

Why Deep Learning Works: A Manifold Disentanglement Perspective

Pratik Prabhanjan Brahma, Dapeng Wu, *Fellow, IEEE*, Yiyuan She

Abstract—Deep hierarchical representations of the data have been found out to provide better informative features for several machine learning applications. In addition, multi-layer neural networks surprisingly tend to achieve better performance when they are subject to an unsupervised pre-training. The booming of deep learning motivates researchers to identify the factors that contribute to its success. One possible reason identified is the flattening of manifold-shaped data in higher layers of neural networks. However, it is not clear how to measure flattening of such manifold-shaped data and what amount of flattening a deep neural network can achieve. For the first time, this paper provides quantitative evidence to validate the flattening hypothesis. To achieve this, we propose few quantities for measuring manifold entanglement under certain assumptions and conduct experiments with both synthetic and real-world data. Our experimental results validate the proposition and lead to new insights on deep learning.

Index Terms—Deep Learning, Manifold Learning, Unsupervised Feature Transformation, Disentanglement

I. INTRODUCTION

THE advent of artificial neural networks ignited a quantum shift from linear machine learning methods. Neural networks, using multiple hidden layers, have very high capacity to model highly varying functions defining the non-linear structure of input data. However, multi-layer neural networks face a critical problem that the lower layers remain under-trained as their corresponding gradients get diminished while using backpropagation [1]. Very often, a neural network gets stuck in either a local minima or a saddle point [2] and the adaptive gradient descent fails to move on from there. Although several heuristic solutions have been suggested, the problems of poor local minima and over-fitting in neural networks have been among the primary reasons that led to the subsequent decrease in popularity. Kernel methods [3], with a strong theoretical foundation, paved the way for the third generation machine learning algorithms and proved to be the state-of-the-art in cases like speaker identification [4] and text categorization [5]. These methods project the data to a very high (even infinite) dimensional space without explicitly worrying about the mapping function and comfortably assume that linear regression and classification methods can be used consequently. However, these methods depend on using a

kernel function that is usually pre-defined and hence, in general, not data-adaptive. The discussion in [6] shows that a large number of training examples is required in case of local kernels, like Gaussian, even if the Kolmogorov complexity of the target function may be low. Recent research works [7] have pointed out that deep architectures, such as multi-layer neural networks, showed better performance when the traditional backpropagation algorithm was preceded by an unsupervised pre-training. These designs can also be thought of as kernel methods with a data dependent Gram matrix. This brought back the interest of the academic fraternity into multi-layer neural networks again and deep learning methods [4] have since excelled in being the top performer in many applications such as object detection and recognition, handwriting recognition [8], speaker identification [9] and text mining [7].

Though a concrete mathematical reasoning of the success behind these techniques is yet unclear, several intuitive reasons have been articulated in [10] and [11]. One of the ideas is based on the manifold perspective where it is hypothesized that deeper layers can help extract the underlying factors of variations that define the structure of the data geometrically. Through the learning process, the data manifolds are unfolded and flattened so that subsequent processing using the output features can be made easier. A feature space that can correctly represent the hidden factors underlying observed data is better to deal with for further learning [12]. For example in prediction or interpolation, if we can flatten [13] or linearize the complicated manifolds along the factors that led to the generation of data, then the task can be performed easily using even linear regression techniques. Researchers, like in [14], have attempted to measure invariance across layers in a deep network. However, these measures are not based on the structural connectivity of the data points and may not be suitable to quantify the extent to which disentanglement of manifolds is achieved under the deep learning paradigm. Novel degrees of entanglement, based on the manifold hypothesis, have been designed and tested in this work to quantitatively show the unfolding of class specific manifolds in the data.

The remainder of the paper is organized as follows. Following the introduction, Section II deals with a brief introduction to deep learning methods and especially the unsupervised pre-training techniques. Section III lays out the fundamental definitions of manifolds and explains the popular hypothesis of real world data containing intrinsically low dimensional manifolds embedded in high dimensional space. In Section IV, we design metrics from a differential geometry perspective to quantify the entanglement of a nonlinear manifold with respect

Pratik Prabhanjan Brahma and Dapeng Wu are with Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611. (email: prprbr@ufl.edu; email: wu@ece.ufl.edu).

Yiyuan She is with Department of Statistics, Florida State University, Tallahassee, FL 32306. (email: yshe@stat.fsu.edu).

This work was supported in part by National Science Foundation under Grant DMS-1352259 and NSFC under grant 61529101.

to a completely flat or linear subspace. These measures are then used in Section V on features obtained from different layers of deep neural networks when applied onto synthetic and real world image datasets. Finally, we conclude our paper in Section VI summarizing the various remarks observed from the experimental results.

II. DEEP LEARNING METHODS

Multi-layer neural networks have large capacity to model high degree of non-linearity in the input data but face the problems of poor training of lower layers, getting stuck in unwanted local optima and over-fitting. However, it was the unsupervised pre-training introduced in [7] that gave a new dimension to the deep learning research. It discusses an undirected graphical model, called Boltzmann machine, consisting of hidden (h_i) and visible (x_i) nodes. The joint probability of the clique is given as

$$P(x, h) = \frac{e^{-E(x, h)}}{Z} \quad (1)$$

where $E(x, h)$ is the energy function describing the interactions in the graphical model that behaves as a Markov Random Field. Z is the partition function which acts as a normalization to make $P(x, h)$ a probability measure. The energy function for a restricted Boltzmann machine (RBM), where there are no intra-hidden or intra-visible interactions, is given by a simple equation

$$E(x, h) = -b'x - c'h - h'Wx \quad (2)$$

where $\Theta = \{b, c, W\}$ is the set of model parameters. Due to the bipartite restrictions put on an RBM, the hidden nodes are independent of each other when conditioned on the visible nodes and vice-versa as shown in

$$P(h|x) = \prod_{i \in Hid} P(h_i|x). \quad (3)$$

The individual conditional probabilities turn out to be simple sigmoid function when both the hidden and visible nodes are conditioned to be taking only binary values:

$$P(h_i|x) = \text{sigm}(c_i + \sum_{j \in Vis} W_{ji}x_j). \quad (4)$$

In general, it is desired to create a generative model so that the hidden nodes can learn to generate new model data in accordance to the actual probability density function (PDF) of the input data. Thus, its log likelihood is set to be maximized in the objective function. Although it involves intractable parameters like the partition function Z , a slightly biased sampling procedure called contrastive divergence [15] is used by constructing model visible and hidden features through consecutive Gibbs sampling. The overall gradient of the log-likelihood, that is to be used as the weight update, is given as

$$\Delta w_{ij} = \eta(\langle h_i x_j \rangle_{\text{original}} - \langle h_i x_j \rangle_{\text{reconstruction}}) \quad (5)$$

where \langle, \rangle (inner product) calculates the number of times the i^{th} hidden and j^{th} visible nodes are simultaneously on. Similarly, the biases b and c can also be trained iteratively

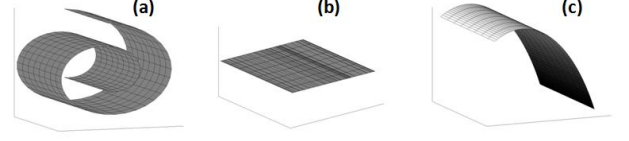


Fig. 1. Comparison of differently folded manifolds.

using the gradient information. The above analysis can also be extended to both hidden and visible nodes being characterized by Gaussian random variables by changing the energy function to

$$E(x, h) = \sum_{vis} \frac{(x_i - a_i)^2}{2\sigma_i^2} + \sum_{hid} \frac{(h_j - b_j)^2}{2\sigma_j^2} - \sum_{i,j} W_{ij} \frac{x_i h_j}{\sigma_i \sigma_j}. \quad (6)$$

Such a graphical model can be trained using the Continuous RBM learning algorithm given in [16].

