Nonlinear Learning using Local Coordinate Coding

Kai Yu
NEC Laboratories America
kyu@sv.nec-labs.com

Tong Zhang
Rutgers University
tzhang@stat.rutgers.edu

Yihong Gong
NEC Laboratories America
ygong@sv.nec-labs.com

Abstract

This paper introduces a new method for semi-supervised learning on high dimensional nonlinear manifolds, which includes a phase of unsupervised basis learning and a phase of supervised function learning. The learned bases provide a set of anchor points to form a local coordinate system, such that each data point x on the manifold can be locally approximated by a linear combination of its nearby anchor points, and the linear weights become its local coordinate coding. We show that a high dimensional nonlinear function can be approximated by a global linear function with respect to this coding scheme, and the approximation quality is ensured by the locality of such coding. The method turns a difficult nonlinear learning problem into a simple global linear learning problem, which overcomes some drawbacks of traditional local learning methods.

1 Introduction

Consider the problem of learning a nonlinear function $f(x)$ on a high dimensional space $x \in \mathbb{R}^d$. We are given a set of labeled data $(x_1, y_1), \ldots, (x_n, y_n)$ drawn from an unknown underlying distribution. Moreover, assume that we observe a set of unlabeled data $x \in \mathbb{R}^d$ from the same distribution. If the dimensionality d is large compared to n, then the traditional statistical theory predicts overfitting due to the so called "curse of dimensionality". One intuitive argument for this effect is that when the dimensionality becomes larger, pairwise distances between two similar data points become larger as well. Therefore one needs more data points to adequately fill in the empty space. However, for many real problems with high dimensional data, we do not observe this so-called curse of dimensionality. This is because although data are physically represented in a high-dimensional space, they often lie on a manifold which has a much smaller intrinsic dimensionality.

This paper proposes a new method that can take advantage of the manifold geometric structure to learn a nonlinear function in high dimension. The main idea is to locally embed points on the manifold into a lower dimensional space, expressed as coordinates with respect to a set of anchor points. Our main observation is simple but very important: we show that a nonlinear function on the manifold can be effectively approximated by a linear function with such an coding under appropriate localization conditions. Therefore using Local Coordinate Coding, we turn a very difficult high dimensional nonlinear learning problem into a much simpler linear learning problem, which has been extensively studied in the literature. This idea may also be considered as a high dimensional generalization of low dimensional local smoothing methods in the traditional statistical literature.

2 Local Coordinate Coding

We are interested in learning a smooth function $f(x)$ defined on a high dimensional space $\mathbb{R}^d$. Let $\| \cdot \|$ be a norm on $\mathbb{R}^d$. Although we do not restrict to any specific norm, in practice, one often employs the Euclidean norm (2-norm): $\|x\| = \|x\|_2 = \sqrt{x_1^2 + \cdots + x_d^2}$.

1

Definition 2.1 (Lipschitz Smoothness) *A function $f(x)$ on $\mathbb{R}^d$ is (α, β, p)-Lipschitz smooth with respect to a norm $\|\cdot\|$ if $|f(x') - f(x)| \leq \alpha\|x - x'\|$ and $|f(x') - f(x) - \nabla f(x)^\top (x' - x)| \leq \beta\|x - x'\|^{1+p}$, where we assume $\alpha, \beta > 0$ and $p \in (0, 1]$.*

Note that if the Hessian of $f(x)$ exists, then we may take $p = 1$. Learning an arbitrary Lipschitz smooth function on $\mathbb{R}^d$ can be difficult due to the curse of dimensionality. That is, the number of samples required to characterize such a function $f(x)$ can be exponential in d. However, in many practical applications, one often observe that the data we are interested in approximately lie on a manifold $\mathcal{M}$ which is embedded into $\mathbb{R}^d$. Although d is large, the intrinsic dimensionality of $\mathcal{M}$ can be much smaller. Therefore if we are only interested in learning $f(x)$ on $\mathcal{M}$, then the complexity should depend on the intrinsic dimensionality of $\mathcal{M}$ instead of d. In this paper, we approach this problem by introducing the idea of localized coordinate coding. The formal definition of (non-localized) coordinate coding is given below, where we represent a point in $\mathbb{R}^d$ by a linear combination of a set of "anchor points". Later we show it is sufficient to choose a set of "anchor points" with cardinality depending on the intrinsic dimensionality of the manifold rather than d.

