algorithms

MDPI

Article

Improved Neural Networks Based on Mutual Information via Information Geometry

Meng Wang [1], Chuang-Bai Xiao [1], Zhen-Hu Ning [1], Jing Yu [1], Ya-Hao Zhang [2,*] and Jin Pang [2]

[1] College of Computer Science, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China; mmking@emails.bjut.edu.cn (M.W.); cbxiao@bjut.edu.cn (C.-B.X.); ningzhenhu@126.com (Z.-H.N.); jinp0917@gmail.com (J.Y.)

[2] State Grid Information & Telecommunication Co.,Ltd., 1401 Main Building No 2 BaiGuang Avenue, Xi Cheng District, Beijing 100031, China; pangjin2019@gmail.com

* Correspondence: oliverzhang2016@gmail.com; Tel.: +86-1355-245-7610

Received: 18 February 2019; Accepted: 2 May 2019; Published: 13 May 2019

Abstract: This paper presents a new algorithm based on the theory of mutual information and information geometry. This algorithm places emphasis on adaptive mutual information estimation and maximum likelihood estimation. With the theory of information geometry, we adjust the mutual information along the geodesic line. Finally, we evaluate our proposal using empirical datasets that are dedicated for classification and regression. The results show that our algorithm contributes to a significant improvement over existing methods.

Keywords: neural networks; information geometry; geodesic line

1. Introduction

An artificial neural network is a framework for many different machine learning algorithms to work together and process complex data inputs; it is vaguely inspired by biological neural networks that constitute animal brains [1]. Neural networks have been widely used in many application areas, such as systems identification [2], signal processing [3,4], and so on. Furthermore, several variants of neural networks have been derived from the context of applications. The convolutional neural network is one of the most popular variants. It is composed of one or more convolutional layers with fully connected layers and pooling layers [5]. In addition, the deep belief network (DBN) is considered to be a composition of simple learning modules that make up each layer [6]. Several restricted Boltzmann machines [7] are stacked and trained greedily to form the DBN. In 2013, Grossberg proposed a recurrent neural network, which is a class of artificial neural network in which connections between nodes form a directed graph along a sequence [8]. The output layer can obtain information from past and future states simultaneously.

However, improving the performance of a neural network remains an open question. From the viewpoint of information theory, mutual information is used to optimize neural networks. In the method of the ensemble, two neural networks are forced to convey different information about some features of their input by minimizing the mutual information between the variables extracted by the two neural networks [9]. In this method, mutual information is used to measure the correlation between two hidden neurons. In 2010, an adaptive merging and splitting algorithm (AMSA) pruned hidden neurons by merging correlated hidden neurons and added hidden neurons by splitting existing hidden neurons [10]. In this method, the mutual information is used to measure the correlation between the two hidden neurons. In 2015, Berglund et al. proposed measuring the usefulness of hidden units in Boltzmann machines with mutual information [11]. However, the measure is not suitable as the sole criterion for model selection. In addition, the measure that was shown to correlate well with

the entropy does not agree with the well-known and observed phenomenon that sparse features that have in general low entropy are good for many machine learning tasks, including classification. By a new objective function, a neural network with maximum mutual information training is proposed to solve the problem of two-class classification [12]. It focuses on maximizing the difference between the estimated possibility of the two classes. However, when the two classes are completely separable, the algorithm cannot outperform the traditional algorithm of maximum likelihood estimation. To address the problems above, we introduce the theory of information geometry into neural networks.

Information geometry is a branch of mathematics that applies the theory of Riemannian manifolds to the field of probability theory [13]. By the Fisher information [14], the probability distributions for a statistical model are treated as the points of a Riemannian manifold. This approach focuses on studying the geometrical structures of parameter spaces of distributions. Information geometry has found various applications in many fields, such as control theory [15], signal processing [16], the expectation–maximization algorithm [17], and many others [18,19]. In addition, information geometry is used to study neural networks. In 1991, Amari [20] studied the dualistic geometry of the manifold of higher-order neurons. Furthermore, the natural gradient works efficiently in the learning of neural networks [21]. The natural gradient learning method can overcome some disadvantages in the learning process of the networks effectively, such as the slow learning speed and the existence of plateaus. In 2005, a novel information geometric-based variable selection criterion for multilayer perceptron networks was described [22]. It is based on projections of the Riemannian manifold as defined by a multilayer perceptron network on submanifolds with reduced input dimension.

In this paper, we propose a new algorithm based on the theory of mutual information and information geometry. We study the information covered by a whole neural network quantitatively via mutual information. In the proposed method, the learning of the neural network attempts to keep as much information as possible while maximizing the likelihood. Then, with the theory of information geometry, the optimization of the mutual information turns into minimizing the difference information between the estimated distribution and the original distribution. The distributions are mapping to points on a manifold. The Fisher distance is a proper measurement of the path between the points. Then, we obtain a unique solution of the mutual information by minimizing the Fisher distance between the estimated distribution and the original distribution along the shortest path (geodesic line).