The concept of depth in a deep belief network (DBN) comes from stacking RBMs on top of one another and greedily training them. The weights and biases obtained after the unsupervised pre-training then act as a starting set up for a neural network that can be further trained using supervised backpropagation. It has been shown to serve as a better initialization as compared to traditionally performed random initialization. Out of the several intuitions that have come up on why such an unsupervised pre-training helps, we hereby state and experimentally validate one that is based on disentanglement of manifolds present in the data.

Other than the deep belief network approach discussed above, there are other popular designs for multi-layer neural architectures [4] too like Stacked Denoising AutoEncoders (SDAE), Deep Boltzmann Machine (DBM) and Convolutional Neural Networks (CNN). Apart from neural networks, the stacking blocks can also be replaced with random projections [17] or spectral embedding too. Most of the deep learning paradigms share an unsupervised pre-training method and it is thus important to understand the impact of this stage on overall model's behavior. As will be shown later, it is the amalgamation of generative and discriminative model design that leads to better performance in any machine learning task. Although only the DBN architecture is considered for all the experiments in our current work, the performance evaluation metrics developed here can also be used to test other kinds of deep learning designs like the ones mentioned above.

III. MANIFOLD CONCEPTS

Definition 1. A topological space [18] (set of points along with neighborhood for each point) is a topological manifold M if and only if:

- 1) M is Hausdorff (distinct points have disjoint neighborhood).
- 2) M is second countable (there exists a countable basis to the topology of M).
- 3) M is also locally Euclidean, i.e. $\forall p \in M, \exists$ an open set $U \subset M$ containing p that has a homeomorphism

$\varphi : U \rightarrow V$ (a mapping function with a continuous inverse) with an open set $V \subset \mathbb{R}^n$.

As an example shown in the leftmost figure of Fig. 1, a Swiss roll in \mathbb{R}^3 is actually a 2-manifold. A pair (U, φ) is called a *coordinate chart*. An *atlas* for M is defined to be a collection of charts whose domains cover the entire M . An atlas A is called a *smooth atlas* if any two charts in A are smoothly compatible with each other.

Definition 2. A smooth manifold is a topological n -manifold with a smooth atlas A of M .

The sphere S^2 lying in \mathbb{R}^3 is a smooth 2-manifold. The rest of the paper assumes that data in each of the categories, like all images belonging to a single person, lie on a smooth connected manifold.

Most real world data are assumed to lie in a complicated nonlinear manifold fashion. The manifold hypothesis [19] suggests that complex data manifolds are intrinsically low dimensional. It means the manifold has a local homeomorphism with a Euclidean space of dimensionality lower than the original space it is embedded in. For instance, the pictures of a human face may have pixels of the order of millions depending on the resolution of the camera. However, the images can be characterized using a few factors of variation [20] like lighting, orientation, shades, shape, pose, etc. According to the hypothesis, if the underlying factors of variation can be discovered appropriately, the projected data in its intrinsic form will exhibit flatter or linearized geometry and learning tasks like classification become easier, possibly using even a linear classifier. This has been the basis for several nonlinear dimensionality reduction techniques. A linear manifold is rather less complicated to deal with and is actually just a linear subspace that has possibly been shifted away from the origin $\mathbf{0}$.

Definition 3. A non-empty subset L of a vector space V is a linear subspace or a linear manifold if along with every pair, x and y , of vectors contained in L , every linear combination $\alpha x + \beta y$ is also contained in L . While a linear subspace has to contain the $\mathbf{0}$ vector in it, the linear manifold may or may not pass through the origin.

Points and lines in \mathbb{R}^2 and points, lines and flat planes in \mathbb{R}^3 are examples of linear manifolds. Learning on such hyperplanes is comparatively easier than non-linear manifolds, be it for classification and regression, as conventional linear methods can be directly applied.

For non-linear manifolds, various manifold learning algorithms [19] like ISOMAP, Local Linear Embedding, Laplacian Eigenmaps and others have been developed. These methods have been shown to clearly outclass their linear counterparts like Principal Components Analysis (PCA). A simple pictorial illustration is given in Fig. 1 where a swiss roll in (a), intrinsically 2D, cannot be reduced to its original \mathbb{R}^2 version by just applying PCA. However, when it is unfolded into linear varieties (b) or even substantially disentangled versions as in (c), the task of subspace learning is made easier. It is worth noting here that a 2 dimensional PCA projection of (c) can also

serve the purpose if, for instance, classification between two half portions of the swiss roll was the learning objective. Most of these manifold learning methods try to project an isometric embedding of the data onto a low dimensional space but fail to produce an explicit feature transformation function that can be readily applied onto a new test data. Also, these methods are severely limited by the choice of neighborhood.

It was first proposed in [11] that regularized autoencoders tend to disentangle the complex manifold structures in data by unfolding and volume expansion. They showed that linear mixing between feature vectors in the higher layers can help generate more plausible training examples using the decoder. This is possible only when the projection of the manifold in the higher layers is linear as suggested in Definition 3. The data exhibits high probability concentration along the low-dimensional manifold and it drastically reduces as one goes away from it. The intuition is that a hidden representation can capture the variations along the manifold in input space and ignore the variations orthogonal to the tangent spaces of the manifold. Since an RBM or any regularized autoencoder is designed on the connection between a set of hidden and visible variables, it is argued in [11] that the hidden variables can be representative of the governing variations in the visible input space. However, it is worth investigating how much success is achieved by a deep architecture in the objective of linearizing the data manifolds. In a classification paradigm, each class present in the data can be modeled as a nonlinear manifold itself and thus the data is a composition of jumbled manifolds, just like each mode in a multi-modal distribution. Disentanglement aims at extracting the inherent features that can then be used as input to a simple classifier. It is worth mentioning here that just unfolding alone may not necessarily mean better classification as the disentangled linear hypersurfaces may still be intersecting. However, as it is shown as a sanity check in [10], the deep network unfolds as well as separates the class structures. For the rest of the paper, the concept of disentanglement can be understood as flattening as well as separating the individual data manifolds for respective classes. As will be shown later in this current work, disentanglement is maintained and even pronounced after discriminative backpropagation which helps in achieving good classification.

IV. MEASURES OF ENTANGLEMENT

In this section, we are interested in indicating how much the individual class manifolds have unfolded with respect to the original structures formed by the actual data points. In order to quantify this process, we make use of distances between the training examples. Distances can be Euclidean (l_2 norm of the distance vector) or can be geodesic in nature.

Definition 4. A geodesic distance [18] on a Riemannian manifold M between two points p and q on the manifold is defined as the infimum of the lengths taken over all piecewise continuously differentiable curves $\gamma : [a, b] \rightarrow M$ where $\gamma(a) = p$ and $\gamma(b) = q$.