Definition 2.2 (Coordinate Coding) *A coordinate coding is a pair (γ, C), where $C \subset \mathbb{R}^d$ is a set of anchor points, and γ is a map of $x \in \mathbb{R}^d$ to $[\gamma_v(x)]_{v \in C} \in R^{|C|}$ such that $\sum_v \gamma_v(x) = 1$. It induces the following physical approximation of x in $\mathbb{R}^d$: $\gamma(x) = \sum_{v \in C} \gamma_v(x)v$. Moreover, for all $x \in \mathbb{R}^d$, we define the corresponding coding norm as $\|x\|_\gamma = \left(\sum_{v \in C} \gamma_v(x)^2\right)^{1/2}$.*

The quantity $\|x\|_\gamma$ will become useful in our learning theory analysis. The condition $\sum_v \gamma_v(x) = 1$ follows from the shift-invariance requirement, which means that the coding should remain the same if we use a different origin of the $\mathbb{R}^d$ coordinate system for representing data points. It can be shown (see the appendix file accompanying the submission) that the map $x \to \sum_{v \in C} \gamma_v(x)v$ is invariant under any shift of the origin for representing data points in $\mathbb{R}^d$ if and only if $\sum_v \gamma_v(x) = 1$. The importance of the coordinate coding concept is that if a coordinate coding is sufficiently localized, then a nonlinear function can be approximate by a linear function with respect to the coding. This critical observation, illustrate in the following linearization lemma, is the foundation of our approach. Due to the space limitation, all proofs are left to the appendix that accompanies the submission.

Lemma 2.1 (Linearization) *Let (γ, C) be an arbitrary coordinate coding on $\mathbb{R}^d$. Let f be an (α, β, p)-Lipschitz smooth function. We have for all $x \in \mathbb{R}^d$:*

$$\left| f(x) - \sum_{v \in C} \gamma_v(x)f(v) \right| \leq \alpha \|x - \gamma(x)\| + \beta \sum_{v \in C} |\gamma_v(x)| \|v - \gamma(x)\|^{1+p}.$$

To understand this result, we note that on the left hand side, a nonlinear function $f(x)$ in R^d is approximated by a linear function $\sum_{v \in C} \gamma_v(x)f(v)$ with respect to the coding $\gamma(x)$, where $[f(v)]_{v \in C}$ is the set of coefficients to be estimated from data. The quality of this approximation is bounded by the right hand side, which has two terms: the first term $\|x - \gamma(x)\|$ means x should be close to its physical approximation $\gamma(x)$, and the second term means that the coding should be localized. The quality of a coding γ with respect to C can be measured by the right hand side. For convenience, we introduce the following definition, which measures the locality of a coding.

Definition 2.3 (Localization Measure) *Given α, β, p, and coding (γ, C), we define*

$$Q_{\alpha,\beta,p}(\gamma, C) = \mathbb{E}_x \left[\alpha\|x - \gamma(x)\| + \beta \sum_{v \in C} |\gamma_v(x)| \|v - \gamma(x)\|^{1+p} \right].$$

Observe that in $Q_{\alpha,\beta,p}$, α, β, p may be regarded as tuning parameters; we may also simply pick $\alpha = \beta = p = 1$. Since the quality function $Q_{\alpha,\beta,p}(\gamma, C)$ only depends on unlabeled data, in principle, we can find $[\gamma, C]$ by optimizing this quality using unlabeled data. Later, we will consider simplifications of this objective function that are easier to compute.

Next we show that if the data lie on a manifold, then the complexity of local coordinate coding depends on the intrinsic manifold dimensionality instead of d. We first define manifold and its intrinsic dimensionality.

2

Definition 2.4 (Manifold) *A subset $\mathcal{M} \subset \mathbb{R}^d$ is called a p-smooth ($p > 0$) manifold with intrinsic dimensionality $m = m(\mathcal{M})$ if there exists a constant $c_p(\mathcal{M})$ such that given any $x \in \mathcal{M}$, there exists m vectors $v_1(x), \ldots, v_m(x) \in \mathbb{R}^d$ so that $\forall x' \in \mathcal{M}$: $\inf_{\gamma \in \mathbb{R}^m} \left\| x' - x - \sum_{j=1}^m \gamma_j v_j(x) \right\| \leq c_p(\mathcal{M}) \|x' - x\|^{1+p}$.*

This definition is quite intuitive. The smooth manifold structure implies that one can approximate a point in $\mathcal{M}$ effectively using local coordinate coding. Note that for a typical manifold with well-defined curvature, we can take $p = 1$.