2. Materials and Methods

Let us consider a neural network with one input layer, one output layer, and one hidden layer. The weights for the input are denoted by w. The weights for the hidden neurons are v. Here, x is the input, while $\widetilde{y}$ is the actual output, and z is the activation of the hidden neurons. The concatenation of the output and hidden neurons is represented as a vector $r = [\widetilde{y}, z]$. Figure 1 illustrates the architecture of the neural network.

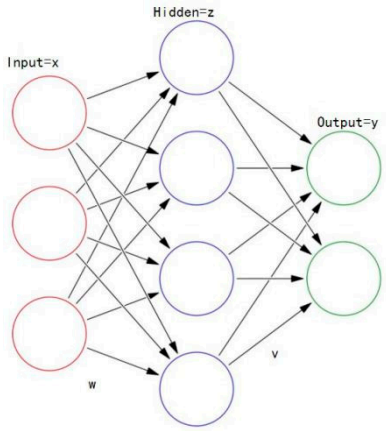


Figure 1. The architecture of the neural network.

The neural network is a static and usually nonlinear model

$$p(r|x) = p(\widetilde{y}, z|x, u) = p(\widetilde{y}|z, v)p(z|x, w) \tag{1}$$

where the parameters of the neural network are denoted by $u = [w, v]$.

The mutual information (MI) [23] of two random variables quantifies the "amount of information" obtained about one random variable, through the other random variable. In the proposed algorithm, we treat the input x and the combination of the output and hidden neurons r as two random variables. Correspondingly, the MI quantifies the "amount of information" obtained by the output and hidden variable from the input. The learning of the neural network attempts to keep as much information as possible while maximizing the likelihood. We define a novel objective function as

$$J = E_x[(\widetilde{y} - y)^2] + \lambda I(x; r) \tag{2}$$

where y is the desired output (i.e., the true value of the label). The first item is the objective function of traditional backpropagation neural networks, while the second item is the MI of x and r. With the first item, we maximize the likelihood similar to what traditional neural networks do. With the second item, the proposed method helps the neural networks to keep information from the input. Here, λ is a constant determined by experience.

Many methods exist for minimizing the first item, such as conjugate gradient [24]. Therefore, we just discuss how to work with the second item in this paper. By the definition of MI and Kullback–Leibler divergence (KLD) [25], the mutual information of x and r is

$$\begin{aligned} I(x; r) &= \int_{x \in X} \int_{r \in R} p(x, r) \log \tfrac{p(x,r)}{p(x)p(r)} dr dx \\ &= \int_{x \in X} p(x) \int_{r \in R} p(r|x) \log \tfrac{p(x,r)}{p(x)p(r)} dr dx \\ &= \int_{x \in X} p(x) \int_{r \in R} p(r|x) \log \tfrac{p(r|x)}{p(r)} dr dx \\ &= \int_{x \in X} p(x) KLD(p(r|x) \| p(r)) dx \end{aligned} \tag{3}$$

where X is the value domain of x, while R is the value domain of r.

In general, the density $p(x)$ is assumed to be unknown. Therefore, we make a finite-sample approximation, since we work with a finite dataset. The size of the set is N. Then, the mutual information in (3) is calculated as

$$I(x; r) = \frac{1}{N} \sum_{i=1}^{N} KLD(p(r|x_i) \| p(r)) \tag{4}$$

where x_i is the ith input data.

In addition, we have

$$\begin{aligned} KLD(p(r|x_i) \| p(r)) &= \iint_{\widetilde{y} \in Y, z \in Z} p(\widetilde{y}|z, v) p(z|x_i, w) \log \tfrac{p(\widetilde{y}|z,v) p(z|x_i,w)}{p(\widetilde{y}|z) p(z)} d\widetilde{y} dz \\ &= \int_{z \in Z} p(z|x_i, w) \log \tfrac{p(z|x_i,w)}{p(z)} \left(\int_{\widetilde{y} \in Y} p(\widetilde{y}|z, v) d\widetilde{y} \right) dz + \iint_{\widetilde{y} \in Y, z \in Z} p(\widetilde{y}|z, v) p(z|x_i, w) \log \tfrac{p(\widetilde{y}|z,v)}{p(\widetilde{y}|z)} d\widetilde{y} dz \\ &= \int_{z \in Z} p(z|x_i, w) \log \tfrac{p(z|x_i,w)}{p(z)} dz + E_z \left[\int_{\widetilde{y} \in Y} p(\widetilde{y}|z, v) \log \tfrac{p(\widetilde{y}|z,v)}{p(\widetilde{y}|z)} d\widetilde{y} \right] \\ &= KLD(p(z|x_i, w) \| p(z)) + E_z[KLD(p(\widetilde{y}|z, v) \| p(\widetilde{y}|z))] \\ &= KLD(p(z|x_i, w) \| p(z)) + \tfrac{1}{N} \sum_{i=1}^{N} KLD(p(\widetilde{y}|z_i, v) \| p(\widetilde{y}|z_i)) \\ &= KLD(p(z|x_i, w) \| p(z)) \end{aligned} \tag{5}$$

where z_i is the value of the vector of hidden neurons when the input is x_i, while Z is the value domain of z, and $p(\widetilde{y}|z_i, v) = p(\widetilde{y}|z_i)$ so that $KLD(p(\widetilde{y}|z_i, v)||p(\widetilde{y}|z_i)) = 0$.

