $2^d\varepsilon^{-2}$ free parameters paved on $\frac{c_0 d}{4}\left(\log\frac{1}{\varepsilon}+d\right)$ hidden layers, while eDCNN [31] requires at most $\frac{75d}{\varepsilon^2}\log\frac{1}{\varepsilon^2}$ free parameters paved on $\frac{4}{\varepsilon}\log\frac{1}{\varepsilon}$ hidden layers. The independence of the required number of hidden layers on the dimension illustrates evidence for the power of eDCNNs. Furthermore, given fixed approximation accuracy ε, eDCNNs performs excellently in approximating high dimensional functions, since the required number of free parameters is linear with respect to d for eDCNN while exponential with respect to d for DFCNs. We refer the readers to [31, Section 2] for more details of the outperformance of eDCNNs over shallow nets and DFCNs. We highlight that there are not any exponential factors of d involved in our proof, the mentioned dimension-independent property of eDCNN also holds for the learning purpose. Different from cDCNNs, it should be mentioned that the width and number of free parameters in eDCNNs increase linearly with the depth. We can use either a threshold-sharing strategy in [31] or a pooling-type down-sampling strategy [32] to reduce them, while the universal approximation and universal consistency properties still hold. We also conduct a series of numerical experiments to verify our theoretical assertions and show the excellent learning performance of eDCNNs in real applications like human activity recognition and heartbeat classification. Our numerical results show that with a unified simple structure, eDCNNs perform not worse than cDCNNs equipped with fully connected layers with different widths and depths.

II. Universal Consistency of eDCNN

In learning theory [7], [9], the samples in the data set $D := \{z_i\}_{i=1}^m := \{(x_i, y_i)\}_{i=1}^m$ are assumed to be drawn independently and identically from an unknown Borel probability distribution ρ on $Z := \mathcal{X} \times \mathcal{Y}$, where $x_i \in \mathcal{X} \subseteq \mathbb{R}^d$, and $y_i \in \mathcal{Y} \subseteq R$. Throughout the paper, we assume $\mathcal{X}$ is a compact set. The aim is to learn a function f_D based on D to minimize the generalization error

$$\mathcal{E}(f) := \int_{\mathcal{Z}} (f(x) - y)^2 d\rho.$$

Noting that the regression function $f_\rho(x) := \int_{\mathcal{Y}} y d\rho(y|x)$ defined by means of the conditional distributions $\rho(\cdot|x)$ of ρ at $x \in \mathcal{X}$ minimizes the generalization error, our aim is then to find an estimator f_D to minimize

$$\mathcal{E}(f) - \mathcal{E}(f_\rho) = \|f - f_\rho\|^2_{L^2_{\rho_X}}, \tag{6}$$

where ρ_X is the marginal distribution of ρ on $\mathcal{X}$.

We build up the estimator via empirical risk minimization (ERM):

$$f_{D,L,s} := \arg\min_{f \in \mathcal{H}_{L,s}} \mathcal{E}_D(f), \tag{7}$$

where $\mathcal{E}_D(f) = \frac{1}{m}\sum_{i=1}^m (f(x_i) - y_i)^2$ denotes the empirical risk of f and

$$\mathcal{H}_{L,s} := \left\{h_L(x) : \vec{w}_k \text{ is of length } s,\ \vec{b}_k \in \mathbb{R}^{d+ks}, \hat{a}_L \in \mathbb{R}^{d+Ls}, k = 1, \dots, L\right\} \tag{8}$$

be the set of all output functions produced by the eDCNN defined by (5) with $\odot = *$. One of the most important properties that a learner should have is that, as the sample size m grows, the deduced estimator converges to the real relation between the input and output. This property, featured as the strongly universal consistency [11], can be defined as follows.

Definition 1: A sequence of regression estimators $\{f_m\}_{m=1}^\infty$ is called strongly universally consistent, if

$$\lim_{m\to\infty} \mathcal{E}(f_m) - \mathcal{E}(f_\rho) = 0$$

holds with probability one for all Borel probability distributions ρ satisfying $\int_{\mathcal{Y}} y^2 d\rho(y|x) < \infty$.

Our main result is the following theorem, which shows that running ERM on eDCNN yields strongly universally consistent learners.

Theorem 1: Let $\theta \in (0, 1/2)$ be an arbitrary real number and $2 \le s \le d$. If $L = L_m \to \infty$, $M = M_m \to \infty$, $M_m^2 m^{-\theta} \to 0$ and

$$\frac{M_m^4 L_m^2 (L_m + d) \log L_m \log(M_m m)}{m^{1-2\theta}} \to 0, \tag{9}$$

then $\pi_{M_m} f_{D,L_m,s}$ is strongly universally consistent, where $\pi_M t = \min\{M, |t|\} \cdot \mathrm{sgn}(t)$ is the well known truncation operator.

Remark 1: There is a truncation operator involved in Theorem 1. It should be mentioned that such a truncation operator is essential. The reasons are two folds. On the one hand, since we do not impose any boundedness restrictions on the free parameters, it is obvious that functions in $\mathcal{H}_{L,s}$ are not uniformly bounded. Under this circumstance, it is difficult to derive a covering number estimate for $\mathcal{H}_{L,s}$ since the relation between covering numbers and pseudo-dimension is built upon bounded functions [14]. On the other hand, expect for $\int_{\mathcal{Y}} y^2 d\rho(y|x) < \infty$, there is no any other restriction on y, making the analysis difficult. A preferable way is to consider a truncation operator on y, $y_{M_m} := \pi_{M_m} y$, with $M_m \to \infty$, which corresponds a truncation operator on $f_{D,L,s}$. Such a truncation operator is widely adopted in demonstrating the universal consistency for numerous learning schemes [11], including local average regression, linear least squares, and shallow neural networks learning.