In other words, these distances are locally length-minimizing paths inscribed on the manifold. Although in-

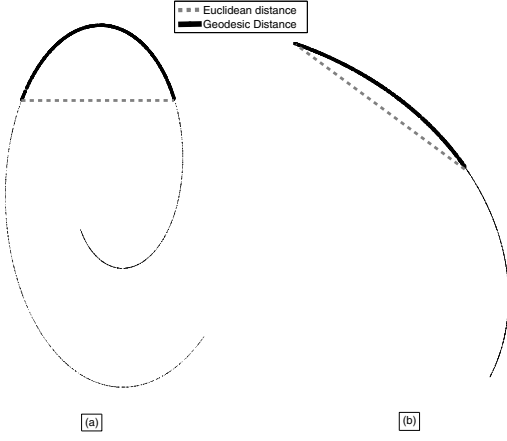


Fig. 2. Geodesic and Euclidean distances for (a) entangled and (b) slightly disentangled manifolds.

tractable without given the exact function defining the geometry, it can be approximated by using the graph geodesic distance [21]. A nearest neighbor graph can be formed out of the given P -dimensional data samples $x_i \in \mathbb{R}^P$ and the shortest path between two points is a geodesic path approximately traversed along the manifold surface. Let the total number of data samples be N . The sum of the individual Euclidean distances serves as an estimate for the manifold distance as given by

$$G_M(i, j) = \sum_k |e_k| \quad (7)$$

where $|e_k|$ denotes the lengths of the k edges contained in the shortest path connecting x_i and x_j data nodes. The pairwise euclidean distances are stored in an $N \times N$ matrix as

$$G_E(i, j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2. \quad (8)$$

Let \mathbf{r}_E and \mathbf{r}_M be vectors formed by concatenation of the upper-triangular entries of symmetric matrices G_E and G_M respectively. Thus, both \mathbf{r}_E and $\mathbf{r}_M \in \mathbb{R}^{N(N-1)/2}$. The Euclidean distance is also the geodesic distance between two points lying on a linear manifold. As can be seen in Fig. 2, the geodesic and euclidean distances tend to be closer to each other as the manifold gets more and more unfolded. The first measure, based on similarity between geodesic and euclidean distances, is the residual normalized cross-correlation between \mathbf{r}_E and \mathbf{r}_M . Let c_k be defined as

$$c_k = 1 - \frac{(r_M(k) - \mu_{r_M})(r_E(k) - \mu_{r_E})}{\sigma_{r_M} \sigma_{r_E}} \quad (9)$$

where μ_{r_M} and μ_{r_E} are the means and σ_{r_M} and σ_{r_E} are the standard deviations of \mathbf{r}_E and \mathbf{r}_M respectively. The first measure can then be written as

$$\tilde{c} = \frac{2}{N(N-1)} \sum_k c_k = E[c]. \quad (10)$$

The second measure is again another average value of the difference between distinct ($i \neq j$) pair-wise euclidean and

geodesic distances normalized by its corresponding euclidean distance and is given by

$$\tilde{d} = \frac{2}{N(N-1)} \sum_{(i,j)} \frac{|G_M(i,j) - G_E(i,j)|}{|G_E(i,j)|} = E[d]. \quad (11)$$

Both the definitions of \tilde{d} and \tilde{c} involve an averaging using the mean operator. However, the simple mean cannot be a robust measure for highly noisy data which may also contain outliers lying far away from the intrinsic manifold structure. To increase the robustness of the measures, we propose using α -trimmed mean. This chops off the top $(\alpha/2)\%$ and lowest $(\alpha/2)\%$ of all the entries c_k and d_{ij} in (10) and (11) from being considered. The general form for α -trimmed mean for any vector $Y \in \mathbb{R}^{n \times 1}$ is given as

$$E^\alpha[Y] = \frac{1}{n - 2\alpha n} \sum_{i=\alpha n+1}^{n-\alpha n} Y_{(i)} \quad (12)$$

where $Y_1 \leq Y_2 \leq Y_3 \dots \leq Y_n$ are entries of the vector Y sorted in increasing order. Similarly, \tilde{d}^α and \tilde{c}^α are the robust α -trimmed versions of the measures introduced earlier in this section.

Together with robustness, the measures should also remain sensitive to the slightest of disentanglement. For this, a close neighborhood for each point is ignored as the measures remain indifferent for that region in both a flat and a non-flat manifold. For each point x_i , we take \mathcal{S} number of neighbors in which the $G_M(i, j)$ and $G_E(i, j)$, for a certain point x_j in that \mathcal{S} -neighborhood of x_i , are going to be almost similar to each other and will thus have a high correlation too. In order to make the measures sensitive, distances for these nearby pairs should be ignored from the computation. The decision that two points x_j and x_i are not lying in the considered neighborhood range is denoted by $\mathcal{S}(i, j)$ being not equal to 0 and thus the measures can now be modulated as

$$\tilde{d}^\mathcal{S} = E_{\mathcal{S}(i,j) \neq 0}[d] \quad (13)$$

and

$$\tilde{c}^\mathcal{S} = E_{\mathcal{S}(i,j) \neq 0}[c]. \quad (14)$$

Both sensitivity and robustness are considered while computing our measures $\tilde{d}^{\mathcal{S}, \alpha}$ and $\tilde{c}^{\mathcal{S}, \alpha}$, which are henceforth referred to as simply d and c respectively. Through experiments on synthetic and real world image datasets, it has been observed that the measures do not change much with the tuning of parameters α and \mathcal{S} . However, if these are taken to be very high, we end up losing a lot of information about the manifold structure of the data and its variation across the layers of a deep learning network. Similarly, closer to zero values of α and \mathcal{S} may make the measures of entanglement susceptible to noise and outliers. Smaller values of d occur when the pair-wise geodesic and euclidean distances have values closer to each other whereas smaller values of c would mean that the pair-wise geodesic and euclidean distances are more correlated to each other, i.e. two points farther away from each other with respect to the l_2 norm are also farther away in a geodesic sense. Reduction of both the measures will suggest that the manifold

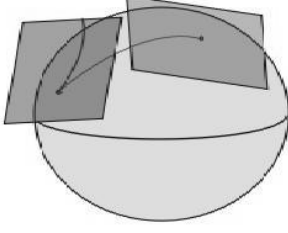


Fig. 3. Alignment of tangent spaces along a geodesic path.

is more unfolded or flattened. For an ideal flat manifold, the values of both c and d should be equal to 0.

The advantages of choosing such forms for the measures are that these are now both translation and rotation invariant since the distances between samples are not affected as long as the geometry is maintained. Also, any isotropic scaling of the data matrix by a multiplicative scalar will cause the numerator and denominator in both (10) and (11) to change by the same order and hence canceling out the effect on the resultant values of c and d . Thus, any change in these measures has to be due to a geometrical modification of the structure of the data manifold.

As can be seen in Fig. 3, one can also use the alignment of the tangent space estimated at different locations on a geodesic path as another quantifier for the flatness of a manifold [22]. An estimate of the tangent subspace at a particular data point x_i on the sampled manifold can be obtained using its k -nearest neighborhood $N(x_i)$ by forming the $P \times k$ matrix

$$\mathbf{X}_i = [x_1^i x_2^i \dots x_k^i] \quad (15)$$

where $x_j^i \in N(x_i)$. The mean centering of each of the matrices is performed as

$$\hat{\mathbf{X}}_i = \mathbf{X}_i - \frac{1}{k} \mathbf{X}_i \mathbf{1}. \quad (16)$$

The neighborhood covariance matrix for each x_i is then given by

$$\mathbf{W}_i = \hat{\mathbf{X}}_i^T \hat{\mathbf{X}}_i. \quad (17)$$

A local PCA using the covariance matrix gives rise to

$$\mathbf{W}_i = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^T \quad (18)$$

where $\mathbf{\Lambda}$ has all the eigenvalues of \mathbf{W}_i in its diagonal. If the intrinsic dimensionality of the manifold is known to be p , then it is known through [22] that the eigen-vectors from \mathbf{Q} corresponding to the p largest eigen-values span the tangent space \mathcal{T}_i of the manifold at x_i . All the tangent spaces of a connected manifold have the same intrinsic dimension which is equal to the intrinsic dimension of the manifold p . In case of a flat manifold, the tangent spaces at all locations are perfectly aligned with each other. Principal or canonical angle [23] between subspaces is used to quantify the alignment. To find the angle between two tangent spaces \mathcal{T}_i and \mathcal{T}_j , the inner