The original distribution $p(z)$ can be derived from a finite-sample approximation, which gives us the following:

$$p(z) = \frac{1}{N}[p(z|x_1, w) + p(z|x_2, w) + \ldots + p(z|x_N, w)] = \frac{1}{N}\sum_{i=1}^{N} p(z|x_i, w) \tag{6}$$

Next, we will show an effective way to calculate the *KLD* in (5) with the theory of information geometry.

Any function that can be written in the following form is from the exponential family [26]:

$$p(x|\theta) = \exp(x \cdot \theta - \varphi(\theta)) \tag{7}$$

where φ is a function of θ.

Without loss of generality, the probability of the activation of each hidden unit can be approximated with a function from the exponential family [27]. Here, z^j represents the jth hidden unit; then,

$$p(z^j|x_i, w) = p(z^j|\theta^j) = \exp(z^j \cdot \theta^j - \varphi^j(\theta^j)) \tag{8}$$

where θ^j is a function of $x \cdot w^j$ (i.e., the linear weighted input of z^j), while w_j is the jth column of w, for $j \in \{1, 2, \ldots, l\}$, and l is the size of the hidden layer. Additionally, φ^j is a function of θ^j. Considering that the activations of the hidden neurons are independent, the probability of z is calculated as

$$p(z|x_i, w) = p(z|\theta) = \prod_{j=1}^{l} p(z^j|\theta^j) = \prod_{j=1}^{l} \exp(z^j \cdot \theta^j - \varphi^j(\theta^j)) = \exp(\sum_{j=1}^{l} z^j \cdot \theta^j - \sum_{j=1}^{l} \varphi^j(\theta^j)) = \exp(z \cdot \theta - \varphi(\theta)) \tag{9}$$

Thus, $p(z|\theta)$ is from the exponential family, while θ is a function of the linear input $x \cdot w$ of z. In addition, φ is a function of θ.

A classic parametric space for this family of probability density functions (PDFs) is

$$H = \left\{\eta = \frac{\partial \varphi(\theta)}{\partial \theta}\right\} \tag{10}$$

For the distributions in the exponential family, there are two dual coordinate systems θ and η on the manifold (which is defined as a topological space that locally resembles Euclidean space near each point) of the parameters [28]. Here, η is given by

$$\eta = \frac{\partial \varphi(\theta)}{\partial \theta} = E(z) \tag{11}$$

With (9) and (11), we calculate the η_i for each z_i. From (6), it is evident that the $p(z)$ is the mean of the conditional probability over each input x_i. Thus, with (11), the parameters in the density in (6) are approximated by

$$\overline{\eta} = \frac{1}{N}\sum_{i=1}^{N} \eta_i \tag{12}$$

For the distribution in the exponential family, we have

$$\psi(\eta) = \int p(z|\eta) \log p(z|\eta) dr \tag{13}$$

For distribution with n parameters, the Fisher distance between two points θ' and $\theta*$ in the half-plane H reflects the dissimilarity between the associated PDF's. It is defined as the minimal integral [29]

$$d_F(\theta', \theta*) := \min_{\theta(t)} \int_{\theta'}^{\theta*} \left(\sqrt{\left(\frac{d\theta}{dt}\right)^T G(\theta)\left(\frac{d\theta}{dt}\right)}\right) dt \tag{14}$$

where $\theta(t)$ represents a curve that is parameterized by t, while $G(\theta)$ is the Fisher information matrix which is defined as

$$G(\theta) = [g_{ab}(\theta)] \tag{15}$$

$$g_{ab}(\theta) = E\left[\frac{\partial \log p(z|\theta)}{\partial \theta_a} \frac{\partial \log p(z|\theta)}{\partial \theta_b}\right] \tag{16}$$

where $a \in \{1, 2, \ldots, n\}$ and $b \in \{1, 2, \ldots, n\}$ are the indexes of the elements in G.

The curve that satisfies the minimization condition in (14) is called a geodesic line.