Remark 2: There are totally four conditions involved in Theorem 1 on the depth $L = L_m$ and truncation value $M = M_m$ to guarantee the universal consistency of eDCNN: 1) $L_m \to \infty$; 2) $M_m \to \infty$; 3) $M_m^2 m^{-\theta} \to 0$; 4) limit (9). The first restriction is natural since $L_m \to \infty$ is necessary for the universal approximation of eDCNN. The second one is also mild since there is no any boundedness assumption on y. It is impossible to derive a bounded learner that can learn unbounded samples well. The third one, restricting the growth of M_m with respect to m, is also widely used in the literature [11], since the sample error frequently depends on $g_1(M_m)g_2(m^{-1})$ for some increasing univariate functions g_1 and g_2. Therefore, it is necessary to tailor M_m such that $g_1(M_m)g_2(m^{-1}) \to 0$. The last one is technical, which focuses on the relation among M_m, L_m and m. It presents a guidance of selecting M_m and L_m in eDCNNs learning. In particular, $M_m = \log m$ and $L_m = m^\alpha$ with $\alpha < 1/3$ satisfy all these assumptions and can yield strongly universally consistent eDCNN estimator.

Remark 3: Theorem 1 only considers eDCNN with one dimensional convolution. It would be interesting to derive similar results for eDCNN with two dimensional convolution, which is widely used in practice. Different from 1-d eDCNN that corresponds to Toeplitz type weight matrices, the structures of weight matrices in 2-d eDCNN are much more sophisticated, making the existing convolutional factorization [30], [31] infeasible and the universal approximation property of 2-d eDCNN remains open. The analysis for 2-d eDCNN can be eased if more channels are permitted. In particular, the universal approximation property for 2-d eDCNN with numerous channels was studied in the recent work [33]. Since we are interested in eDCNN with only one channel, our result cannot be extended to 2-d eDCNN directly by using the approach developed in this paper.

The proof of Theorem 1 can be found in Appendix A. Generally speaking, boundedness of free parameters play a crucial role in the classical literature of learning with neural networks [3]. In particular, without any restrictions on free parameters, it can be found in [21], [22] that there exists a bounded sigmoid function such that the pseudo-dimension of a deep net with this activation function, two hidden layers and $O(d)$ free parameters is infinite, which implies that it is impossible to derive universal consistency for running ERM on such deep nets. On the contrary, with a controllable magnitude of free parameters, the universal consistency holds for deep nets with an arbitrary bounded sigmoid activation function [2]. The main breakthrough in Theorem 1 is that without any restrictions on free parameters, implementing ERM on DCNNs also yields universally consistent estimators. The main reason for this breakthrough is the piecewise linear property of ReLU, which is crucial to derive tight pseudo-dimension estimates for eDCNNs [4].

The universal consistency in Theorem 1 demonstrates the versatility of eDCNNs for different learning tasks, which is totally different from cDCNN that requires some fully connected layers suitably chosen for various learning tasks. This phenomenon is also verified by our real data experiments, where eDCNNs with the same structure are adaptive for different data but cDCNNs need different fully connected layers to enhance their learning performance (See Appendix B).

III. Numerical Experiments

In this section, we shall illustrate the versatility of eDCNNs through several simulated data and real data examples.

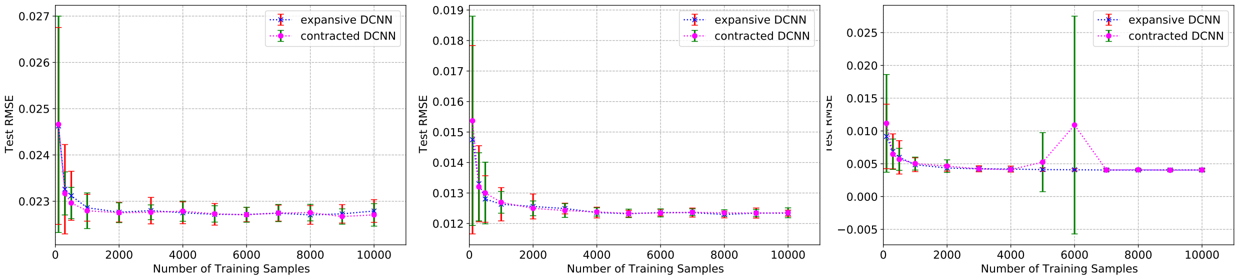
A. Simulated data examples

We consider the following regression model

$$y = \frac{\sin(\|x\|_2)}{\|x\|_2} + \varepsilon, \tag{10}$$

for generating training data, where x is a random vector with entries uniformly distributed in $[-10, 10]^d$, and ε is a random Gaussian noise with mean 0 and variance 0.01. To verify our theoretical assertion, we mainly consider three cases of the dimension of x, that is, the dimension d varies in $\{30, 100, 1000\}$. Then by using (10), we generate the training data sets with the number m varying in $\{100, 300, 500, 1000, 2000, \cdots, 9000, 10000\}$ for each d. For the network structure, we fix the filter length s as 2, and the number of network layers as $L = \text{ceil}(\sqrt[4]{m})$ that is consistent with the assumption of our theorem, where $\text{ceil}(\cdot)$ returns the value of a number rounded upwards to the nearest integer. To evaluate the prediction performance of the trained network, we further generate the test data sets in the same way as the training data, except that they are computed without noise, that is, $y_{test} = \dfrac{\sin(\|x_{test}\|_2)}{\|x_{test}\|_2}$. The size of the test data is chosen as 2000 for $d = 30, 100$, and 10000 for $d = 1000$, respectively.