Definition 2.5 (Covering Number) *Given any subset $\mathcal{M} \subset \mathbb{R}^d$, and $\epsilon > 0$. The covering number, denoted as $\mathcal{N}(\epsilon, \mathcal{M})$, is the smallest cardinality of an ϵ-cover $C \subset \mathcal{M}$. That is, $\sup_{x \in \mathcal{M}} \inf_{v \in C} \|x - v\| \leq \epsilon$.*

For a compact manifold with intrinsic dimensionality m, there exists a constant $c(\mathcal{M})$ such that its covering number is bounded by $\mathcal{N}(\epsilon, \mathcal{M}) \leq c(\mathcal{M})\epsilon^{-m}$. The following result shows that there exists a local coordinate coding to a set of anchor points C of cardinality $O(m(\mathcal{M})\mathcal{N}(\epsilon, \mathcal{M}))$ such that any (α, β, p)-Lipschitz smooth function can be linearly approximated using local coordinate coding up to the accuracy $O(\sqrt{m(\mathcal{M})}\epsilon^{1+p})$.

Theorem 2.1 (Manifold Coding) *If the data points x lie on a compact p-smooth manifold $\mathcal{M}$, and the norm is defined as $\|x\| = (x^\top A x)^{1/2}$ for some positive definite matrix A. Then given any $\epsilon > 0$, there exist anchor points $C \subset \mathcal{M}$ and coding γ such that*

$$|C| \leq (1 + m(\mathcal{M}))\mathcal{N}(\epsilon, \mathcal{M}), \quad Q_{\alpha,\beta,p}(\gamma, C) \leq [\alpha c_p(\mathcal{M}) + (1 + \sqrt{m(\mathcal{M})} + 2^{1+p}\sqrt{m(\mathcal{M})})\beta]\,\epsilon^{1+p}.$$

Moreover, for all $x \in \mathcal{M}$, we have $\|x\|_\gamma^2 \leq 1 + (1 + \sqrt{m(\mathcal{M})})^2$.

The approximation result in Theorem 2.1 means that the complexity of linearization in Lemma 2.1 depends only on the intrinsic dimension $m(\mathcal{M})$ of $\mathcal{M}$ instead of d. Although this result is proved for manifolds, it is important to observe that the coordinate coding method proposed in this paper does not require the data to lie precisely on a manifold, and it does not require knowing $m(\mathcal{M})$. In fact, similar results hold even when the data only approximately lie on a manifold.

In the next section, we characterize the learning complexity of the local coordinate coding method. It implies that linear prediction methods can be used to effectively learn nonlinear functions on a manifold. The nonlinearity is fully captured by the coordinate coding map γ (which can be a nonlinear function). This approach has some great advantages because the problem of finding local coordinate coding is much simpler than direct nonlinear learning:

- Learning (γ, C) only requires unlabeled data, and the number of unlabeled data can be significantly more than the number of labeled data. This step also prevents overfitting with respect to labeled data.

- In practice, we do not have to find the optimal coding because the coordinates are merely features for linear supervised learning. This significantly simplifies the optimization problem. Consequently, it is more robust than some standard approaches to nonlinear learning that direct optimize nonlinear functions on labeled data (e.g., neural networks).

3   Learning Theory

In machine learning, we minimize the expected loss $\mathbb{E}_{x,y}\phi(f(x), y)$ with respect to the underlying distribution within a function class $f(x) \in \mathcal{F}$. In this paper, we are interested in the function class $\mathcal{F}_{\alpha,\beta,p} = \{f(x) : (\alpha, \beta, p) - \text{Lipschitz smooth function in } \mathbb{R}^d\}$.

The local coordinate coding method considers a linear approximation of functions in $\mathcal{F}_{\alpha,\beta,p}$ on the data manifold. Given a local coordinate coding scheme (γ, C), we approximate each $f(x) \in \mathcal{F}^a_{\alpha,\beta,p}$ by $f(x) \approx f_{\gamma,C}(\hat{w}, x) = \sum_{v \in C} \hat{w}_v \gamma_v(x)$, where we estimate the coefficients using ridge regression as:

$$[\hat{w}_v] = \arg\min_{[w_v]} \left[\sum_{i=1}^n \phi\left(f_{\gamma,C}(w, x_i), y_i\right) + \lambda \sum_{v \in C} (w_v - g(v))^2 \right], \tag{1}$$

3

where $g(v)$ is an arbitrary function assumed to be pre-fixed. In the Bayesian interpretation, this can be regarded as the prior mean for the weights $[w_v]_{v \in C}$. The default values of $g(v)$ are simply $g(v) \equiv 0$. Given a loss function $\phi(p, y)$, let $\phi'_1(p, y) = \partial \phi(p, y)/\partial p$. For simplicity, in this paper we only consider convex Lipschitz loss function, where $|\phi'_1(p, y)| \leq B$. This includes the standard classification loss functions such as logistic regression and SVM (hinge loss), both with $B = 1$.