Theorem 1 [30]**.** *For the distributions in the exponential family, the Fisher distance is the Kullback–Leibler divergence given by*

$$KLD(p(z|\theta')||p(z|\eta*)) = \varphi(\theta') + \psi(\eta*) - \theta' \cdot \eta* \tag{17}$$

With (11) to (13), we obtain a dual description for $p(z)$, as follows:

$$\overline{\eta} = \nabla\varphi(\theta) \tag{18}$$

$$\psi(\overline{\eta}) = \int p(z|\overline{\eta}) \log p(z|\overline{\eta}) dr \tag{19}$$

Let $\theta' = \theta$ and $\eta* = \overline{\eta}$ in (14). With (5), we have

$$KLD(p(r||x_i)||p(r)) = KLD(p(z|\theta)||p(z)) = \varphi(\theta) + \psi(\overline{\eta}) - \theta \cdot \overline{\eta} \tag{20}$$

Then, we substitute (20) into (3) to obtain the second item in (2). By the definition of the Fisher distance (14) and Theorem 1, we know that minimizing the second item in (2) is equivalent to minimizing the Fisher distance between the estimated distribution $p(r|x_i)$ and the original distribution $p(r)$ along the shortest path (geodesic line). To describe the geodesic line using local coordinates, we must solve the geodesic equations given by the Euler–Lagrange equations, as follows:

$$\frac{d^2\theta_k}{dt^2} + \sum_{a=1}^{n}\sum_{b=1}^{n}\left(\frac{1}{2}\sum_{a=1}^{n} g^{kl}\left(\frac{\partial g_{al}}{\partial \theta_b} + \frac{\partial g_{bl}}{\partial \theta_a} - \frac{\partial g_{ab}}{\partial \theta_l}\right)\right)\frac{d\theta_a}{dt}\frac{d\theta_b}{dt} = 0, \forall a, b, k, l \in \{1, \ldots, n\} \tag{21}$$

where $\theta = (\theta_1, \theta_2, \ldots, \theta_n)'$. By solving (21), one obtains a unique solution for the mutual information by minimizing the Fisher distance along the shortest path (geodesic line). In practice, we replace this step by the neural gradient descent [31], which makes the parameters in the second item in (2) update along the geodesic line. Here, the direction of the updating of the parameters is

$$\hat{\nabla}\theta = G^{-1}(\theta)\nabla I(x; r) \tag{22}$$

where ∇ is the common gradient.

3. Results

The proposed algorithm is used to improve a shallow neural network (NN) and a DBN [31]. The architectures of the networks are determined by the method in [32]. We use three popular datasets, namely, Iris, Poker Hand, and Forest Fires, from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/datasets.html). With these datasets, we design several experiments to solve typical problems in machine learning, such as problems that involve classification and regression.

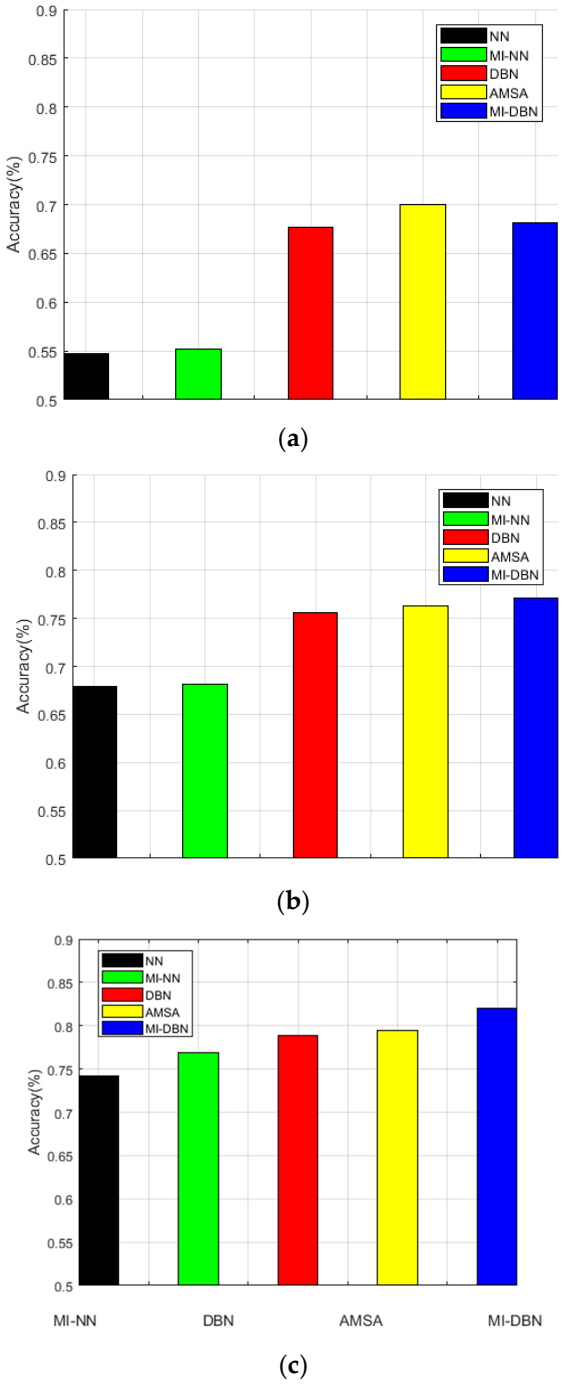
3.1. Classification

In machine learning, classification is the problem of classifying instances into one of two or more classes. For this issue, accuracy is defined as the accuracy rate of the classification.