Fig. 2 depicts the average results over 20 independent trials of both eDCNN and cDCNN in terms of RMSE (root-mean-square error). It is not hard to observe from this figure that, for all the three cases, the test RMSE of eDCNN gradually decreases and then reaches a stable manner as the number of training samples m grows. This conforms Theorem 1, since $\mathcal{E}(f_{D,L,s}) - \mathcal{E}(f_\rho) \to 0$ implies $\mathcal{E}(f_{D,L,s}) \to \mathcal{E}(f_\rho)$. Moreover, the proposed eDCNN illustrates similar prediction behaviors as the traditional cDCNN, except for the case of $m = 6000$ and $d = 1000$.



(a) (b) (c)

Fig. 2: The prediction results of both cDCNN and eDCNN on simulated data sets. (a) the error bar of $d = 50$; (b) the error bar of $d = 100$; (c) the error bar of $d = 1000$.

B. Real data examples

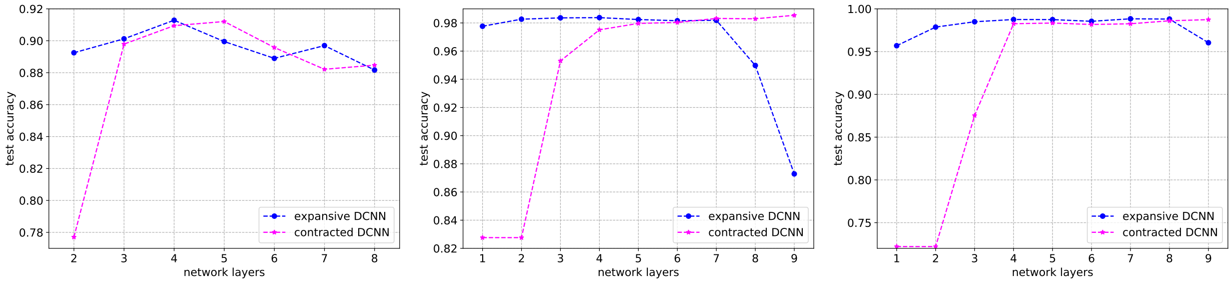
We now apply the proposed eDCNNs to deal with two real-world applications.

1. *Human Activity Recognition.* In this application, we would like to recognize the type of movement (walking, running, jogging, etc.) based on a given set of accelerometer data from a mobile device carried around a person's waist. The data set considered here is the WISDM data set firstly released in [19] and includes 1098207 samples with 6 categories. Different from the methods used in [19], we consider the traditional 1D cDCNN equipped with some fully connected layers as the baseline method for comparison. For the network structure of our eDCNN, we fix the filter length as $s = 9$, and the number of network

layers varies in $\{2, \ldots, 8\}$. The detailed architectures of the proposed eDCNN and the baseline network, and data descriptions can be found in Appendix B.

2. *ECG Heartbeat Classification.* An ECG is a 1D signal that is the result of recording the electrical activity of the heart using an electrode. It is a very useful tool that cardiologists use to diagnose heart anomalies and diseases. The data sets considered here are the MIT-BIH Arrhythmia Database and the PTB Diagnostic ECG Database that were preprocessed in [16]. The MIT-BIH Arrhythmia data set includes 109446 samples with 5 categories, and the PTB Diagnostic ECG Database includes 14552 samples with 2 categories. We also compare the eDCNN with a traditional 1D CNN whose network architecture can be found in Appendix B. In this application, we design our eDCNN in the same way as for the Human Activity Recognition application, except that the filter length is changed to $s = 19$. One can find more details about the description and structures for this real application in Appendix B.

Fig. 3 shows the comparison results of the proposed eDCNN over the traditional 1D cDCNN in terms of the misclassification rate on the test data. Here we change the the number of layers for the baseline networks to make a more thorough comparison. It is easy to see that, as for all the three data sets, the best predication results obtained by both eDCNN and cDCNN are comparable; as for the MIT-BIH Arrhythmia Database and the PTB Diagnostic ECG Database, eDCNN gives more stable predication results than cDCNN for a wide range of network layers. In addition, the test RMSE of the original baseline networks for those datasets are 0.905, 0.984 and 0.990, respectively. Considering the good theoretical guarantees and simple structures (see Appendix B) of eDCNN, we would prefer it over the traditional 1D cDCNN in practice.



(a) (b) (c)

Fig. 3: The comparison results on real data sets. (a) the WISDM dataset; (b) the MIT-BIH Arrhythmia Database; (c) the PTB Diagnostic ECG Database.

Acknowledge

The work of S. B. Lin is supported partially by the National Key R&D Program of China (No.2020YFA0713900) and the National Natural Science Foundation of China (No.618761332). The work of Y. Wang is supported partially by the National Natural Science Foundation of China (No.11971374). The work of D. X. Zhou is supported partially by the Research Grants Council of Hong Kong [Project # CityU 11308020], Germany/Hong Kong Joint Research Scheme [Project No. G-CityU101/20], Hong Kong Institute for Data Science, Laboratory for AI-Powered Financial Technologies, and National Science Foundation of China [Project No. 12061160462]. This paper was written when the last author visited SAMSI/Duke during his sabbatical leave. He would like to express his gratitude to their hospitality and financial support.