Theorem 3.1 (Generalization Bound) *Suppose $\phi(p, y)$ is Lipschitz: $|\phi'_1(p, y)| \leq B$. Consider coordinate coding (γ, C), and the estimation method (1) with random training examples $S_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. Then the expected generalization error satisfies the inequality:*

$$
\mathbb{E}_{S_n} \, \mathbb{E}_{x,y} \phi(f_{\gamma,C}(\hat{w}, x), y)
$$
$$
\leq \inf_{f \in \mathcal{F}_{\alpha,\beta,p}} \left[\mathbb{E}_{x,y} \phi\left(f(x), y\right) + \lambda \sum_{v \in C} (f(v) - g(v))^2 \right] + \frac{B^2}{2\lambda n} \mathbb{E}_x \|x\|_{\gamma}^2 + B Q_{\alpha,\beta,p}(\gamma, C).
$$

We may choose the regularization parameter λ that optimizes the bound in Theorem 3.1. Moreover, if we pick $g(v) \equiv 0$, and find (γ, C) at some $\epsilon > 0$, then Theorem 2.1 implies the following simplified generalization bound for any $f \in \mathcal{F}_{\alpha,\beta,p}$ such that $|f(x)| = O(1)$: $\mathbb{E}_{x,y} \phi\left(f(x), y\right) + O\left[\sqrt{\epsilon^{-m(\mathcal{M})}/n} + \epsilon^{1+p}\right]$. By optimizing over ϵ, we obtain a bound: $\mathbb{E}_{x,y} \phi\left(f(x), y\right) + O(n^{-(1+p)/(2+2p+m(\mathcal{M}))})$.

By combining Theorem 2.1 and Theorem 3.1, we can immediately obtain the following simple consistency result. It shows that the algorithm can learn an arbitrary nonlinear function on manifold when $n \to \infty$. Note that Theorem 2.1 implies that the convergence only depends on the intrinsic dimensionality of the manifold $\mathcal{M}$, not d.

Theorem 3.2 (Consistency) *Suppose the data lie on a compact manifold $\mathcal{M} \subset \mathbb{R}^d$, and the norm $\| \cdot \|$ is the Euclidean norm in $\mathbb{R}^d$. If loss function $\phi(p, y)$ is Lipschitz. As $n \to \infty$, we choose $\alpha, \beta \to \infty$, $\alpha/n, \beta/n \to 0$ (α, β depends on n), and $p = 0$. Then it is possible to find coding (γ, C) using unlabeled data such that $|C|/n \to 0$ and $Q_{\alpha,\beta,p}(\gamma, C) \to 0$. If we pick $\lambda n \to \infty$, and $\lambda|C| \to 0$. Then the local coordinate coding method (1) with $g(v) \equiv 0$ is consistent as $n \to \infty$: $\lim_{n \to \infty} \mathbb{E}_{S_n} \, \mathbb{E}_{x,y} \phi(f(\hat{w}, x), y) = \inf_{f: \mathcal{M} \to \mathbb{R}} \mathbb{E}_{x,y} \phi\left(f(x), y\right)$.*

4 Practical Learning of Coding

Given a coordinate coding (γ, C), we can use (1) to learn a nonlinear function in $\mathbb{R}^d$. We showed that (γ, C) can be obtained by optimizing $Q_{\alpha,\beta,p}(\gamma, C)$. In practice, we may also consider the following simplifications of the localization term:

$$
\sum_{v \in C} |\gamma_v(x)| \, \|v - \gamma(x)\|^{1+p} \approx \sum_{v \in C} |\gamma_v(x)| \, \|v - x\|^{1+p} .
$$