We designed neural networks as classifiers to solve the classification problems from two datasets, Iris and Poker Hand. For each of these datasets, 50% of the training samples are used for learning.

In these samples, the percent of labeled training samples α samples ranges from 10% to 50%. In the remainder of the dataset, 10% of the samples are randomly selected as test data. The constant λ is set to -0.1.

The test accuracy on the Iris dataset for NN, DBN, AMSA, MI-NN, and MI-DBN ("MI-" denotes the proposed methods) can be seen in Figure 2. The network structures used in this experiment are 4-3-3 and 4-6-7-3, which correspond to the shallow networks (i.e., NN and MI-NN) and the deep networks (i.e., DBN, AMSA, and MI-DBN), respectively. The activation and output function is sigmoid.



(**a**)

(**b**)

(**c**)

Figure 2. The tests of NN, MI-NN, DBN, AMSA, and MI-DBN with Iris. The accuracies are plotted on the horizontal axis. (**a**) $\alpha = 10\%$; (**b**) $\alpha = 30\%$; (**c**) $\alpha = 50\%$.

The test accuracies on the Poker Hand dataset for NN, DBN, AMSA, MI-NN, and MI-DBN ("MI-" denotes the proposed methods) can be seen in Figure 3. The network structures used in this experiment are 85-47-10 and 85-67-48-10, which correspond to the shallow networks (i.e., NN and MI-NN) and the other networks (i.e., DBN, AMSA, and MI-DBN), respectively. The activation and output function is sigmoid.

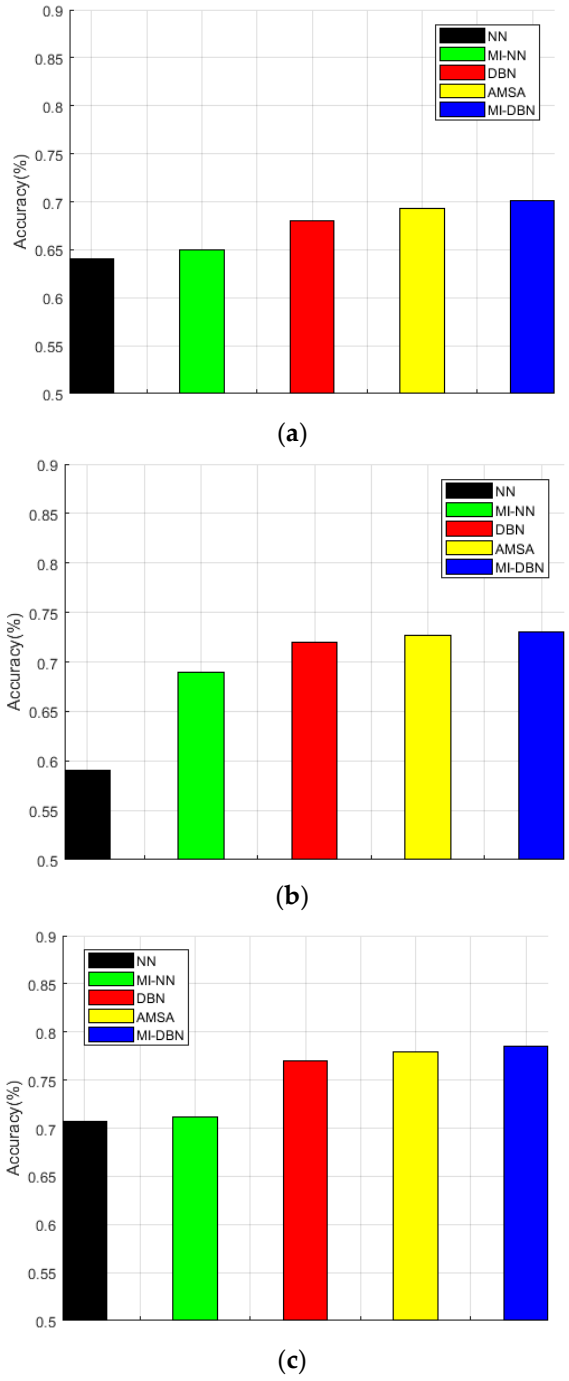
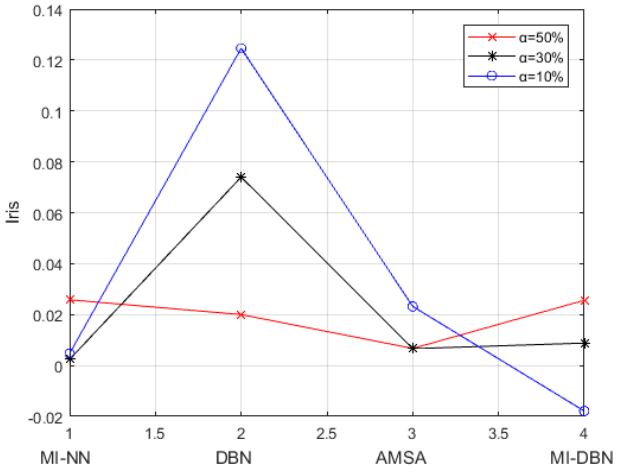