References

[1] Z. Allen-Zhu, Y. Li, and Z. Song. A convergence theory for deep learning via over-parameterization. ICML, 2019.
[2] M. Anthony and P. L. Bartlett. Neural Network Learning: Theoretical Foundations. Cambridge University Press, 2009.
[3] P. Bartlet. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the networks. IEEE Trans. Inf. Theory, 44: 525-536, 1998.
[4] P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian. Nearly-tight VC-dimension and Pseudodimension Bounds for Piecewise Linear Neural Networks. J. Mach. Learn. Res., 20(63): 1-17, 2019.
[5] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intel., 3: 1798-1828, 2013.
[6] C. K. Chui, S. B. Lin, B. Zhang, and D. X. Zhou. Realization of spatial sparseness by deep ReLU nets with massive data. *IEEE Transactions on Neural Networks and Learning Systems*, In Press, 2020.
[7] F. Cucker and D. X. Zhou. Learning Theory: an Approximation Theory Viewpoint. Cambridge University Press, Cambridge, 2007.
[8] I. Goodfellow, Y. Bengio, and A. Courville. Deep Learning. MIT Press, 2016.
[9] X. Guo, L. X. Li, and Q. Wu. Modeling interactive components by coordinate kernel polynomial models. Math. Found. Comput. 3: 263–277, 2020.
[10] Z. C. Guo, S. Lei, and S. B. Lin. Realizing data features by deep nets. IEEE Trans. Neural Netw. Learn. Syst., 31(10): 4036-4048, 2019.
[11] L. Györfy, M. Kohler, A. Krzyzak, and H. Walk. A Distribution-Free Theory of Nonparametric Regression. Springer, Berlin, 2002.
[12] B. Hanin. Universal function approximation by deep neural nets with bounded width and relu activations. Mathematics, 7 (10): 992, 2019.
[13] B. Hanin and M. Sellke. Approximating continuous functions by ReLU nets of minimal width. arXiv preprint arXiv:1710.11278, 2017.

[14] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. Inform. Comput., 100: 78-150, 1992.

[15] K. He, X. Zhang, S. Ren and J. Sun. Deep residual learning for image recognition, CVPR, 2016.

[16] M. Kachuee, S. Fazeli, and M. Sarrafzadeh. ECG heartbeat classification: a deep transferable representation. In Proceedings of IEEE International Conference on Healthcare Informatics (ICHI), 2018.

[17] M. Kohler and A. Krzyzak. Nonparametric regression based on hierarchical interaction models. IEEE Trans. Inf. Theory, 63: 1620-1630, 2017.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. NIPS, 1097–1105, 2012.

[19] J. R. Kwapisz, G. M. Weiss, and S. A. Moore. Activity recognition using cell phone accelerometers. In Proceedings of the Fourth International Workshop on Knowledge Discovery from Sensor Data, 2010.

[20] S. B. Lin. Generalization and expressivity for deep nets. IEEE Trans. Neural Netw. Learn. Syst., 30: 1392-1406, 2019.

[21] V. Maiorov and J. Ratsaby. On the degree of approximation by manifolds of finite pseudo-dimension. Constr. Approx., 15: 291-300, 1999.

[22] V. Maiorov and A. Pinkus. Lower bounds for approximation by MLP neural networks. Neurocomputing, 25: 81-91, 1999.

[23] T. Mao, Z. Shi, and D. X. Zhou. Theory of deep convolutional neural networks III: Approximating radial functions. Neural Networks, 144, 778-790, 2021.

[24] S. Mei, A. Montanari, and P. M. Nguyen. A mean field view of the landscape of two-layer neural networks. Proc. Nat. Acad. Sci. USA, 115 (33): E7665-E7671.

[25] S. Mendelson and R. Vershinin. Entropy and the combinatorial dimension. Invent. Math., 125: 37-55, 2003.

[26] K. Oono and T. Suzuki. Approximation and non-parametric estimation of ResNet-type convolutional neural networks. ICML, 2019: 4922-4931.

[27] J. Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. Ann. Statist., 48(4): 1875-1897, 2020.

[28] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. V. D. Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, and M. Lanctot. Mastering the game of go with deep neural networks and tree search. Nature, 529(7587): 484–489, 2016.

[29] D. Yarotsky. Error bounds for aproximations with deep ReLU networks. Neural Networks, 94: 103-114, 2017.

[30] D. X. Zhou. Deep distributed convolutional neural networks: Universality. Anal. Appl., 16: 895-919, 2018.

[31] D. X. Zhou. Universality of deep convolutional neural networks. Appl. Comput. Harmonic. Anal., 48: 784-794, 2020

[32] D. X. Zhou. Theory of deep convolutional neural networks: Downsampling. Neural Netw., 124: 319-327, 2020.

[33] T. Y. Zhou and D. X. Zhou. Theory of deep convolutional neural networks: 2D convolutions. Manuscript, 2021.

Appendix

A. Proof of Theorem 1

We divide our proof into three parts: capacity estimate, error analysis for bounded samples and universal consistency.

1) Capacity estimate: Let ν be a probability measure on $\mathcal{X}$. For a function $f:\mathcal{X}\to\mathbb{R}$, set $\|f\|_{L^p(\nu)}:=\left\{\int_{\mathcal{X}}|f(x)|^p d\nu\right\}^p$. Denote by $L^p(\nu)$ the set of all functions satisfying $\|f\|_{L^p(\nu)}<\infty$. For $\mathcal{V}\subset L^p(\nu)$, denote by $\mathcal{N}(\epsilon,\mathcal{V},\|\cdot\|_{L^p(\nu)})$ the covering number [11, Def. 9.3] of $\mathcal{V}$ in $L^p(\nu)$, which is the number of elements in a least ε-net of $\mathcal{V}$ with respect to $\|\cdot\|_{L^p(\nu)}$. In particular, denote by $\mathcal{N}_p(\epsilon,\mathcal{V},x_1^m):=\mathcal{N}(\epsilon,\mathcal{V},\|\cdot\|_{L^p(\nu_m)})$ with ν_m the empirical measure with respect to $x_1^m=(x_1,\dots,x_m)\in\mathcal{X}^m$. Define further $\mathcal{M}(\epsilon,\mathcal{V},\|\cdot\|_{L^p(\nu)})$ to be the ε-packing number of $\mathcal{V}$ with respect to $\|\cdot\|_{L^p(\nu)}$, i.e. $\mathcal{M}(\epsilon,\mathcal{V},\|\cdot\|_{L^p(\nu)})$ is the largest integer N such that a subset $\{g_1,\dots,g_N\}$ of $\mathcal{V}$ satisfies $\|g_j-g_k\|_{L^p(\nu)}\geq\varepsilon$ for $1\leq j<k\leq N$. For the sake of brevity, we also denote $\mathcal{M}_p(\epsilon,\mathcal{V},x_1^m):=\mathcal{M}(\epsilon,\mathcal{V},\|\cdot\|_{L^p(\nu_m)})$ with respect to $x_1^m=(x_1,\dots,x_m)\in\mathcal{X}^m$. The following lemma found in [11, Lemma 9.2] presents a relation between ε-covering numbers and ε-packing numbers.