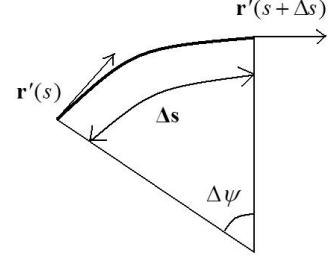


Fig. 4. Calculating curvature from changing tangent space along the surface of a manifold.

product matrix $\mathbf{Z} = \mathcal{T}_i^T \mathcal{T}_j$ is taken and a Singular Value Decomposition (SVD) is performed on \mathbf{Z}

$$\mathbf{Z} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \quad (19)$$

to obtain the singular values $\mathbf{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \dots)$. The principal angle between tangent subspaces \mathcal{T}_i and \mathcal{T}_j is given by

$$\theta_{i,j} = \min(\cos^{-1}(\sigma_1), \cos^{-1}(\sigma_2), \dots). \quad (20)$$

From a differential geometry perspective [24], let $\mathbf{r}(s)$ be an arc length parametrized curve on the manifold M as shown in Fig. 4. The tangent unit vector is denoted by $\mathbf{r}'(s)$. The second derivative information contained in $\mathbf{r}''(s)$ is obtained by

$$\mathbf{r}''(s) = \lim_{\Delta s \rightarrow 0} \frac{\mathbf{r}'(s + \Delta s) - \mathbf{r}'(s)}{\Delta s}. \quad (21)$$

As can be seen in the Fig. 4, $\mathbf{r}'(s)$, $\mathbf{r}'(s + \Delta s)$ and $\mathbf{r}'(s + \Delta s) - \mathbf{r}'(s)$ form an isosceles triangle with the vertex angle $\Delta\psi$. Since $\mathbf{r}'(s)$ is a unit vector and assuming Δs to be very small,

$$|\mathbf{r}'(s + \Delta s) - \mathbf{r}'(s)| = \Delta\psi. \quad (22)$$

The absolute value of the second derivative in (21) becomes

$$|\mathbf{r}''(s)| = \lim_{\Delta s \rightarrow 0} \frac{\Delta\psi}{\Delta s} = \kappa \quad (23)$$

where κ is popularly called as the curvature. It is an indication of the rate of change of tangent spaces along a geodesic curve on the manifold.

To analyze the trend of the variation of principal angles between tangent spaces, a geodesic curve (shortest path) on the manifold originating from source i_1 to destination i_n and passing through n data points is found out. The principal angles are computed between \mathcal{T}_1 and \mathcal{T}_i , where $i \in 2, 3, \dots, n$ and stored as $\theta_{1,i}$. Similarly, $g_{1,i}$ denotes the estimate of the geodesic distance between x_1 and x_i . From (23), the variation of $\theta_{1,i}$ with respect to $g_{1,i}$ measures the curviness or the rate of change of the tangent space as the manifold is traversed along the geodesic curve. Thus, the metric used is the absolute value of the slope which is estimated, in discrete sense, as



Fig. 5. Two swiss rolls entangled within each other.

$$\Phi = \frac{\partial \theta}{\partial g} \approx E \left[\frac{\theta_{1,i} - \theta_{1,i-1}}{g_{1,i} - g_{1,i-1}} \right]. \quad (24)$$

Ideally for a flat manifold, Φ equals to 0 because all the tangent spaces are perfectly aligned with each other. The larger the value, the more curved the manifold is.

After having come up with measures to quantify the unfolding of individual manifolds, it is worth noting here that class separability between them is also important to guarantee disentanglement in a classification paradigm. Conventional classification accuracy or any kind of margin based quantifiers can be used to measure this aspect of disentanglement. The normalized margin [25] for an example x_n is given by

$$m_n = \frac{\|x_n - \mathcal{M}(x_n)\| - \|x_n - \mathcal{H}(x_n)\|}{\|x_n - \mathcal{M}(x_n)\|} \quad (25)$$

where $\mathcal{M}(x_n)$ is the nearest miss (nearest neighbor from a different class) and $\mathcal{H}(x_n)$ is the nearest hit (nearest neighbor from the same class) of x_n . Maximization of the margin means more separability of individual class clusters. Now, we apply the measures discussed above on a deep belief network acting on synthetic and real world image datasets.

V. EXPERIMENTS

A. Experiments with synthetic data

Two Swiss roll type manifolds are entangled with each other as shown in Fig. 5 to form the dataset. It is obvious that no linear hyperplane can separate these two classes in the original 3-dimensional space. As discussed above, if representational features can be found out depicting the underlying dynamics that has led to the construction of this dataset, then it may be easier to classify the two. The structure of the underlying low-dimensional manifold is known [21] to be flat or linear for the Swiss rolls since they are intrinsically curvature-less functions which can easily be represented in a flat Euclidean \mathbb{R}^2 subspace. In total, $N = 4000$ points are sampled from the two Swiss rolls and the two entanglement quantifying measures d and c are now applied on the output features of each of the three layers in a 3-30-30 deep belief network

TABLE I
DEGREES OF ENTANGLEMENT d AND c MEASURED ON THE TWO MANIFOLDS PROJECTED AT DIFFERENT LAYERS OF A 3-30-30 DBN.

Layers	Manifold-1		Manifold-2	
	d	c	d	c
L0 (3)	0.6174	0.7853	0.6390	0.7947
L1 (30)	0.3971	0.4099	0.4294	0.5084
L2 (30)	0.3250	0.2661	0.3622	0.3399

TABLE II
CLASSIFICATION ACCURACY USING FEATURES FROM DIFFERENT LAYERS

Layers (Dimensions)	LDA	QDA
L0 (3)	60.8	60.2
L1 (30)	73.8	98.9
L2 (30)	76.7	97.2
L0+L1 (33)	76.9	99.6
L0+L1+L2 (63)	98.8	100.0

(DBN) formed by stacking two greedily trained continuous RBMs. By experimenting with different choices for width of the hidden layers, it was seen that best results were obtained by increasing the number of neurons in the higher layers beyond 30. This might be because of the low dimensionality of the given Swiss-roll data. Since the actual manifold structure is known, the shortest path distances using a 25 nearest neighbor graph give a good estimate of the actual geodesic distances. For sensitivity and robustness considerations as discussed in the previous section, the values of α and \mathcal{S} are taken to be 10 and 20 respectively. The input layer is denoted as L0 and the subsequent layers are named L1 and L2. As can be seen in Table I, both d and c decrease monotonically as we climb up the layers giving an indication that deeper layers tend to unfold the structure of the individual class manifolds.

Remark 1. Higher layers in a deep neural network flatten the individual class manifolds.

As was mentioned earlier, disentanglement need not refer to just flattening or linearization. Thus, the separability of the class manifolds was also studied using the features generated at different layers. It was seen in Table II that when the features of all three layers are concatenated and subject to even a simple Linear discriminant Analysis (LDA), which is a linear classifier, 98.8% classification accuracy could be obtained. Higher layers generate features which have more separation between the classes and a quadratic classifier like Quadratic Discriminant Analysis (QDA) can produce perfect classification with all features combined. The features using the stacked continuous RBMs were extracted in a fully unsupervised manner. This also gives us an insight into how these learning networks are helping in proper disentanglement of the contained manifolds in the data.