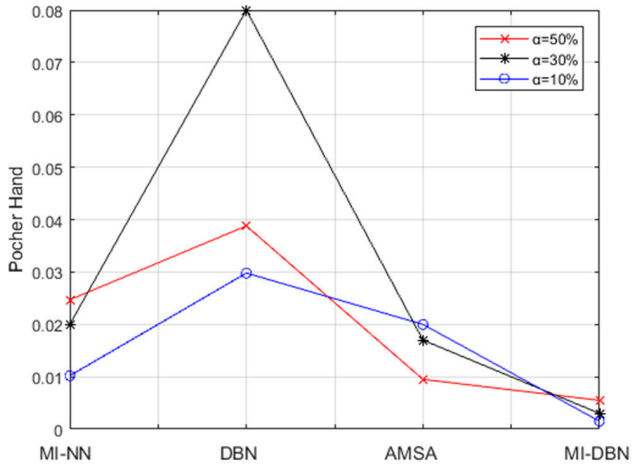
Figure 3. The tests of NN, MI-NN, DBN, AMSA, and MI-DBN with Poker Hand. The accuracies are plotted on the horizontal axis. (**a**) $\alpha = 10\%$; (**b**) $\alpha = 30\%$; (**c**) $\alpha = 50\%$.

It can be seen that the proposed algorithm outperforms the previous methods on both datasets successfully, although a sole exception is that the MI-NN is slightly worse than AMSA with Iris when $\alpha = 10\%$. This finding means that the proposed algorithm can be applied to the problem of

classification. In addition, the proposed algorithm improves the traditional algorithms constantly, when the percent of labeled training samples changes. To study the correlation between the performance and α, we illustrate the difference between the two methods with varied α in Figure 4. According to Figure 4, there is a positive correlation between the improvement brought by the proposed method and the value of α. The reason is that more label data leads the proposed method to absorb more useful information for classification. In contrast, this aspect does not hold true for the other algorithms.



(**a**)



(**b**)

Figure 4. Improvements over the previous methods. (**a**) Iris; (**b**) Poker Hand.

3.2. Regression

In machine learning, regression is a set of statistical processes for estimating the relationships among the variables. In this approach, the accuracy is the error rate of regression substituted by 1.

We design neural networks as regression functions to solve the regression problems in the Forest Fires dataset. For the dataset, 50% of the training samples are used for learning. In these samples, the percent of the labeled training samples α ranges from 10% to 50%. In the remainder of the dataset, 10% of the samples are randomly selected as test data. The constant λ is set to –0.1.

The test accuracy on the Forest Fires dataset for NN, DBN, AMSA, MI-NN, and MI-DBN ("MI-"denotes the proposed methods) can be seen in Figure 5. The network structures used in this experiment are 13-15-1 and 13-15-17-1, which correspond to shallow networks (i.e., NN and MI-NN) and deep networks (i.e., DBN, AMSA, and MI-DBN), respectively. The activation function is sigmoid, and the output function is linear.