Lemma 1: Let $\mathcal{V}$ be a class of functions on $\mathcal{X}$ and let ν be a probability measure on $\mathcal{X}$, $p\geq 1$ and $\varepsilon>0$. Then

$$\mathcal{M}(2\varepsilon,\mathcal{V},\|\cdot\|_{L^p(\nu)})\leq\mathcal{N}(\varepsilon,\mathcal{V},\|\cdot\|_{L^p(\nu)})\leq\mathcal{M}(\varepsilon,\mathcal{V},\|\cdot\|_{L^p(\nu)}).$$

In particular,

$$\mathcal{M}_p(2\epsilon,\mathcal{V},x_1^m)\leq\mathcal{N}_p(\epsilon,\mathcal{V},x_1^m)\leq\mathcal{M}_p(\epsilon,\mathcal{V},x_1^m).$$

Denote further by $Pdim(\mathcal{V})$ the pseudo-dimension [2, Chap. 14] of $\mathcal{V}$, which is the largest integer ℓ for which there exists $(\xi_1,\dots,\xi_m,\eta_1,\dots,\eta_m)\in\mathcal{X}^m\times\mathbb{R}^m$ such that for any $(a_1,\dots,a_\ell)\in\{0,1\}^\ell$ there is some $v\in\mathcal{V}$ satisfying

$$\forall\ i:\quad v(\xi_i)>\eta_i\Leftrightarrow a_i=1.$$

The following lemma that can be found in [14, Theorem 6] (see also [25, Theorem 1]) presents a close relation between ε-packing numbers and pseudo-dimensions.

Lemma 2: Let $R>0$ and $\mathcal{V}_R$ be a set of functions from $\mathcal{X}$ to $[-R,R]$. Then for any $\varepsilon\in(0,R]$, there holds

$$\mathcal{M}(\varepsilon,\mathcal{V}_R,\|\cdot\|_{L^1(\nu)})\leq 2\left(\frac{2eR}{\varepsilon}\ln\frac{2eR}{\varepsilon}\right)^{Pdim(\mathcal{V}_R)}.$$

From (5), there are $s+1$ tunable weights from $\vec{w}_k$ and $d+ks$ tunable bias vector components from $\vec{b}_k$ in the k-th layers for $k=1,\dots,L-1$. Noting additional $d+Ls$ tunable outer weights in the L-th layer, there are totally

$$n_{L,s}:=(s+1)L+d+Ls+\sum_{k=1}^{L}(d+ks) \tag{11}$$

free parameters paved on

$$d_{L,s}:=1+d+\sum_{k=1}^{L}(d+ks) \tag{12}$$

neurons in the eDCNN.

Our main tool is a tight pseudo-dimension estimate for deep nets with a piecewise linear activation. In fact, combining [4, Theorem 7] and [2, Theorem 14.1], we can get the following pseudo-dimension estimate for the eDCNN without any restrictions on the magnitudes of free parameters.

Lemma 3: There exists an absolute constant C_0 such that

$$Pdim(\mathcal{H}_{L,s}) \leq C_0 L n_{L,s} \log d_{L,s}, \tag{13}$$

where $n_{L,s}$ and $d_{L,s}$ are given in (11) and (12) respectively.

Our aim is to use the above three lemmas to derive a tight bound for the covering numbers of eDCNNs. For $M > 0$, define

$$\pi_M \mathcal{H}_{L,s} := \{\pi_M f : f \in \mathcal{H}_{L,s}\}. \tag{14}$$

Since $Pdim(\pi_M \mathcal{H}_{L,s}) \leq Pdim(\mathcal{H}_{L,s})$ [21, p. 297], it follows from Lemma 3 that

$$Pdim(\pi_M \mathcal{H}_{L,s}) \leq C_0 L n_{L,s} \log d_{L,s}.$$

Plugging the above estimate into Lemma 2, we then have

$$\mathcal{M}(\varepsilon, \pi_M \mathcal{H}_{L,s}, \|\cdot\|_{L^1(\nu)}) \leq 2\left(\frac{2eM}{\varepsilon}\right)^{2C_0 L n_{L,s} \log d_{L,s}}.$$

Then it follows from Lemma 1 with $\nu = \nu_m$ and an arbitrary $x_1^m \in \mathcal{X}^m$ that the following covering number estimates for the eDCNN without any restrictions to the magnitudes of parameters hold.

Lemma 4: For any $0 < \varepsilon \leq M$, there holds

$$\log_2 \sup_{x_1^m \in \mathcal{X}^m} \mathcal{N}_1(\epsilon, \pi_M \mathcal{H}_{L,s}, x_1^m) \leq c^* L^2 (Ls + d) \log(L(s+d)) \log \frac{M}{\epsilon},$$

where c^* is an absolute constant.

