MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

A Deep Neural Network Architecture Using Dimensionality Reduction with Sparse Matrices

Matsumoto, W.; Hagiwara, M.; Boufounos, P.T.; Fukushima, K.; Mariyama, T.; Xiongxin, Z.

TR2016-134 October 2016

Abstract

We present a new deep neural network architecture, motivated by sparse random matrix theory that uses a low-complexity embedding through a sparse matrix instead of a conventional stacked autoencoder. We regard autoencoders as an information-preserving dimensionality reduction method, similar to random projections in compressed sensing. Thus, exploiting recent theory on sparse matrices for dimensionality reduction, we demonstrate experimentally that classification performance does not deteriorate if the autoencoder is replaced with a computationally-efficient sparse dimensionality reduction matrix.

International Conference on Neural Information Processing (ICONIP)

A Deep Neural Network Architecture Using Dimensionality Reduction with Sparse Matrices

Wataru Matsumoto[1], Manabu Hagiwara[2], Petros T. Boufounos[3], Kunihiko Fukushima[1, 4], Toshisada Mariyama[1], Zhao Xiongxin[1]

[1] Mitsubishi Electric Corporation, Information Technology R&D Center, Kanagawa, Japan
[2] Chiba University, Chiba, Japan
[3] Mitsubishi Electric Research Laboratories, Cambridge, MA, USA
[4] Fuzzy Logic System Institute, Fukuoka, Japan

Abstract.
We present a new deep neural network architecture, motivated by sparse random matrix theory that uses a low-complexity embedding through a sparse matrix instead of a conventional stacked autoencoder. We regard autoencoders as an information-preserving dimensionality reduction method, similar to random projections in compressed sensing. Thus, exploiting recent theory on sparse matrices for dimensionality reduction, we demonstrate experimentally that classification performance does not deteriorate if the autoencoder is replaced with a computationally-efficient sparse dimensionality reduction matrix.

Keywords: deep learning, deep neural network, autoencoder, compressed sensing, sparsity recovery, sparse random matrices

1 Introduction

Image and signal classification is one of the most fundamental problems in computer vision and pattern recognition. Recently, deep learning has become a popular solution in this area owing to its superior performance [1, 2]. One of the most commonly used deep learning architectures is the feedforward deep neural network (DNN) [2]. Typical DNNs include a stacked autoencoder as an initial component in their structure. A stacked autoencoder is a multilayer network, trained to reduce data dimensionality while preserving information, thus providing an effective feature extraction and signal representation mechanism for further processing [3].

In a similar vein, compressed sensing (CS) has attracted considerable attention for suggesting that it is possible to overcome the traditional limits of sampling theory. Specifically, Candes et al. [4, 5] and Donoho [6] demonstrated that a signal having a sparse representation can be recovered exactly from a small set of linear, nonadaptive measurements. The CS measurement mechanism is related to Johnson-Lindenstrauss (JL) embeddings [7,8], and has been extended to other signal sets, such as manifolds [9]. The results suggest that it may be possible to sense structured signals by taking

far fewer measurements. Furthermore, Berinde and Indyk [10] demonstrated information embedding and algorithms for sparse recovery, based on sparse random matrices. Such matrices are attractive because of their low computational complexity.

In the remainder of this paper, we demonstrate that efficient randomized dimensionality reduction, using a sparse measurement matrix, can be used to replace the autoencoder stage of a DNN. Thanks to its embedding properties, this dimensionality reduction still preserves the information necessary for classification. The sparsity of the matrix significantly reduces the classification complexity. This paper is organized as follows. Section 2 discusses the motivation and develops the proposed sparse construction using recent theoretical work. Section 3 presents an experimental performance comparison of the proposed sparse construction to conventional networks. Section 4 concludes the document.