Note that we may simply chose $p = 0$ or $p = 1$. The formulation is related to sparse coding [6] which has no locality constraints with $p = -1$. In this representation, we may either enforce the constraint $\sum_v \gamma_v(x) = 1$ or for simplicity, remove it because the formulation is already shift-invariant. Putting the above together, we try to optimize the following objective function in practice:

$$
Q(\gamma, C) = \mathbb{E}_x \inf_{[\gamma_v]} \left[\left\| x - \sum_{v \in C} \gamma_v v \right\|^2 + \mu \sum_{v \in C} |\gamma_v| \|v - x\|^{1+p} \right] .
$$

We update C and γ via alternating optimization. The step of updating γ can be transformed into a canonical LASSO problem, where efficient algorithms exist. The step of updating C is a least-squares problem in case $p = 1$.

5 Relationship to Other Methods

Our work is related to several existing approaches in the literature of machine learning and statistics. The first class of them is nonlinear manifold learning, such as LLE [8], Isomap [9], and Laplacian

4

Eigenmaps [1]. These methods find *global* coordinates of data manifold based on a pre-computed affinity graph of data points. The use of affinity graphs requires expensive computation and lacks a coherent way of generalization to new data. Our method learns a compact set of bases to form *local* coordinates, which has a linear complexity with respect to data size and can naturally handle unseen data. More importantly, local coordinate coding has a direct connection to nonlinear function approximation on manifold, and thus provides a theoretically sound unsupervised pre-training method to facilitate further supervised learning tasks.

Another set of related models are local models in statistics, such as local kernel smoothing and local regression, e.g.[4, 2], both traditionally using fixed-bandwidth kernels. Local kernel smoothing can be regarded as a zero-order method; while local regression is higher-order, including local linear regression as the 1st-order case. Traditional local methods are not widely used in machine learning practice, because data with non-uniform distribution on the manifold require to use adaptive-bandwidth kernels. The problem can be somehow alleviated by using K-nearest neighbors. However, adaptive kernel smoothing still suffers from the high-dimensionality and noise of data. On the other hand, higher-order methods are computationally expensive and prone to overfitting, because they are highly flexible in locally fitting many segments of data in high-dimension space. Our method can be seen as a generalized 1st-order local method with basis learning and adaptive locality. Compared to local linear regression, the learning is achieved by fitting a single globally linear function with respect to a set of learned local coordinates, which is much less prone to overfitting and computationally much cheaper. This means that our method achieves better balance between local and global aspects of learning. The importance of such balance has been recently discussed in [10].

Finally, local coordinate coding draws connections to vector quantization (VQ) coding, e.g., [3], and sparse coding, which have been widely applied in processing of sensory data, such as acoustic and image signals. Learning linear functions of VQ codes can be regarded as a generalized zero-order local method with basis learning. Our method has an intimate relationship with sparse coding. In fact, we can regard local coordinate coding as locally constrained sparse coding. Inspired by biological visual systems, people has been arguing sparse features of signals are useful for learning [7]. However, to the best of our knowledge, there is no analysis in the literature that directly answers the question why sparse codes can help learning nonlinear functions in high dimensional space. Our work reveals an important finding — a good first-order approximation to nonlinear function requires the codes to be local, which consequently requires the codes to be sparse. However, sparsity does not always guarantee locality conditions. Our experiments demonstrate that sparse coding is helpful for learning only when the codes are local. Therefore locality is more essential for coding, and sparsity is a consequence of such a condition.

6 Experiments

Due to the space limitation, we only include two examples: one synthetic and one real, to illustrate various aspects of our theoretical results. We note that image classification based on LCC recently achieved state-of-the-art performance in PASCAL Visual Object Classes Challenge 2009. [1]

6.1 Synthetic Data

Our first example is based on a synthetic data set, where a nonlinear function is defined on a Swiss-roll manifold, as shown in Figure 1-(1). The primary goal is to demonstrate the performance of nonlinear function learning using simple linear ridge regression based on representations obtained from traditional sparse coding and the newly suggested local coordinate coding, which are, respectively, formulated as the following,

$$\min_{\gamma, C} \ \sum_x \frac{1}{2} \|x - \gamma(x)\|^2 + \mu \sum_{v \in C} |\gamma_v(x)| \|v - x\|^2 + \lambda \sum_{v \in C} \|v\|^2 \tag{2}$$

where $\gamma(x) = \sum_{v \in C} \gamma_v(x)v$. We note that (2) is an approximation to the original formulation, mainly for the simplicity of computation.