Remark 2. With a deep learning network, the individual class manifolds can not only be unfolded but also their classifiability can be enhanced by using the features extracted from higher layers.

TABLE III

DEGREES OF ENTANGLEMENT d AND c MEASURED AT DIFFERENT LAYERS OF A 100-1000-1000 DBN APPLIED ONTO THE DIMENSIONS-EXTENDED VERSIONS OF THE MANIFOLDS IN FIG. 5.

Layers	d	c	Layers	d	c
L0 (100)	0.5966	0.7657	L0 (100)	0.6225	0.7854
L1 (1000)	0.5971	0.7688	L1 (1000)	0.6236	0.7891
L2 (1000)	0.5961	0.7772	L2 (1000)	0.5936	0.6708

Another experiment was conducted to increase the dimensionality P of the Swiss roll data by adding redundant features to it. Raw real world data, usually noisy and high-dimensional ($P \gg N$), often comes with features which are absolutely irrelevant to the learning task. We wanted to check whether the deep learner can perform disentanglement as well when the data contains lots of irrelevant dimensions. Let the original Swiss-roll data be \mathbf{X} that belongs to $\mathbb{R}^{N \times 3}$. Redundant dimensions are added in the form of an initially all-zero matrix \mathbf{Z} of size $\mathbb{R}^{N \times (D-3)}$ to form

$$\mathbf{X}_R = [\mathbf{X} \ \mathbf{Z}] \quad (26)$$

and thus, $\mathbf{X}_R \in \mathbb{R}^{N \times D}$. An orthonormal matrix $\mathbf{L} \in \mathbb{R}^{D \times D}$ is multiplied with \mathbf{X}_R and summed up with additive noise $\mathcal{N} \in \mathbb{R}^{N \times D}$ to get the rotated manifold projected in a higher dimensional space

$$\mathbf{Y} = \mathbf{X}_R \mathbf{L} + \mathcal{N}. \quad (27)$$

For our experiments, N is kept the same 4000 as before and D was taken to be 100. Also, the noise \mathcal{N} was considered to be zero mean Gaussian with a small variance that is about 1% of that of the original data \mathbf{X} . This newly generated dataset \mathbf{Y} , that belongs to $\mathbb{R}^{4000 \times 100}$, is now fed to a continuous RBM based neural network of size 100-1000-1000. As can be seen with the entanglement measures listed in Table III, no significant flattening is obtained since the values of d and c do not vary along the layers for both the class manifolds. This experiment was also done in the absence of any added noise and the trend of the measures across the layers was again similar. It means that too many nuisance dimensions in the data confuses the deep learning model.

Remark 3. *Disentanglement is severely degraded in presence of a large number of redundant dimensions in input data.*

Subsequently, the curvature based metric Φ was calculated using the features at different layers. Since the Swiss-roll is known to be a 2-manifold, the intrinsic dimensionality p was taken to be 2. To estimate the tangent spaces, the value of k in (15) was taken to be 5. For each of the two manifolds, source (S) and destination (D) data examples were chosen that were geodesically distant enough so that a curve connecting them can traverse most of the manifold as can be seen in Fig. 6. Alignment of the tangent spaces along the geodesic path with respect to the tangent space at source was analyzed and the metric Φ was calculated as per (24). Since the DBN creates a one-to-one representational mapping, the same data nodes were also used as S and D while computing the shortest path and the curvature Φ at different layers of the autoencoder. A

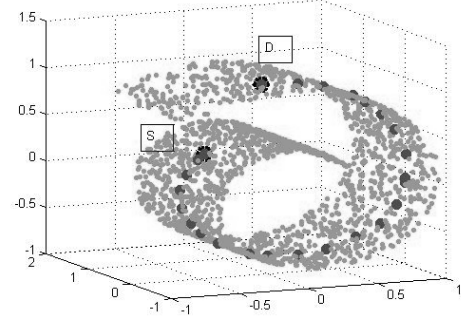


Fig. 6. A geodesic path (in darker dots) between S and D.

TABLE IV

CURVATURE BASED DEGREE OF ENTANGLEMENT Φ AT DIFFERENT LAYERS OF 3-30-30 DBN

Layers	Φ for Manifold-1	Layers	Φ for Manifold-2
L0 (3)	87.3768	L0 (3)	88.3762
L1 (30)	65.4784	L1 (30)	77.1275
L2 (30)	42.7480	L2 (30)	39.7518

TABLE V

DEGREES OF ENTANGLEMENT d AND c MEASURED ON THE TWO CLASSES IN FIG. 7 AT DIFFERENT LAYERS OF A 3-30-30 DBN

LAYERS	Swiss-roll Manifold		\$-shaped Manifold	
	d	c	d	c
L0 (3)	0.5356	0.6708	0.2148	0.1548
L1 (30)	0.4975	0.6178	0.2196	0.1550
L2 (30)	0.3358	0.3110	0.2114	0.1272

clear decrease in Φ as we go up the layers reaffirms Remark 1 stating manifold unfolding in deeper layers.

It is important to note here that the measures of entanglement discussed above have severe limitations and may specially fail to denote unfolding when each class, in itself, consists of a mixture of manifolds or self-intersecting ones. A dollar shaped (\$) manifold, which can be considered to be either self-intersecting or a combination of an S-shaped and a linear (vertically aligned) manifold, belongs to one of the classes and the other class is represented by a Swiss roll in Fig. 7. If the unfolding measures c and d are measured across the layers of a stacked RBM based 3-30-30 network for the dollar shaped manifold, variations in the structure of the manifold cannot be observed in Table V. However, a single connected component like the Swiss-roll does exhibit unfolding and reduction of entanglement measures as has been shown previously too.

Therefore, it is important to enlist the explicit set of assumptions that are made on the data while applying these measures of entanglement. These are as follows:

- 1) In its original space, the data belonging to each class lie

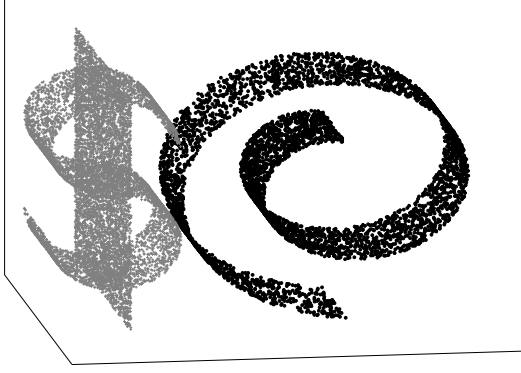


Fig. 7. A dataset of two classes where the one in the shape of a dollar (\$) contains a mixture of manifolds in itself.

on a single connected manifold. The measures will fail to quantify disentanglement if each class, in itself, is a mixture of multiple disconnected manifolds.

- 2) Each of the class manifolds is considered to be smoothly varying and without self-intersections.
- 3) A completely unfolded manifold is considered as a linear or Euclidean subspace in the projected space.
- 4) The training data represents a densely sampled version of the original manifold such that all the relevant variations are captured by choosing the neighborhood size to form a nearest neighbor graph.