2 Deep neural network with sparse construction

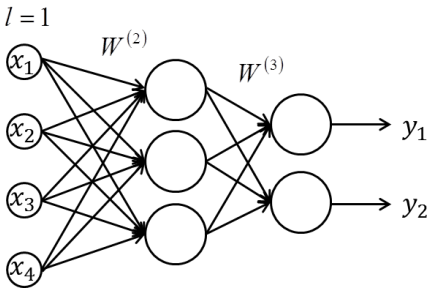

Fig. 1. Multilayer neural network

To establish notation, we first define a deep neural network structure. We use the N-dimensional vector $\mathbf{x} = (x_1, x_2, \cdots, x_N)^{\mathrm{T}} \in \mathbb{R}^N$ to denote the input signal. We assume a multilayer structure, with l denoting the layer's index as $l = 1,2,3, \cdots$. The vector $\mathbf{u}^{(l)} = \left(u_1^{(l)}, u_2^{(l)}, \cdots, u_J^{(l)}\right)^{\mathrm{T}} \in \mathbb{R}^J$ denotes the weighted sum for the l-th layer, computed as $u_j^{(l)} = \sum_{i=1}^{I} w_{ij}^{(l)} x_i^{(l)} + b_i^{(l)}$, where $\mathbf{W}^{(l)} = \begin{bmatrix} w_{11}^{(l)} & \cdots & w_{1I}^{(l)} \\ \vdots & \ddots & \vdots \\ w_{J1}^{(l)} & \cdots & w_{JI}^{(l)} \end{bmatrix}$ is the weight matrix and $\mathbf{b}^{(l)} = \left(b_1^{(l)}, b_2^{(l)}, \cdots, b_J^{(l)}\right)^{\mathrm{T}} \in \mathbb{R}^J$ is the bias vector. Given $u_j^{(l)}$, the activation function f produces the input vector $x_j^{(l+1)}$ for the next (i.e., $l+1$-th) layer, using the element-wise computation $x_j^{(l+1)} = f\left(u_j^{(l)}\right)$. To simplify discussion, in the remainder of this paper we assume $b_j^{(l)} = 0$ and $(u) = u$.

2.1 Similarity of autoencoder and compressed sensing

A stacked autoencoder is pre-trained using unsupervised learning, separately from the DNN used for classification. The goal is to convert high-dimensional input data to low dimensional features that capture the salient information of the input and can be used for classification. Each layer is typically trained by minimizing the discrepancy between the input data and its reconstruction using the autoencoder. This training is applied layer by layer from bottom to top using gradient descent and backpropagation. However, this works well only if the initial weights are close to a reasonable solution.

In this section, motivated by sparse matrix theory, we describe an effective method of initializing the weights to construct deep neural networks. Instead of a dense autoencoder, we learn and use low-dimensional sparse codes to reduce data dimensionality. The implicit regularization also removes the requirement to use dropout during learning.

Given a network layer, $\mathbf{x}^{(l+1)} = \mathbf{W}^{(l)}\mathbf{x}^{(l)}$, the recovered vector $\hat{\mathbf{x}}^{(l)}$ is generated from $\mathbf{x}^{(l)}$ using a weight matrix $\widehat{\mathbf{W}}^{(l)}$ by computing $\mathbf{x}^{(l)} = \widehat{\mathbf{W}}^{(l)}\mathbf{x}^{(l+1)}$. Training estimates the weight matrices $\mathbf{W}^{(l)} \approx \widehat{\mathbf{W}}^{(l)}$ and $\mathbf{x}^{(l+1)}$ by solving the optimization

$$\min_{\{\hat{\mathbf{x}}^{(l)}\}} \left\|\mathbf{x}^{(l)} - \hat{\mathbf{x}}^{(l)}\right\|_2^2 = \min_{\{\hat{\mathbf{x}}^{(l)}\}\{\mathbf{x}^{(l+1)}\}} \left\|\mathbf{x}^{(l)} - \hat{\mathbf{x}}^{(l)}\right\|_2^2.$$

We use $J^{(l)}$ to denote the length of $\mathbf{x}^{(l)}$. In general, $J^{(l+1)} \leq J^{(l)}$, so that the autoencoder reduces the dimensionality of the data, similar to CS measurements. That is, given a measured signal $\mathbf{x}^{(l+1)}$, the autoencoder problem can be regarded as a problem to recover the original signal $\mathbf{x}^{(l)}$ using a measurement matrix $\mathbf{W}^{(l)}$.