---

[1]http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2009/workshop/index.html

5

(1) A nonlinear function   (2) RMSE=4.394   (3) RMSE=0.499   (4)RMSE=4.661

(5) RMSE=0.201   (6) RMSE=0.109   (7) RMSE=0.669   (8) RMSE=1.170
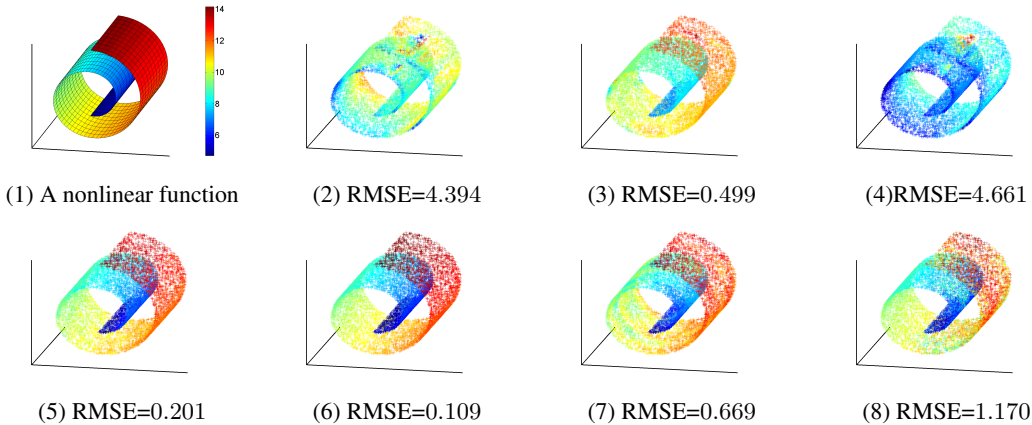
Figure 1: Experiments of nonlinear regression on Swiss-roll: (1) a nonlinear function on the Swiss-roll manifold, where the color indicates function values; (2) result of sparse coding with fixed random anchor points; (3) result of local coordinate coding with fixed random anchor points; 4) result of sparse coding; (5) result of local coordinate coding; (6) result of local kernel smoothing; (7) result of local coordinate coding on noisy data; (8) result of local kernel smoothing on noisy data.

We randomly sample $50,000$ data points on the manifold for unsupervised basis learning, and 500 labeled points for supervised regression. The number of bases is fixed to be 128. The learned nonlinear functions are tested on another set of $10,000$ data points, with their performances evaluated by root mean square error (RMSE).

In the first setting, we let both coding methods use the same set of fixed bases, which are 128 points randomly sampled from the manifold. The regression results are shown in Figure 1-(2) and (3), respectively. Sparse coding based approach fails to capture the nonlinear function, while local coordinate coding behaves much better. We take a closer look at the data representations obtained from the two different encoding methods, by visualizing the distributions of distances from encoded data to bases that have positive, negative, or zero coefficients in Figure 2. It shows that sparse coding lets bases faraway from the encoded data have nonzero coefficients, while local coordinate coding allows only nearby bases to get nonzero coefficients. In other words, sparse coding on this data does not ensure a good locality and thus fails to facilitate the nonlinear function learning. As another interesting phenomenon, local coordinate coding seems to encourage coefficients to be nonnegative, which is intuitively understandable — if we use several bases close to a data point to linearly approximate the point, each basis should have a positive contribution. However, whether there is any merit by explicitly enforcing nonnegativity will remain an interesting future work.

In the next two experiments, given the random bases as a common initialization, we let the two algorithms learn bases from the $50,000$ unlabeled data points. The regression results based on the learned bases are depicted in Figure 1-(4) and (5), which indicate that regression error is further reduced for local coordinate coding, but remains to be high for sparse coding. We also make a comparison with local kernel smoothing, which takes a weighted average of function values of K-nearest training points to make prediction. As shown in Figure 1-(6), the method works very well on this simple low-dimensional data, even outperforming the local coordinate coding approach. However, if we increase the data dimensionality to be 256 by adding 253-dimensional independent Gaussian noises with zero mean and unitary variance, local coordinate coding becomes superior to local kernel smoothing, as shown in Figure 1-(7) and (8). This is consistent with our theory, which suggests that local coordinate coding can work well in high dimension; on the other hand, local kernel smoothing is known to suffer from high dimensionality and noise.