B. Experiments with face images

The Olivetti Face dataset is a popularly used database of human faces that contains ten gray-scale images each of forty different people, making it 400 images in total. We augmented this dataset following the lines of [7] by applying rotations (-25 to +25 degrees), translation and re-sampling. The final cropped input to the DBN contains 20400 images of 25×25 pixels each. In the experiments, each face is treated as a separate manifold and the pixel intensity values represent a feature vector for each image. All the images were mean centered and normalized to make average standard deviation to be 1. A 3-layered deep belief network 625-2000-625 was designed by stacking two RBMs on top of one another and greedily training them to form an autoencoder. Another output softmax layer containing 40 nodes may also be added in order to perform the task of classification. However, the current experiment aims at analyzing the change in structure of the data using the hidden layers. The first RBM is a Gaussian-Binary RBM since the gray scale images of faces can be better modeled as Gaussian real values rather than binary while the latent or hidden variables were taken to be binary units. The energy function of such a RBM turns out to be

$$E(x, h) = \sum_{vis} \frac{(x_i - a_i)^2}{2\sigma^2} - c'h - \sum_{i,j} W_{ij} \frac{x_i h_j}{\sigma_i \sigma_j}. \quad (28)$$

Also, the output code layer was modeled as linear units. If the deep learner can actually disentangle the individual face

TABLE VI
COMPARING d AND c FOR SOME OF THE FACE MANIFOLDS BETWEEN THE INPUT AND PROJECTED CODE LAYERS OF A 625-2000-625 DBN





Face Manifolds	c		d	
	Input	Projected	Input	Projected
	0.1954	0.1670	0.9091	0.6722
	0.1305	0.0951	1.0452	0.6606
	0.1630	0.1355	0.9943	0.6581
	0.2242	0.2103	1.0304	0.6990

TABLE VII
COMPARISON OF MARGIN VALUES USING FEATURES FROM DIFFERENT LAYERS OF THE DBN.

Layers	Margin
L0 (625)	0.7019
L1 (2000)	0.8084
L2 (625)	0.8579

manifolds, then a reduction of the flattening metrics c and d and an increase in separability margin will be observed in the projected code layer as compared to the input layer. For fairness of comparison, dimensionality of input and code layers were kept same, though the measures are independent of the dimension of the data space. Of the forty subjects, four faces manifolds were chosen at random and the values of c and d measured at the input and the projected code layer were enlisted in Table VI. A decrease in the measures signifies the occurrence of flattening caused due to unsupervised deep training with respect to Remark 1. On an average over all the faces, c and d recorded a decrease of 18.92% and 33.54% respectively as compared to their values on the raw input.

Next, the average normalized margin was observed across the layers in Table VII and an increase in the values is indicative of the increasing separability of class manifolds with depth.

Though it is hypothesized that a deep model can extract the governing (hidden) factors of variation in the data, lack of ground truth for most real world data hinders its provability. Hence, we considered experimenting on an image dataset, with known ground-truth factors of variation, for which manifold learning methods have been previously implemented successfully to do dimensionality reduction. The ISOMAP face dataset [21] consists of 698 images of a fixed face rendered under different pose (horizontal and vertical alignment) and lighting (azimuthal angle of the lighting source) conditions. Although each image (64X64 pixels) lie on a 4096-dimensional input space, the intrinsic dimensionality of the manifold defining all the images is rather just three, given by the two pose and one lighting angle variables. Given the complexity of the manifold embedded in such a high-

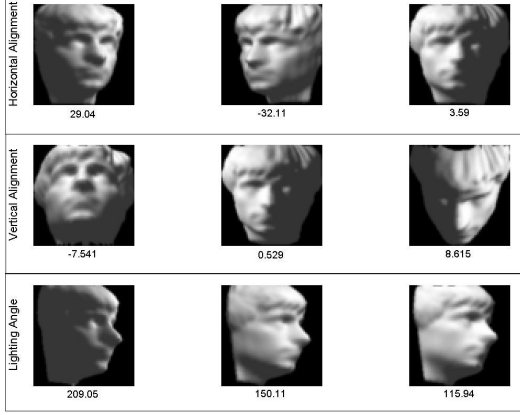


Fig. 8. Images from the ISOMAP face dataset with respect to the pose variables in the first two rows and lighting angle in the last row.

TABLE VIII

CROSS-CORRELATION OF 3-DIMENSIONAL EMBEDDINGS OBTAINED USING CLASSICAL PCA, ISOMAP AND DBN+PCA WITH GROUND-TRUTH FOR HORIZONTAL POSE, VERTICAL POSE AND LIGHTING DIRECTION

Hidden Factors	PCA	ISOMAP	DBN+PCA
Horizontal Pose	0.75	0.98	0.9143
Vertical Pose	0.33	0.901	0.673
Lighting Angle	0.68	0.927	0.8629

dimensional space, it is intuitive to note that a linear dimensionality reduction method like PCA would fail to recover the values of the causing factors. ISOMAP, on the other hand, is a nonlinear manifold learning algorithm which first forms a nearest neighbor graph, computes shortest path distances between each pair of images and finally uses these distances to perform a isometric low dimensional embedding using eigen-decomposition. In Table VIII, 3-dimensional embeddings of the face image dataset were obtained using both PCA and ISOMAP and the best match cross-correlation with the ground truth values of the two pose and lighting angle variables were reported. As expected, PCA resulted in confused embeddings showing very low correlation with the true parameters while very high correlation was observed with the embeddings obtained through ISOMAP.

The reason behind the failure of PCA is non-linearity of the data in its original dimensional space. Had the data been on a linear or flat manifold (like a plane in \mathbb{R}^3), dimensionality reduction could have easily been obtained through linear methods. The unfolding of manifolds using deep architectures has been shown so far on both synthetic Swiss-roll type as well as real world face datasets. This property of deep learning can well be leveraged to flatten the ISOMAP face manifold first and then apply linearity assumption based PCA to recover the hidden parameters controlling the data generation. All images were down-sampled by 2 and fed to a 1024-2000-1000-300-50 autoencoder formed by stacking RBMs on top of one another just like in the previous case. A 3-dimensional PCA reduction is applied onto the final code layer output and the prominent

TABLE IX
c FOR DIGIT MANIFOLDS AT DIFFERENT LAYERS AFTER UNSUPERVISED PRE-TRAINING.

DIGIT	L0(784)	L1(1000)	L2(1000)	L3(1000)
0	0.2625	0.2230	0.2428	0.2084
2	0.4123	0.3707	0.3734	0.3518
4	0.4126	0.3564	0.3830	0.3293
7	0.2994	0.2542	0.2712	0.2445
9	0.3209	0.2496	0.2864	0.2493

TABLE X
d FOR DIGIT MANIFOLDS AT DIFFERENT LAYERS AFTER UNSUPERVISED PRE-TRAINING.

DIGIT	L0(784)	L1(1000)	L2(1000)	L3(1000)
0	2.1063	2.0355	2.0158	1.9992
2	2.3745	2.3960	2.3418	2.3672
4	2.3732	2.3359	2.3208	2.2805
7	2.1508	2.0696	2.0669	2.0072
9	2.2755	2.1841	2.1980	2.1345

eigen-vectors are compared with the ground truth pose and lighting values for each image. Again, cross-correlation with the factors of variation is taken as a metric to evaluate the model's performance.

The deep unsupervised learning plus linear dimensionality reduction method is undoubtedly better than just PCA but still lags behind the performance of ISOMAP. The possible reason is that the objective function of ISOMAP is explicitly designed to cater to manifold embedding needs. However, the results with a stacked RBM architecture can rather be considered as a by-product of the learning algorithm. Involving manifold prior based regularization in the optimization function of the autoencoder may improve the deep learner's performance significantly.