An accurate reconstruction means that the exact signal is reconstructed. The problem is how to design the measurement matrix $\mathbf{W}^{(l)}$. To solve this problem, we first review the CS literature for desirable properties of $\mathbf{W}^{(l)}$.

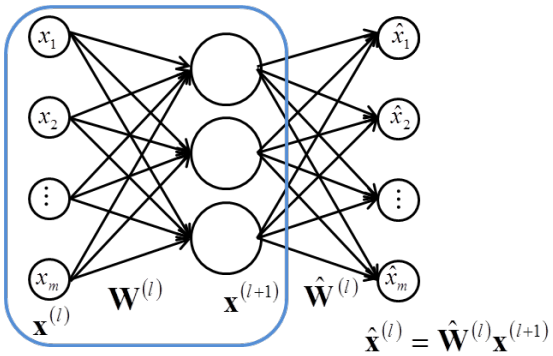

Fig. 2. Structure of autoencoder

Compressed sensing can be regarded as a generalization of Nyquist-Shannon sampling for structured signals - typically sparse signals that have very few non-zero entries. In the Nyquist-Shannon sampling theorem, signals, images, videos, and other

data can be exactly represented with a set of uniformly spaced samples taken at the Nyquist rate, i.e., twice the highest frequency present in the signal of interest. Rather than sampling at specific points in times, CS, instead, uses a few generalized linear measurements of the signal, exploiting the fact that most signals of interest exhibit structure such as sparsity that can be used in the reconstruction.

For example, the majority of natural images are characterized by large smooth or textured regions and relatively few sharp edges. In the wavelet transform of a typical natural image, most coefficients are very small [11]. Hence, we can obtain a good approximation of the signal by setting the small coefficients to zero to obtain a *K*-sparse representation. We say that a vector $\mathbf{x}$ is *K*-sparse if it contains at most *K* non-zero entries.

Recognition for a neural network can be considered as a classification problem, under a signal model that exhibits some structure such as sparsity. In this sense, the autoencoder has the same goal as compressive measurements: reduce the dimensionality of the original signal while preserving the appropriate information.

2.2 Sparse recovery using sparse random matrices

In the subsequent discussion, we consider a single layer and sparse signal models. We first review sparse recovery guarantees that ensure the preservation of information. To simplify the notation, we replace $\mathbf{x}^{(l+1)} = \mathbf{W}^{(l)}\mathbf{x}^{(l)}$ with $\mathbf{y} = \mathbf{W}\mathbf{x}$. We assume deterministic signals, in which $\mathbf{x} \in \mathbb{R}^{\boldsymbol{N}}$ is a fixed but unknown vector with exactly *K* non-zero entries. We use the set $S \coloneqq \mathrm{supp}(\mathbf{x})$ to denote the support of $\mathbf{x}$, i.e, the location of non-zeros. Note that there are $M = \binom{N}{K}$ possible support sets, corresponding to the *M* possible *K*-dimensional subspaces in which $\mathbf{x}$ may lie. We are given a vector of m noisy observations $\mathbf{y} \in \mathbb{R}^{\boldsymbol{m}}$, of the form

$$\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{V},$$

where $\mathbf{W} \in \mathbb{R}^{m \times N}$ is the measurement matrix, and $\mathbf{V} \sim N(0, \sigma^2 \mathbf{I}_{M \times M})$ is additive Gaussian noise. Throughout this paper, we assume, without loss of generality, that $\sigma^2 = 1$, since any scaling of σ can be accounted for in the scaling of $\mathbf{x}$.