6.2 Handwritten Digit Recognition

Our second example is based on the MNIST handwritten digit recognition benchmark, where each data point is a 28×28 gray image, and pre-normalized into a unitary 784-dimensional vector. In our setting, the set C of anchor points is obtained from sparse coding, with the regularization on

6
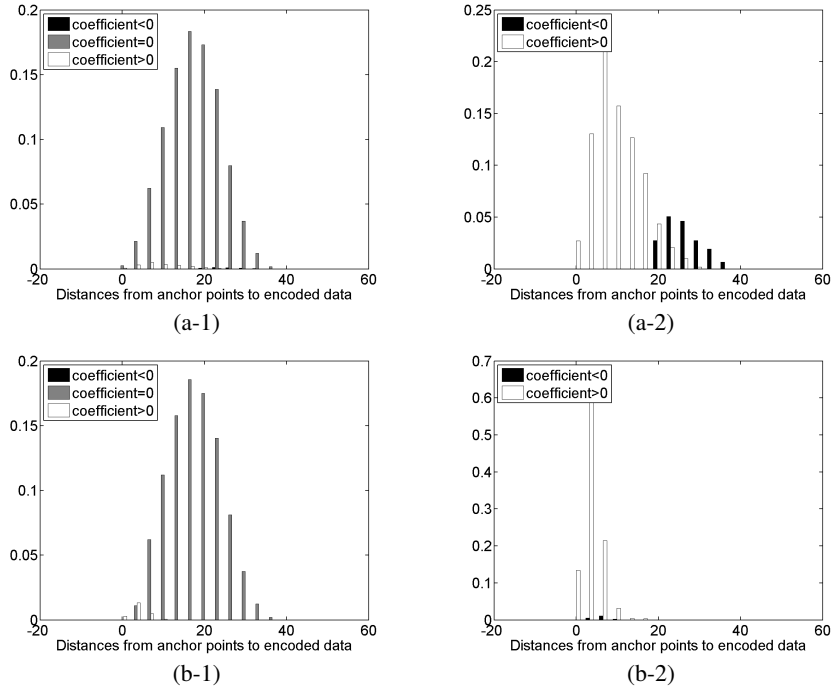
Figure 2: Coding locality on Swiss roll: (a) sparse coding vs. (b) local coordinate coding.

v replaced by inequality constraints $\|v\| \leq 1$. Our focus here is not on anchor point learning, but rather on checking whether a good nonlinear classifier can be obtained if we enforce sparsity and locality in data representation, and then apply simple one-against-all linear SVMs.

Since the optimization cost of sparse coding is invariant under flipping the sign of v, we take a postprocessing step to change the sign of v if we find the corresponding $\gamma_v(x)$ for most of x is negative. This rectification will ensure the anchor points to be on the data manifold. With the obtained C, for each data point x we solve the local coordinate coding problem (2), by optimizing γ only, to obtain the representation $[\gamma_v(x)]_{v \in C}$. In the experiments we try different sizes of bases. The classification error rates are provided in Table 1. In addition we also compare with linear classifier on raw images, local kernel smoothing based on K-nearest neighbors, and linear classifiers using representations obtained from various unsupervised learning methods, including autoencoder based on deep belief networks [5], Laplacian eigenmaps [1], locally linear embedding (LLE) [8], and VQ coding based on K-means. We note that, like most of other manifold learning approaches, Laplacian eigenmaps or LLE is a transductive method which has to incorporate both training and testing data in training. The comparison results are summarized in Table 2. Both sparse coding and local coordinate coding perform quite good for this nonlinear classification task, significantly outperforming linear classifiers on raw images. In addition, local coordinate coding is consistently better than sparse coding across various basis sizes. We further check the locality of both representations by plotting Figure-3, where the basis number is 512, and find that sparse coding on this data set happens to be quite local — unlike the case of Swiss-roll data — here only a small portion of nonzero coefficients (again mostly negative) are assigned onto the bases whose distances to the encoded data exceed the average of basis-to-datum distances. This locality explains why sparse coding works well on MNIST data. On the other hand, local coordinate coding is able to remove the unusual coefficients and further improve the locality. Among those compared methods in Table 2, we note that the error rate 1.2% of deep belief network reported in [5] was obtained via unsupervised pre-training followed by supervised backpropagation. The error rate based on unsupervised training of deep belief networks is about 1.90%.[2] Therefore our result is competitive to the-state-of-the-art results that are based on unsupervised feature learning plus linear classification without using additional image geometric information.

---

[2]This is obtained via a personal communication with Ruslan Salakhutdinov at University of Toronto.