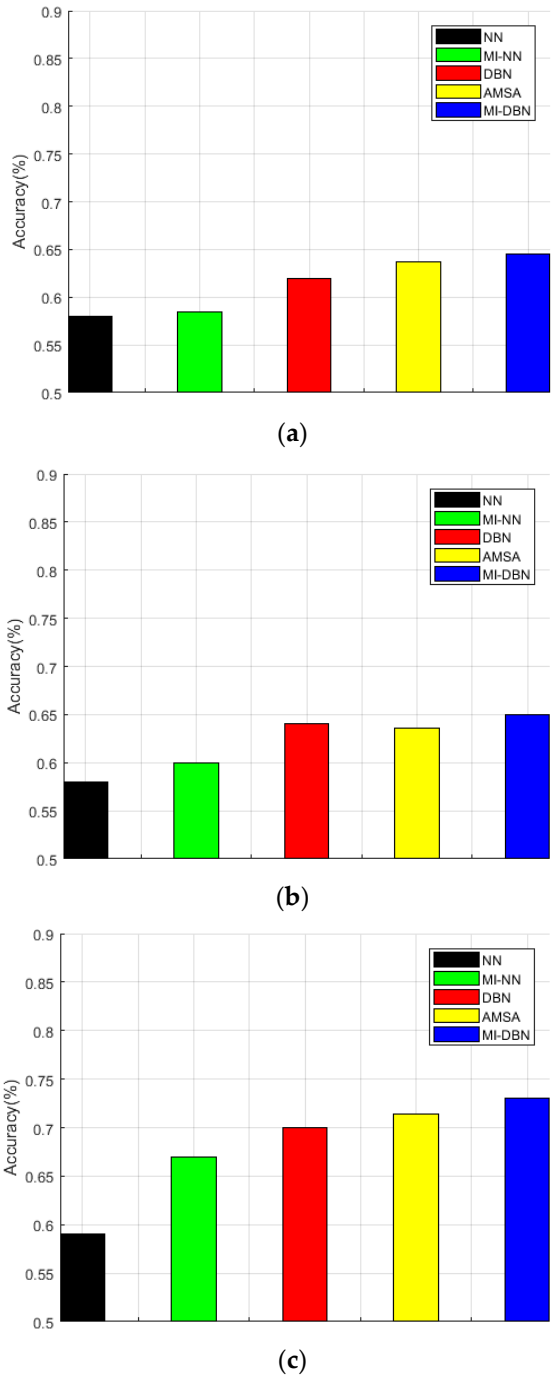


Figure 5. The tests of NN, MI-NN, DBN, AMSA, and MI-DBN with Forest Fires. The accuracies are plotted on the horizontal axis. (**a**) $\alpha = 10\%$; (**b**) $\alpha = 30\%$; (**c**) $\alpha = 50\%$.

We can see that the proposed algorithm outperforms the previous methods on the dataset successfully. This finding means that the proposed algorithm can be applied to the problem of regression. In addition, the proposed algorithm improves the traditional algorithms constantly, when the percent of labeled training samples changes. To study the correlation between the performance and α, we illustrate the difference between the two methods with varied α in Figure 6. According to Figure 6, there is a positive correlation between the improvement brought by the proposed method and the value of α. The reason is that more labeled data leads the proposed method to absorb more useful information for its classification. In contrast, this aspect does not hold true for the other algorithms.

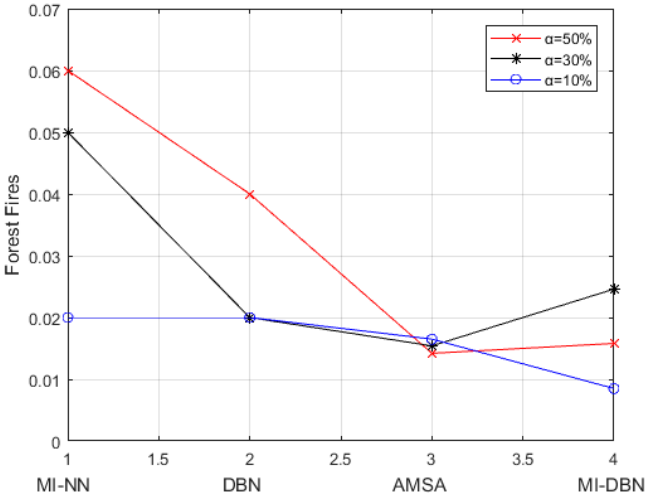


Figure 6. Improvements over previous methods.

4. Conclusions

The proposed algorithm for training neural networks is based on information theory and information geometry. The novel objective function is minimized with the theory of information geometry. The experiments show that the proposed method has better accuracy compared with the existing algorithms.

Author Contributions: Data curation, J.Y.; formal analysis, Y.-H.Z.; methodology, M.W.; supervision, C.-B.X.; validation, J.P.; writing—original draft, M.W.; writing—review and editing, Z.-H.N.

Funding: This research was funded by the Beijing Science and Technology Planning Program of China (Z171100004717001), the Beijing Natural Science Foundation (4172002), and the Natural Science Foundation of China (61701009).

Acknowledgments: This research was supported by the Beijing Science and Technology Planning Program of China (Z171100004717001), the Beijing Natural Science Foundation (4172002), and the Natural Science Foundation of China (61701009).

Conflicts of Interest: We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled.

References

1. Carter, K.M.; Raich, R.; Finn, W.G.; Hero, A.O. Fisher information nonparametric embedding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 2093–2098. [CrossRef] [PubMed]
2. Vapourware, S.; Kar, I.N.; Jha, A.N. Nonlinear System Identification Using Neural Networks. *Int. J. Res.* **2007**, *13*, 312–322.
3. Song, Y.D.; Lewsis, F.L.; Marios, P.; Danil, P.; Dongbin, Z. Guest Editorial Special Issue on New Developments in Neural Network Structures for Signal Processing, Autonomous Decision, and Adaptive Control. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 494–499. [CrossRef]
4. Pardey, J.; Roberts, S.; Tarassenko, L. Application of artificial neural networks to medical signal processing. In Proceedings of the Application of Artificial Neural Networks to Medical Signal Processing, London, UK, 15 December 1994; pp. 9–11.
5. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [CrossRef] [PubMed]
6. Larochelle, H.; Bengio, Y. Classification using discriminative restricted Boltzmann machines (PDF). In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; p. 536.
7. Grossberg, S. Recurrent neural networks. *Scholarpedia.* **2013**, *8*, 18–88. [CrossRef]