We consider exact recovery of the support set S, which corresponds to model selection. More precisely, we measure the error between the estimate $\hat{\mathbf{x}}$ and the true signal $\mathbf{x}$ using the $\{0,1\}$-valued indicator loss function of the support:

$$\rho(\hat{\mathbf{x}}, \mathbf{x}) \coloneqq \mathbb{I}\left[\{\hat{x}_i \neq 0, \forall i \in S\} \cap \left\{\hat{x}_j = 0, \forall j \notin S\right\}\right].$$

A decoder is a mapping g from observations $\mathbf{y}$ to a support estimate $\hat{S} = \mathrm{g}(\mathbf{y})$. We are interested in both sparse and dense encoding matrices, denoting by γ the sparsification fraction of the matrix, i.e., the fraction of non-zero elements.

A general performance lower bound for arbitrary decoders is developed in [11]. Specifically, let $\mathbb{P}[\mathrm{g}(\mathbf{y}) \neq S | S]$ be the conditional probability of error given that the true support is S. Assuming that $\mathbf{x}$ has support S chosen uniformly at random over the M possible subsets of size K, the average probability of error is given by

$$p_{err} = \frac{1}{\binom{N}{K}} \sum_S \mathbb{P}[\mathrm{g}(\mathbf{y}) \neq S|S].$$

We say that sparsity recovery is asymptotically reliable if $p_{err} \to 0$ as $m \to \infty$. Since our goal is *exact* support recovery from noisy measurements, the minimum value of $\mathbf{x}$ on its support is important,

$$x_{\min} \coloneqq \min_{i \in S} |x_i|.$$

Thus, given $x_{\min}$, we can define a signal class

$$C(x_{\min}) \coloneqq \{\mathbf{x} \in \mathbb{R}^N | |x_i| \geq x_{\min} \; \forall i \in S\}.$$

For this class, [11] derives the necessary conditions on ters $(m, N, K, x_{\min}, \gamma)$, such that a decoder with asymptotically reliable recovery can exist, regardless of its computational complexity. The lower bounds describe the required number of measurements *m,* in general settings, where both the signal sparsity K, and the measurement sparsity γ are allowed to scale with the signal dimension N. The analysis in [11] applies to random ensembles of measurement matrices $\mathbf{W} \in \mathbb{R}^{m \times N}$, where each entry w_{ij} is drawn i.i.d. from some underlying distribution. The most commonly used ensemble is the standard Gaussian distribution, in which $w_{ij} \sim N(0,1)$. This choice generates a dense measurement matrix $\mathbf{W}$, with mN non-zero entries. Theorem 1 in [11] applies to more general ensembles satisfying the moment conditions $\mathbb{E}[x_{ij}] = 0$ and $\mathrm{var}(x_{ij}) = 1$, allowing for a variety of non-Gaussian distributions (e.g., uniform, Bernoulli). Further, Theorem 2 in [11] is derived for γ-sparsified matrices $\mathbf{W}$, in which each entry w_{ij} is i.i.d., drawn according to

$$w_{ij} = \begin{cases} N\left(0, \frac{1}{\gamma}\right) & \text{w. p. } \gamma \\ 0 & \text{w. p. } 1-\gamma \end{cases} . \quad (1)$$

Note that when $\gamma = 1$, $\mathbf{W}$ is the standard Gaussian ensemble. We refer to the sparsification parameters $0 \leq \gamma \leq 1$ as the measurement sparsity. The analysis allows this parameter to vary as a function of (m, N, K).

2.3 Bounds on dense ensembles and sparse ensembles

Theorem 1 in [11] provides a necessary condition for asymptotically reliable recovery on measurement matrices with dense ensembles.