Remark 4. *With a deep generative network, the hidden factors governing the variations in the data can be recovered which are also representative of the true intrinsic structure of the manifold.*

C. Experiments with images of MNIST handwritten digits

The MNIST database of handwritten digits [26] has a training set of 60,000 images and a test set of 10,000 images of the ten numeric digits written by various people. Each digit has been normalized and centered in a 28×28 image. The thickness, height, angular alignment, relative position in frame are some of the intrinsic hidden properties that govern the generation of the examples for each digit manifold. The entire dataset of images is considered as a collection of blended manifolds plus additive noise. A multi-layer 784-1000-1000-1000 DBN was set up and pre-trained layer by layer as binary-binary RBMs without any label supervision. The disentanglement measures c and d , as reported in Table IX and X for five digits using a 6 nearest neighbor graph, show a decreasing trend, though not monotonic, as the deeper layers are traversed. The parameters for robustness (α) and sensitivity



Fig. 9. Digit 7 (a) without a middle stick and (b) with a middle stick.

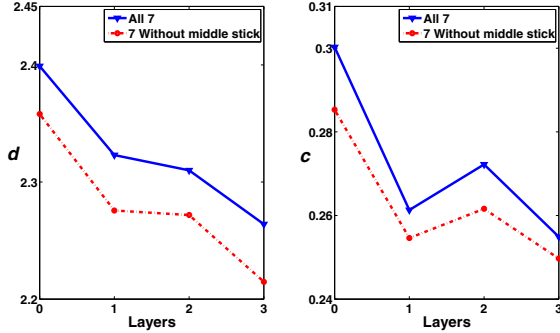


Fig. 10. d and c for whole class and a subclass of digit 7.

(\mathcal{S}) were also taken to be 10 and 20 respectively. Φ was not calculated on this dataset since it requires the knowledge of the true intrinsic dimensionality. However, experiments can also be conducted using a guess value of the parameter.

If each class consists of multiple disconnected manifolds, then the distances considered between some pairs of images which do not belong to the same manifold may lead to discrepancies in the estimates of the measures as it violates the assumptions mentioned earlier. Unlike the Swiss-roll example in previous subsection, describing all images belonging to a unique digit as a single manifold may be debatable. Thus, we tried to find subclasses among the digits characterized with known physical appearance features and then evaluate c and d over them. For example in Fig. 9, the digit 7 can be broadly written in two different styles- either (a) without a middle stick, or (b) with it. When only those examples of the digit 7 falling under category (a) were considered, both c and d were found to be lower than those for the combined class and also experienced the same trend that deeper layers have more unfolding, as can be seen in Fig. 10.

It is also shown in Table XI and XII that after supervised backpropagation (using another softmax output layer of 10 nodes for each of the 10 digits), the model not only retains the disentanglement but also improves upon it while increasing the discriminative abilities between the class manifolds too. Though the latter is quite expected for a discriminative procedure like supervised backpropagation, the observation that unfolding of manifolds also improves with it is rather surprising and interesting. The final test classification error was 1.12% that matches with the results in [7]. Also, it was shown in [10] that concatenation of raw input with higher representations were more linearly discriminative. Thus, it also explains that the deep network produces a set of unfolded manifolds which are also more separable than the original

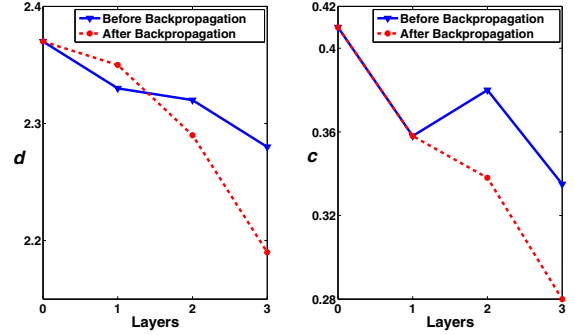


Fig. 11. d and c across the layers for the manifold of digit '4' before and after supervised backpropagation.

TABLE XI
 c FOR DIGIT MANIFOLDS AT DIFFERENT LAYERS AFTER SUPERVISED FINE-TUNING USING BACKPROPAGATION.

DIGIT	L0(784)	L1(1000)	L2(1000)	L3(1000)
0	0.2625	0.2042	0.1791	0.1612
2	0.4123	0.3535	0.3144	0.2747
4	0.4126	0.3576	0.3333	0.2912
7	0.2994	0.2551	0.2361	0.2069
9	0.3209	0.2788	0.2551	0.2266

TABLE XII
 d FOR DIGIT MANIFOLDS AT DIFFERENT LAYERS AFTER SUPERVISED FINE-TUNING USING BACKPROPAGATION.

DIGIT	L0(784)	L1(1000)	L2(1000)	L3(1000)
0	2.1063	2.0577	2.0243	1.9629
2	2.3745	2.4215	2.3604	2.3165
4	2.3732	2.3482	2.2997	2.1851
7	2.1508	2.0912	2.0846	1.9878
9	2.2755	2.2395	2.2199	2.1121

input manifolds. A basic problem with conventional multi-layer neural networks was that the lower layers remained under-trained due to the problem of vanishing gradient. Thus, the main contribution of the generative pre-training using RBMs can be thought of better training of the lower layers for which we see a clear drop in the entanglement measures (lower in L1 as compared to L0). It is intuitive that the unsupervised stage of deep belief network, which is based on sampling based distribution learning technique, loses a lot of relevant information while moving up the layers. Thus, we see a zig-zag structure in the plots of Fig. 10 instead of an ideal monotonic decrease. A supervised and discriminative backpropagation pronounces the disentanglement process for the higher layers for which we see a more expected monotonic decrease of entanglement measures as we go up the layers.

Remark 5. *Unsupervised pre-training (generative) helps in disentangling the class manifolds in the lower layers of a deep learner while the upper layers are better disentangled when it is subject to a supervised backpropagation (discriminative). A combination of the two helps boosts the overall performance.*

TABLE XIII
COMPARISON OF MARGIN VALUES USING FEATURES FROM DIFFERENT
LAYERS OF A 784-1000-1000-1000 DEEP NETWORK.

Layers (Dimensions)	After Pre-training	After backpropagation
L0 (784)	0.2754	0.2754
L1 (1000)	0.3189	0.3303
L2 (1000)	0.3201	0.3498
L3 (1000)	0.3503	0.4158

Now that the evidence for unfolding of individual class manifolds is established, we look at the separability aspect of disentanglement by comparing the expected value of the margin estimator taken over all observations. It was seen previously that deep layers tend to generate hidden features of importance in all layers and these features provide the intrinsic identity to each class making them more separable. As can be seen clearly in Table XIII, there is a rise in the margin values when the deeper layers are considered as the feature space; thus indicating what was pointed out in Remark 2. In [10], classification accuracy using a linear Support Vector Machine classifier was shown to be competitive enough while using all layer-wise features combined. It is interesting to see here that the maximum impact by the discriminative backpropagation is caused in the higher layers like L3 (which is penultimate to the target layer of ten label nodes). This suggests that a combination of generative modules at the first followed by discriminative modules can actually perform better in the task of classification or clustering. This is also the reason why an unsupervised pre-training helped gain higher classification accuracy as compared to a traditional multi-layer neural network that lacked a pre-trained generative module.