8. Yao, X.; Liu, Y. Evolving Neural Network Ensembles by Minimization of Mutual Information. *Int. J. Hybrid Intell. Syst.* **2004**, *1*, 12–21. [CrossRef]
9. Zhang, Z.; Chen, Q.; Qiao, L. A Merging and Splitting Algorithm Based on Mutual Information for Design Neural Networks. In Proceedings of the IEEE Fifth International Conference on Bio-inspired Computing, Theories & Applications, Changsha, China, 23–26 September 2010; pp. 1268–1272.
10. Berglund, M.; Raiko, T.; Cho, K. Measuring the usefulness of hidden units in Boltzmann machines with mutual information. *Neural Netw.* **2015**, *12*, 12–18. [CrossRef] [PubMed]
11. Niles, L.T.; Silverman, H.F.; Bush, M.A. Neural networks, Maximum Mutual Information Training, and Maximum Likelihood Training. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, Albuquerque, NM, USA, 3–6 April 1990; pp. 493–496.
12. Arwini, K.; Dodson, C.T. *Information Geometry: Near Randomness and Near Independence*; Springer: Berlin, Germany, 2008.
13. Frieden, B.R. *Science from Fisher Information: A Unification*; Cambridge University Press: Cambridge, UK, 2004.
14. Kass, R.E.; Vos, P.W. *Geometrical Foundations of Asymptotic Inference*; Wiley: New York, NY, USA, 1997.
15. Amari, S.; Kawanabe, M. Information geometry of estimating functions in semiparametric statistical models. *Bernoulli* **1997**, *3*, 29–54. [CrossRef]
16. Amari, S. Natural gradient works efficiently in learning. *Neural Comput.* **1998**, *10*, 251–276. [CrossRef]
17. Campbell, L.L. The relation between information theory and the differential, geometry approach to statistics. *Inf. Sci.* **1985**, *35*, 199–210.
18. Amari, S. Information geometry on hierarchy of probability distributions. *IEEE Trans. Inf. Theory* **2001**, *47*, 1701–1711. [CrossRef]
19. Amari, S. Dualistic Geometry of the Manifold of Higher-Order Neurons. *Neural Netw.* **1991**, *4*, 443–451. [CrossRef]
20. Zhao, J.; Zhang, C.; Li, W.; Guo, W.; Zhang, K. Natural Gradient Learning Algorithms for RBF Networks. *Neural Comput.* **2015**, *27*, 481–505. [CrossRef] [PubMed]
21. Eleuteri, A.; Tagliaferri, R.; Milano, L. A novel information geometric approach to variable selection in MLP networks. *Neural Netw.* **2005**, *18*, 1309–1318. [CrossRef] [PubMed]
22. Hutter, M. Distribution of Mutual Information. *Adv. Neural Inf. Process. Syst.* **2001**, *10*, 145–167.
23. Axelsson, O. Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations. *Lin. Alg. Appl.* **1980**, *34*, 1–66. [CrossRef]
24. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. Kullback–Leibler Distance. In *Numerical Recipes. The Art of Scientific Computing (3rd ed.)*; Cambridge University Press: Cambridge, UK, 2007.
25. Cox, D.R.; Hinkley, D.V. *Theoritical Statistics*; Chapman & Hall/CRC: Gainesville, FL, USA, 1974.
26. Ravanbakhsh, S.; Poczos, B.; Schneider, J.; Schuurmans, D.; Greiner, R. Stochastic Neural Networks with Monotonic Activation Functions. In Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (AISTATS), Cadiz, Spain, 9–11 May 2016; pp. 135–146.
27. Amari, S. Information geometry of statistical inference-an overview. In Proceedings of the IEEE Information Theory Workshop, Bangalore, India, 20–25 October 2002; pp. 86–89.
28. Menendez, M.L.; Morales, D.; Pardo, L.; Salicru, M. Salicrij. Statistical tests based on geodesic distances. *Appl. Math. Lett.* **1995**, *8*, 65–69. [CrossRef]
29. Cam, L.L. *Asymptotic Methods in Statistical Decision Theory*; Springer: Berlin, Germany, 1986; pp. 618–621.
30. Yuan, Y.X. Step-sizes for the gradient method. *AMS/IP Stud. Adv. Math.* **1999**, *42*, 785.
31. Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *3*, 32–51. [CrossRef] [PubMed]
32. Liu, Y.; Starzyk, J.A.; Zhu, Z. Optimizing number of hidden neurons in neural networks. *Artif. Intell. Appl.* **2007**, *2*, 67–89.