Theorem 1 [11] (general ensembles). Let the measurement matrix $\mathbf{W} \in \mathbb{R}^{m \times N}$ be drawn with i.i.d. elements from any distribution with zero-mean and variance one. Then a necessary condition for asymptotically reliable recovery over the signal class $C(x_{\min})$ is

$$m > \max\{f_1(N, K, x_{\min}), f_2(N, K, x_{\min}), k - 1\},$$

where

$$f_1(N,K,x_{\min}) := \frac{\log\binom{N}{K}-1}{\frac{1}{2}\log\left(1+Kx_{\min}^2\left(1-\frac{K}{N}\right)\right)}$$

$$f_2(N,K,x_{\min}) := \frac{\log(N-K+1)-1}{\frac{1}{2}\log\left(1+x_{\min}^2\left(1-\frac{1}{N-K+1}\right)\right)}.$$

Furthermore, Theorem 2 in [11] provides a necessary condition for asymptotically reliable recovery on measurement matrices with sparse ensembles.

Theorem 2 and Corollary 2 [11] (sparse ensembles). Let the measurement matrix $\mathbf{W} \in \mathbb{R}^{m\times N}$ be drawn with i.i.d. elements from the γ-sparsified Gaussian ensemble (1). A necessary condition for asymptotically reliable recovery over the signal class $C(x_{\min})$ is

$$m > \max\{g_1(N,K,x_{\min},\gamma), g_2(N,K,x_{\min},\gamma), k-1\},$$

where in general

$$g_1(N,K,x_{\min},\gamma) \geq \frac{\log\binom{N}{K}-1}{\frac{1}{2}\log(1+Kx_{\min}^2)}$$

$$g_2(N,K,x_{\min},\gamma) \geq \frac{\log(N-K+1)-1}{\frac{1}{2}\log(1+x_{\min}^2)}.$$

In general, $N \gg K$, hence $1 \gg \frac{K}{N}$, $1 \gg \frac{1}{N-K+1}$. Under this assumption, from Theorem 1 and 2, and Corollary 2, we can derive the following necessary condition:

$$\begin{aligned} m &> \max\{f_1(N,K,x_{\min}), f_2(N,K,x_{\min}), k-1\} \\ &\approx \max\{g_1(N,K,x_{\min},\gamma), g_2(N,K,x_{\min},\gamma), k-1\}, \end{aligned}$$

where in general

$$g_1(N,K,x_{\min},\gamma) \geq \frac{\log\binom{N}{K}-1}{\frac{1}{2}\log(1+Kx_{\min}^2)}$$

$$g_2(N,K,x_{\min},\gamma) \geq \frac{\log(N-K+1)-1}{\frac{1}{2}\log(1+x_{\min}^2)}.$$

That is, the number of observations necessary is almost independent of whether we use dense or sparse ensembles. Thus, even if we use sparse matrices as measurement matrices, in principle, we can recover the original signals. We exploit this to provide an effective method of initializing the weights that uses sparse random matrices to construct deep neural networks that use low-dimensional codes as a tool to reduce the dimensionality of the data, instead of using an autoencoder.

3 Performance evaluation and comparison

We performed experiments that compared the performance of sparse matrices with those of dense, Gaussian matrices. In our experiments we generated γ-sparsified ma-

trices **W** of m rows and N columns, in which each entry w_{ij} was i.i.d. drawn according to (1). Note that when $\gamma = 1$, **W** is exactly the standard Gaussian ensemble. d is the average non-zero entries in each column for **W**, so that $\gamma = \frac{d}{m}$. To evaluate our experiment we used the MNIST handwritten digit dataset containing ten classes (0-9) consisting of 50,000 training, 10,000 validation and 10,000 test images [13]. The digits were size-normalized and centered in 28 x 28 grayscale images.

3.1 Unsupervised feature learning results

To compare the quality of the features learnt by sparse matrices with dense matrices, we first extracted features using the unsupervised learning algorithm and compared the reconstruction mean square error (MSE) $\frac{1}{N}\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2$. The network had 784 input units (28 x 28 grayscale image, normalized to values ranging from [0,1]). These were connected to 500 hidden units, so that $N = 784, m = 500$. For the unsupervised training, we set the momentum to $\tau_t = 0.5$, the learning rate to $\eta_t = 0.1$, and trained for 10,000 epochs.