2) Error analysis for bounded samples: Write $y_M = \pi_M y$ and $y_{i,M} = \pi_M y_i$. Define

$$\mathcal{E}_{\pi_M}(f) = \int_{\mathcal{Z}} (f(x) - y_M)^2 d\rho,$$

and

$$\mathcal{E}_{\pi_M,D}(f) = \frac{1}{m}\sum_{i=1}^{m} (f(x_i) - y_{i,M})^2.$$

In this part, we aim at bounding $\mathcal{E}_{\pi_M}(\pi_M f_{D,L,s}) - \mathcal{E}_{\pi_M,D}(\pi_M f_{D,L,s})$. Our tool is the following concentration inequality which can be easily deduced from [11, Theorem 11.4].

Lemma 5: Assume $|y| \leq B$ and $B \geq 1$. Let $\mathcal{F}$ be a set of functions $f : \mathcal{X} \to \mathbb{R}$ satisfying $|f(x)| \leq B$. Then for each $m \geq 1$, with confidence at least

$$1 - 14 \max_{x_1^m \in \mathcal{X}^m} \mathcal{N}_1\left(\frac{\beta\epsilon}{20B}, \mathcal{F}, x_1^m\right) \exp\left(-\frac{\epsilon^2(1-\epsilon)\alpha m}{214(1+\epsilon)B^4}\right),$$

there holds

$$\begin{aligned} &\mathcal{E}(f) - \mathcal{E}(f_\rho) - (\mathcal{E}_D(f) - \mathcal{E}_D(f_\rho)) \\ \leq\ & \epsilon(\alpha + \beta + \mathcal{E}(f) - \mathcal{E}(f_\rho)), \qquad \forall f \in \mathcal{F}, \end{aligned}$$

where $\alpha,\beta > 0$ and $0 < \epsilon \leq 1/2$.

Based on Lemma 5 and Lemma 4, we can derive the following result.

Lemma 6: If $M_m^2 m^{-\theta} \to 0$ and (9) holds for some $\theta \in (0, 1/2)$, then

$$\lim_{m\to\infty} \mathcal{E}_{\pi_M}(\pi_M f_{D,L,s}) - \mathcal{E}_{\pi_M,D}(\pi_M f_{D,L,s}) = 0$$

holds almost surely.

Proof: Since $|\pi_M f_{D,L,s}(x)|, |y_M|, |y_{i,M}| \leq M$, we have

$$|\mathcal{E}_{\pi_M}(\pi_M f_{D,L,s}) - \mathcal{E}_{\pi_M,D}(\pi_M f_{D,L,s})| \leq 8M^2.$$

Then it follows from Lemma 5 with $\alpha = \beta = 1$ and $\epsilon = m^{-\theta}$ that with confidence at least

$$1 - 14 \max_{x_1^m \in \mathcal{X}^m} \mathcal{N}_1\left(\frac{1}{20Mm^\theta}, \pi_M \mathcal{H}_{L,s}, x_1^m\right) \exp\left(-\frac{m^{1-2\theta}}{428M^4}\right),$$

there holds

$$\mathcal{E}_{\pi_M}(\pi_M f_{D,L,s}) - \mathcal{E}_{\pi_M}(f_\rho) - (\mathcal{E}_{\pi_M,D}(\pi_M f_{D,L,s}) - \mathcal{E}_{\pi_M,D}(f_\rho)) \leq 8(M^2 + 2)m^{-\theta}.$$

Due to Lemma 4, we have

$$\begin{aligned}&\max_{x_1^m\in\mathcal{X}^m}\mathcal{N}_1\left(\frac{1}{20Mm^{\theta}},\pi_M\mathcal{H}_{L,s},x_1^m\right)\exp\left(-\frac{m^{1-2\theta}}{428M^4}\right)\\ \leq\quad&\exp\left(c^*\log(20M^2m^{\theta})L^2(d+sL)\log(L(s+d))-\frac{m^{1-2\theta}}{428M^4}\right).\end{aligned}$$

Noting (9), we obtain

$$\lim_{m\to\infty}\max_{x_1^m\in\mathcal{X}^m}\mathcal{N}_1\left(\frac{1}{20M_mm^{\theta}},\pi_{M_m}\mathcal{H}_{L_m,s},x_1^m\right)\exp\left(-\frac{m^{1-2\theta}}{428M_m^4}\right)=0.$$

Thus, together with the strong law of large numbers, as $m\to\infty$, we see that

$$\mathcal{E}_{\pi_{M_m}}(\pi_{M_m}f_{D,L_m,s})-\mathcal{E}_{\pi_{M_m},D}(\pi_{M_m}f_{D,L_m,s})\leq 8M^2m^{-\theta}\to 0$$

holds almost surely. This completes the proof of Lemma 6. ■

3) Universal consistency: Our final tool is the universality of eDCNNs, which was proved in [31, Theorem 1].

Lemma 7: Let $2\leq s\leq d$. For any compact subset $\mathcal{X}$ of $\mathbb{R}^d$ and any $f\in C(\mathcal{X})$, there exists some $h_{L,s}\in\mathcal{H}_{L,s}$ such that

$$\lim_{L\to+\infty}\|f-h_{L,s}\|_{C(\mathcal{X})}=0. \tag{15}$$

Now we are in a position to prove Theorem 1.