Remark 6. *The unfolding of class manifolds through deep architectures also creates an expansion of the feature space in which the classes are more separable from each other.*

VI. CONCLUSION

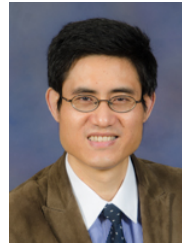
The extraction of the governing factors of variation using deep neural networks has been discussed in the literature before. In this paper, manifold based measures were proposed and tested to justify the hypothesis of unfolding and subsequent separation of intrinsically low-dimensional manifolds in the data. Experiments on synthetic and real world datasets revealed the gradual disentanglement of individual class manifolds at deeper layers following unsupervised pre-training and that got even better when the network was subject to a discriminative backpropagation based fine-tuning. This suggests that the fundamental function of the first-generate-then-discriminate models is to first-unfold-then-separate each individual manifold present in the data. This insight explains why an unsupervised pre-training has helped overcome the deficiencies of traditional neural networks. The change in the entanglement values in case of real world image datasets was rather miniscule. It may be due to Remark 3 which suggests that the presence of too many redundant features in each layer may hinder the learning process. Moreover, the

remarks discussed in this paper also explain the performance gain achieved by other deep learning methods, even those which are not entirely based on neural network framework. For example, in [17], a sparse random projection based dimension reducer is first employed before feeding the projected data into a supervised neural network. It has been researched before that by virtue of the Johnson-Lindenstrauss lemma [27], under certain conditions, random projections can lead to low-dimensional embedding of points such that distances between points are nearly preserved. This can also lead to manifold disentanglement which can subsequently aid any discriminative learner like the neural networks. One immediate application of our work can be better interpolation between examples, like face warping, using features from higher layers. Enforcing disentanglement as a regularization constraint in the optimization objective can also lead to even superior performance. Algorithms like the Contractive Auto-encoder [28] and semi-supervised deep learning [29] have indeed applied a manifold based penalty in their objective functions and have led to better results with real world datasets. It was also shown in this paper that unfolding of manifolds is limited to the lower layers of an unsupervised deep network while the separability is prominent mostly in the higher layers during supervised training. This fact can be used to isolate both the generative and discriminative processes by limiting them only to the lower and higher layers respectively which may ultimately lead to faster learning.

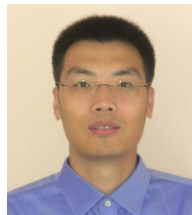
REFERENCES

- [1] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the 19th International conference on Artificial Intelligence and Statistics*, Sardinia, Italy, 2010, pp. 249–256.
- [2] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in *Advances in Neural Information Processing Systems*, Montreal, Canada, 2014, pp. 2933–2941.
- [3] B. Scholkopf and A.J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA: MIT Press, 2002.
- [4] L. Deng and D. Yu, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 34, pp. 197–387, 2014.
- [5] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proceedings of the 10th European Conference on Machine Learning*, Chemnitz, Germany, Springer-Verlag 1998, pp. 137–142.
- [6] Y. Bengio, O. Delalleau, and N.L. Roux, "The curse of highly variable functions for local kernel machines," in *Advances in Neural Information Processing Systems*, Vancouver, Canada, 2005, pp. 107–114.
- [7] G.E. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [8] W. Li, M. Zeiler, S. Zhang, Y.L. Cun, and R. Fergus, "Regularization of neural networks using dropconnect," in *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, GA, USA, Jun. 2013, pp. 1058–1066.
- [9] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
- [10] Y. Bengio, G. Mesnil, Y. Dauphin, and S. Rifai, "Better Mixing via Deep Representations," in *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, GA, USA, Jun. 2013, pp. 552–560.
- [11] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.35, no.8, pp. 1798–1828, Aug. 2013.

- [12] Y. She, "Selectable factor extraction in high dimensions," arXiv: 1403.6212. [Online]. Available: <http://arxiv.org/abs/1403.6212>
- [13] Q. Huang and D. Wu, "Flatten a Curved Space by Kernel," *IEEE Signal Processing Magazine*, vol. 30, no. 5, pp. 132–142, Sep. 2013.
- [14] I. J. Goodfellow, Q. V. Le, A. M. Saxe, H. Lee, and A. Y. Ng, "Measuring invariances in deep networks," in *Advances in Neural Information Processing Systems*, volume 22, pp. 646–654, 2009.
- [15] M.A. Carreira-Perpinan and G.E. Hinton, "On Contrastive Divergence Learning," in *Proceedings of the 10th International workshop on Artificial Intelligence and Statistics*, Barbados, 2005, pp. 33–40.
- [16] H. Chen and A. F. Murray, "Continuous restricted Boltzmann machine with an implementable training algorithm," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 150, no. 3, pp. 153–158, Jun. 2003.
- [17] G.E. Dahl, J.W. Stokes, L. Deng, D. Yu, "Large-scale malware classification using random projections and neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013, pp. 3422–3426.
- [18] J. M. Lee, *Introduction to Smooth Manifolds*. New York: Springer-Verlag, 2002.
- [19] L. Cayton, "Algorithms for manifold learning," University of California at San Diego, CA, Technical Report CS2008-0923, 2005.
- [20] R. Basri and D.W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218–233, Feb. 2003.
- [21] J.B. Tenenbaum, V. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [22] Z. Zhang and H. Zha, "Principal Manifolds and Nonlinear Dimensionality Reduction via Tangent Space Alignment," *SIAM Journal of Scientific Computing*, vol. 26, no. 1, pp. 313–338, 2005.
- [23] A.V. Knyazev and M.E. Argentati, "Principal angles between subspaces in an A-based scalar product: algorithms and perturbation estimates," *SIAM Journal on Scientific Computing*, vol. 23, no.6, pp. 2008–2040, 2002.
- [24] E. Kreyszig, *Differential Geometry*. New York: Dover, 1991.
- [25] Y. Sun and D. Wu, "Feature Extraction through Local Learning," *Statistical Analysis and Data Mining*, vol. 2, no. 1, pp. 34–47, 2009.
- [26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [27] D. Achlioptas, "Database-friendly random projections: Johnson-Lindenstrauss with binary coins," *Journal of Computer and System Sciences*, vol. 66, no. 4, pp. 671–687, 2003.
- [28] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011, pp. 833–840.
- [29] J. Weston, F. Ratle, H. Mobahi, and R. Collobert, "Deep learning via semi-supervised embedding," in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008, pp. 1168–1175.



Dapeng Wu (S'98–M'04–SM'06–F'13) received Ph.D. in Electrical and Computer Engineering from Carnegie Mellon University, Pittsburgh, PA, in 2003. He is a professor at the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL. His research interests are in the areas of networking, communications, signal processing, computer vision, machine learning, smart grid, and information and network security. He received University of Florida Research Foundation Professorship Award in 2009, AFOSR Young Investigator Program (YIP) Award in 2009, ONR Young Investigator Program (YIP) Award in 2008, NSF CAREER award in 2007, the IEEE Circuits and Systems for Video Technology (CSVT) Transactions Best Paper Award for Year 2001, and the Best Paper Awards in IEEE GLOBECOM 2011 and International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine) 2006. Currently, he serves as an Associate Editor for IEEE Transactions on Circuits and Systems for Video Technology and IEEE Transactions on Signal and Information Processing over Networks. He is an IEEE Fellow.



Yiyuan She received B.S. in Mathematics and M.S. in Computer Science from Peking University in 2000 and 2003, respectively, and received Ph.D. in Statistics from Stanford University in 2008. He is currently an Associate Professor in the Department of Statistics at Florida State University. His research interests include high-dimensional statistics, machine learning, multivariate statistics, robust statistics, statistics computing, and network science.



Pratik Prabhanjan Brahma received B.Tech. in Electronics and Electrical Communication Engineering and M.Tech in Telecommunication Systems Engineering from the Indian Institute of Technology, Kharagpur, India, in May 2011. He is currently a Ph.D. candidate in Electrical and Computer Engineering at University of Florida, Gainesville, FL. His research interests are in the areas of machine learning, computer vision, data science, statistics and network science.