The MSE results for dense and different sparse matrices are compared in Table 1. We can see that the final MSE of the unsupervised learning algorithm with dense and sparse matrices is almost the same, approximately 5.0e-4.

Table 1. MSE of unsupervised learning algorithm with dense and sparse matrices

Sparsity	$\gamma = 1$	$\gamma = 0.33$	$\gamma = 0.16$	$\gamma = 0.081$	$\gamma = 0.04$
	d = 784	d = 256	d = 128	d = 64	d = 32
MSE	5.29e-4	5.12e-4	4.95e-4	5.19e-4	5.88e-4

3.2 Supervised learning results

In supervised learning, it is a common practice to use the weights generated by the unsupervised learning method to initialize the early layers of a multilayer network. A discriminative algorithm is then used to adjust the weights of the last hidden layer and also to fine tune the weights in the previous layers.

The accuracy results for dense and different sparse matrices are compared in Table 2. As is evident, the performance of supervised learning with dense and sparse matrices is almost the same, with average accuracy approximately 98% for the ten digit classes (0-9).

Table 2. Average accuracy of unsupervised learning algorithm with dense and sparse matrices

Sparsity	$\gamma = 1$	$\gamma = 0.33$	$\gamma = 0.16$	$\gamma = 0.081$	$\gamma = 0.04$
	d = 784	d = 256	d = 128	d = 64	d = 32
Average accuracy	98.38%	98.34%	98.13%	98.17%	97.9%

4 Conclusion

In this work, we proposed a sparse coding method with sparse measurement matrices for deep neural networks. We described an effective method of initializing the weights that permits sparse random matrices to construct deep neural networks to learn low-dimensional codes as a tool to reduce the dimensionality of data for autoencoder. The resultant neural network was constructed with extremely sparse edge connecting units for each layer. This proposal can provide benefits such as the reduction of training and inference time, reduction of the number of free parameters, and the enhancement of the generalization capability.

References

1. Fukushima K. "Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position", *Biological Cybernetics*, vol. 36, no. 4, pp 193-202 ,1980.
2. G. E. Hinton, Osindero S., and Teh Y. "A fast learning algorithm for deep belief nets", *Neural Computation*, vol. 18, pp. 1527-1554, 2006.
3. G. E. Hinton and R. R. Salakhutdinov : "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313*,* 28 July, 2006.
4. E. Candes, J. Romberg, and T. Tao. "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, 59(8):1207-1223, 2006.
5. E. Candes and T. Tao. "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inform. Theory*, 52(12):5406-5425, 2006.
6. D. Donoho. "Compressed sensing," *IEEE Trans. Inform. Theory*, 52(4):1289-1306, 2006.
7. Johnson, W. B. and Lindenstrauss, J., "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary mathematics*, 26:189-206, 1984.
8. Baraniuk R. G., Davenport M., DeVore R. and Wakin M. "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, 28(3): 253-263, 2008.
9. Baraniuk R. G., and Michael B. W. "Random projections of smooth manifolds," *Foundations of computational mathematics*, 9(1): 51-77, 2009.
10. Berinde R. and Indyk P., "Sparse recovery using sparse random matrices," *CSAIL Technical Report, MIT-CSAIL-TR-2008-001*, 2008.
11. S. Mallat. "A Wavelet Tour of Signal Processing," *Academic Press*, San Diego, CA, 1999.
12. W. Wang, M.J.Wainwright, K.Ramchandran, "Information-theoretic limits on sparse recovery: Dense versus sparse measurement matrices," *Technical Report*, Department of Statistics, UC Berkeley, May, 2008.
13. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE*, 86(11):2278-2324, November 1998.