Proof of Theorem 1: Since $\mathbf{E}\{y^2\}<\infty$, we have $f_\rho\in L^2(\rho_X)$. It follows from Lemma 7 that for any $\varepsilon>0$, there exists some $g_\varepsilon\in\mathcal{H}_{L_\varepsilon,s}$ with sufficiently large L_ε such that

$$\|f_\rho-g_\varepsilon\|^2_{L^2(\rho_X)}\leq\varepsilon. \tag{16}$$

The triangle inequality then yields

$$\begin{aligned}&\mathcal{E}(\pi_Mf_{D,L,s})-\mathcal{E}(f_\rho)\\ \leq\quad&\mathcal{E}(\pi_Mf_{D,L,s})-(1+\varepsilon)\mathcal{E}_{\pi_M}(\pi_Mf_{D,L,s})\\ +\quad&(1+\varepsilon)(\mathcal{E}_{\pi_M}(\pi_Mf_{D,L,s})-\mathcal{E}_{\pi_M,D}(\pi_Mf_{D,L,s}))\\ +\quad&(1+\varepsilon)(\mathcal{E}_{\pi_M,D}(\pi_Mf_{D,L,s})-\mathcal{E}_{\pi_M,D}(f_{D,L,s}))\\ +\quad&(1+\varepsilon)\mathcal{E}_{\pi_M,D}(f_{D,L,s})-(1+\varepsilon)^2\mathcal{E}_D(f_{D,L,s})\\ +\quad&(1+\varepsilon)^2(\mathcal{E}_D(f_{D,L,s})-\mathcal{E}_D(g_\varepsilon))\\ +\quad&(1+\varepsilon)^2(\mathcal{E}_D(g_\varepsilon)-\mathcal{E}(g_\varepsilon))\\ +\quad&(1+\varepsilon)^2(\mathcal{E}(g_\varepsilon)-\mathcal{E}(f_\rho))\\ +\quad&((1+\varepsilon)^2-1)\mathcal{E}(f_\rho)\\ =:\quad&\sum_{\ell=1}^{8}B_\ell.\end{aligned}$$

To deduce the strongly universal consistency, we should bound B_ℓ, $\ell=1,\dots,8$, in probability, respectively. As

$$(a+b)^2\leq(1+\varepsilon)a^2+(1+1/\varepsilon)b^2\quad\text{for } a,b>0, \tag{17}$$

we have

$$\begin{aligned}B_1\quad=\quad&\int_{\mathcal{Z}}|\pi_Mf_{D,L,s}(x)-y_M+y_M-y|^2d\rho\\ -\quad&(1+\varepsilon)\int_{\mathcal{Z}}|\pi_Mf_{D,L,s}(x)-y_M|^2d\rho\\ \leq\quad&(1+1/\varepsilon)\int_{Z}|y-y_M|^2d\rho.\end{aligned}$$

Since $M=M_m\to\infty$ as $m\to\infty$, we obtain

$$B_1\to 0\quad\text{when } m\to\infty.$$

From Lemma 6, (9) and $M_m^2m^{-\theta}\to 0$, it follows that

$$B_2\to 0\quad\text{when } m\to\infty$$

holds almost surely. The definition of the truncation operator yields

$$\frac{1}{m}\sum_{i=1}^{m}|\pi_Mf_{D,L,s}(x_i)-y_{i,M}|^2-\frac{1}{m}\sum_{i=1}^{m}|f_{D,L,s}(x_i)-y_{i,M}|^2\leq 0.$$

Therefore, we have

$$B_3 \leq 0.$$

According to the strong law of large numbers and (17), we get

$$B_4 \leq (1+\varepsilon)(1+1/\varepsilon)\frac{1}{m}\sum_{i=1}^{m}|y_i - y_{i,M}|^2 \to (1+\varepsilon)(1+1/\varepsilon)\int_Z |y-y_M|^2 d\rho$$

as $m \to \infty$ almost surely. Therefore, $M_m \to \infty$ and the definition of y_M yield

$$B_4 \to 0.$$

Due to (7), we obtain

$$B_5 = (1+\varepsilon)^2\left(\frac{1}{m}\sum_{i=1}^{m}|f_{D,L}(x_i) - y_i|^2 - \frac{1}{m}\sum_{i=1}^{m}|g_\varepsilon(x_i) - y_i|^2\right) \leq 0.$$

By the strong law of large numbers again, we have almost surely

$$B_6 \to 0, \quad (\text{as } m \to \infty).$$

To bound B_7, we note from (6)

$$B_7 = (1+\varepsilon)^2\|g_\varepsilon - f_\rho\|^2_{L^2_{\rho_X}}.$$

The above equality together with (16) and $L_m \to \infty$ yields

$$B_7 \leq (1+\varepsilon)^2\varepsilon.$$

Noting

$$(1+\varepsilon)^2 - 1 = \varepsilon(\varepsilon+2),$$

we have

$$B_8 \leq \varepsilon(\varepsilon+2)\int_Z |f_\rho(x) - y|^2 d\rho.$$

Using all the above assertions, we can concludes that

$$\limsup_{m\to\infty} \mathcal{E}(\pi_M f_{D,L,s}) - \mathcal{E}(f_\rho) \leq (1+\varepsilon)^2\varepsilon + \varepsilon(\varepsilon+2)\int_Z |f_\rho(x) - y|^2 d\rho$$

holds almost surely. This proves Theorem 1 by setting $\varepsilon \to 0$. ■

B. Description and structures for real data sets

In this part, we introduce some auxiliary information for our real data experiments.