7
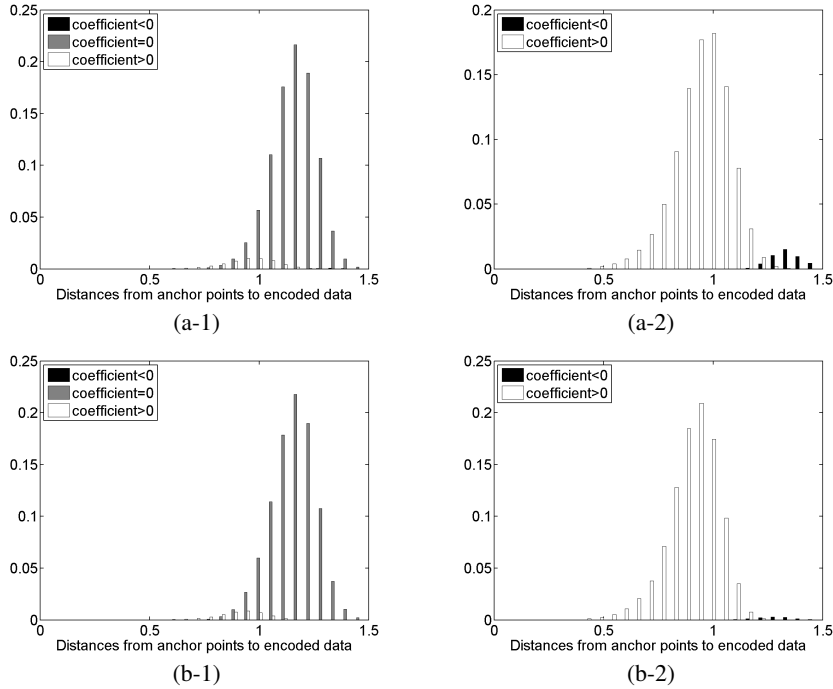
Figure 3: Coding locality on MNIST: (a) sparse coding vs. (b) local coordinate coding.

Table 1: Error rates (%) of MNIST classification with different $|C|$.

$|C|$	512	1024	2048	4096
Linear SVM with sparse coding	2.96	2.64	2.16	2.02
Linear SVM with local coordinate coding	2.64	2.44	2.08	1.90

Table 2: Error rates (%) of MNIST classification with different methods.

Methods	Error Rate
Linear SVM with raw images	12.0
Linear SVM with VQ coding	3.98
Local kernel smoothing	3.48
Linear SVM with Laplacian eigenmap	2.73
Linear SVM with LLE	2.38
Linear classifier with deep belief network	1.90
Linear SVM with sparse coding	2.02
Linear SVM with local coordinate coding	1.90

7 Conclusion

This paper introduces a new method for high dimensional nonlinear learning with data distributed on manifolds. The method can be seen as generalized local linear function approximation, but can be achieved by learning a global linear function with respect to coordinates from unsupervised local coordinate coding. Compared to popular manifold learning methods, our approach can naturally handle unseen data and has a linear complexity with respect to data size. The work also generalizes popular VQ coding and sparse coding schemes, and reveals that locality of coding is essential for supervised function learning. The generalization performance depends on intrinsic dimensionality of the data manifold. The experiments on synthetic and handwritten digit data further confirm the findings of our analysis.

8

References

[1] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373 – 1396, 2003.

[2] Leon Bottou and Vladimir Vapnik. Local learning algorithms. *Neural Computation*, 4:888 – 900, 1992.

[3] Robert M. Gray and David L. Neuhoff. Quantization. *IEEE Transaction on Information Theory*, pages 2325 – 2383, 1998.

[4] Trevor Hastie and Clive Loader. Local regression: Automatic kernel carpentry. *Statistical Science*, 8:139 – 143, 1993.

[5] Geoffrey E. Hinton and Ruslan R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504 – 507, 2006.

[6] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. *Neural Information Processing Systems (NIPS) 19*, 2007.

[7] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. Self-taught learning: Transfer learning from unlabeled data. *International Conference on Machine Learning*, 2007.

[8] Sam Roweis and Lawrence Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323 – 2326, 2000.

[9] Joshua B. Tenenbaum, Vin De Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319 – 2323, 2000.

[10] Alon Zakai and Ya'acov Ritov. Consistency and localizability. *Journal of Machine Learning Research*, 10:827 – 856, 2009.

9