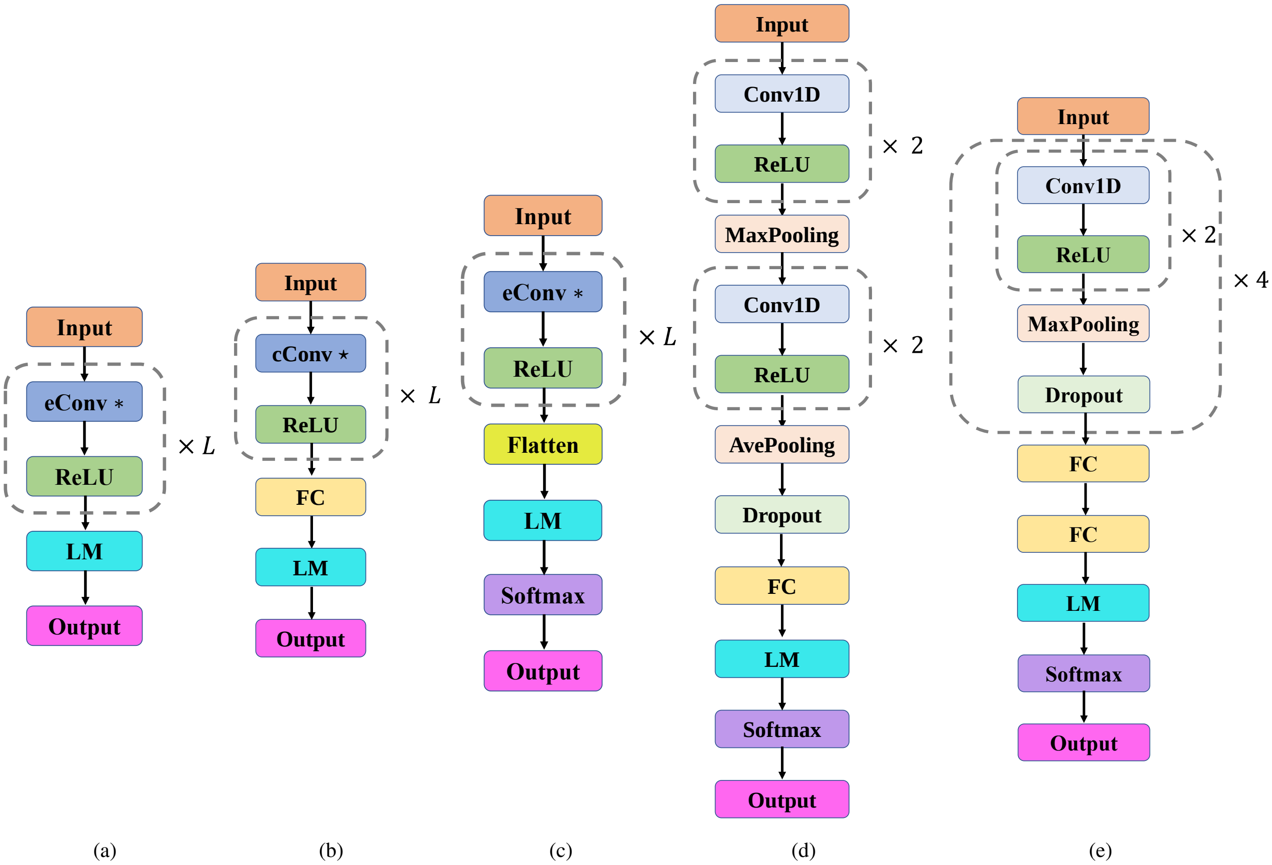
1) Detailed description of implementing real data sets: We shall provide more details on how to implement the three real data sets using the proposed eDCNNs.

1. *WISDM dataset.* This dataset is a collection of accelerometer data taken from a smartphone that various people carried with them while conducting six different exercises, i.e., Downstairs, Jogging, Sitting, Standing, Upstairs, and Walking. For each exercise, the accelerations with respect to the x, y, and z axes were measured and captured with a timestamp and person ID.

We first split all the samples based on the user IDs, that is, users with ID 1 to 28 are used for training the model and users with ID greater than 28 are consider as the test set. We then reshape the data by considering 80 time periods as one record with the accelerations with respect to the x, y, and z axes. Therefore, the input of our network is a matrix of size 80×3 and the output of our network is a vector with length 6. To train the network, we choose the cross entropy function as the loss function, and the adam as the optimizer. Then the batch size and the maximal number of epochs are set as 400 and 50, respectively. Similar to other traditional CNNs, the early stopping strategy is also used for training out network.

2. *MIT-BIH Arrhythmia database.* The MIT-BIH Arrhythmia Database contains 48 half-hour excerpts of two-channel ambulatory ECG recordings, obtained from 47 subjects studied by the BIH Arrhythmia Laboratory between 1975 and 1979. It was preprocessed and reported in [16] based on the beat extraction method described in III.A of the paper, and thus the data set has totally 109446 samples corresponding to five categories.

To test the performance of our network, we randomly choose 80% samples as the training set and the remaining 20% as the test set. Since each sample has 187 attributes, the input of our network is a vector with length 187 and the output of our network is a vector with length 5. Similar to before, the cross entropy function is chosen as the loss function, the adam is consider as the optimizer and the early stopping strategy is used to train our network. Then the batch size and the maximal number of epochs are set as 32 and 100, respectively.



(a) (b) (c) (d) (e)

Fig. 4: The detailed network architectures of our eDCNN and baseline methods used in the simulated and real data experiments, where the "LM" module means linear mapping and the "FC" module is a fully connected layer. (a) eDCNN for simulated data examples; (b) cDCNN for simulated data examples; (c) eDCNN for real data examples; (d) baseline network for the Human Activity Recognition task; (e) baseline network for the ECG Heartbeat Classification task.

3. *PTB Diagnostic ECG Database.* The database contains 549 records from 290 subjects (aged 17 to 87, mean 57.2; 209 men, mean age 55.5, and 81 women, mean age 61.6; ages were not recorded for 1 female and 14 male subjects). Each subject is represented by one to five records. By using the beat extraction method in [16], this data set is transformed to a new one that has totally 14552 samples corresponding to two categories. The implementation details of this data is the same as the MIT-BIH Arrhythmia database, except that the network's output is a vector with length 2.

2) Architectures of the implemented DCNNs: Fig. 4 plots the detailed network architectures of our eDCNNs and baseline networks used in the simulated and real data experiments. It is easy to see that the network structures of the proposed eDCNNs are simpler and cleaner than the baseline networks.