Sparse Representation for Machine Learning

Yifeng Li

School of Computer Science, University of Windsor,
401 Sunset Avenue, Windsor, Ontario, N9B 3P4, Canada
li11112c@uwindsor.ca, yifeng.li.cn@gmail.com

Abstract. Sparse representation is a parsimonious principle that a signal can be approximated by a sparse superposition of basis functions. The main topic of my thesis research is to apply this principle in the machine learning fields including classification, feature extraction, feature selection, and optimization.

Keywords: sparse representation, machine learning, classification, feature extraction, feature selection, optimization.

1 Introduction

Sparse representation (SR) is a principle that a signal can be approximated by a sparse linear combination of dictionary atoms [2]. It can be formulated as $\boldsymbol{b} = x_1\boldsymbol{a}_1 + \cdots + x_k\boldsymbol{a}_k + \boldsymbol{\epsilon} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\epsilon}$, where $\boldsymbol{A} = [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_k]$ is called a *dictionary*, $\boldsymbol{a}_i$ is called a *dictionary atom* or *basis vector*, $\boldsymbol{x}$ is a sparse *coefficient vector*, and $\boldsymbol{\epsilon}$ is an error term. Sparse representation involves sparse coding and dictionary learning. Given a new signal $\boldsymbol{b}$ and dictionary $\boldsymbol{A}$, learning the sparse coefficient $\boldsymbol{x}$ is termed *sparse coding*. Given training data $\boldsymbol{D}$, learning the dictionary $\boldsymbol{A}$ is called *dictionary learning*.

For understanding SR, an example of l_1-regularized SR is given in the following from a Bayesian perspective (more details can be found in [13]). Suppose each atom $\boldsymbol{a}_i$ is normally distributed with zero mean and diagonal covariance, $\boldsymbol{x}$ follows a Laplace distribution with zero mean and diagonal covariance, and the error $\boldsymbol{\epsilon}$ follows a Gaussian distribution with zero mean and diagonal covariance. First, we fix the dictionary $\boldsymbol{A}$ to learn $\boldsymbol{x}$. Its maximum *a posteriori* (MAP) estimation can be formulated as

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1, \tag{1}$$

where $\lambda \geq 0$ is a scalar to balance the trade-off between reconstructive error and sparsity. This model is called l_1-*least-squares* (l_1LS) sparse coding. In regularization theory, it is known as a l_1-*regularized* model. Equation (1) coincides with the well-known *LASSO* [14]. Second, The l_1-regularized dictionary learning model can be expressed as

$$\min_{\boldsymbol{A},\boldsymbol{Y}} f(\boldsymbol{A},\boldsymbol{Y}) = \frac{1}{2}\|\boldsymbol{D} - \boldsymbol{A}\boldsymbol{Y}\|_F^2 + \frac{\alpha}{2}\sum_{i=1}^{k}\|\boldsymbol{a}_i\|_2^2 + \lambda\sum_{i=1}^{n}\|\boldsymbol{y}_i\|_1, \tag{2}$$

where $\alpha \geq 0$ controls the scale of the dictionary atoms.

O. Zaïane and S. Zilles (Eds.): Canadian AI 2013, LNAI 7884, pp. 352–357, 2013.

It has been reported that SR is very robust to noise and redundancy in the data [3]. The main problem I am addressing in my doctoral research is to apply the SR principle in machine learning. Since machine learning is a wide area, I focus on feature extraction, feature selection, and classification in my dissertation. I categorize the implementations of the SR principle into two groups – i) the methods using sparse coding only, ii) and the methods using both sparse coding and dictionary learning. In the subsequent sections, the problem in each group is defined and the existing solutions are surveyed. The optimization issue is also addressed. I describe my current solutions and mention future works to be completed in my thesis. My methods have been applied in various high-throughput genomic data analysis. However, due to page limit, I omit this part in this paper. Interested readers are referred to [1, 12, 13]. Hereafter, I denote the training data by $\boldsymbol{D} \in \mathbb{R}^{m \times n}$ where m and n are the numbers of features and samples, respectively. The class labels are in the column vector $\boldsymbol{c} \in \{1, 2, \cdots, C\}^n$ where C is the number of classes. A set of p new samples is represented with $\boldsymbol{B} \in \mathbb{R}^{m \times p}$.

2 Sparse Coding for Classification

2.1 Problem Statement

Sparse coding classification methods are based on the assumption that a new sample can be approximated by a sparse superposition of all training samples. Given the training data $\{\boldsymbol{D}, \boldsymbol{c}\}$, in order to predict the class label of a new sample $\boldsymbol{b}$ using sparse coding, the sparse coefficient $\boldsymbol{x}$ must be obtained first by optimizing a model, and then the class label of $\boldsymbol{b}$ is predicted by defining a decision function $g(\boldsymbol{b}|\boldsymbol{x}, \boldsymbol{D}, \boldsymbol{c}) \in \{1, 2, \cdots, C\}$.

2.2 Existing Solutions

Basis pursuit (equivalent to Equation (1)) has been applied to face recognition in [15]. First, the sparse code is learned by basis pursuit. Next, *nearest subspace* (NS) rule is used as a decision function. The NS rule is defined as $g(\boldsymbol{b}) = \arg\min_{1 \le i \le C} r_i(\boldsymbol{b})$, where $r_i(\boldsymbol{b})$ is the regression residual corresponding to the i-th class: $r_i(\boldsymbol{b}) = \|\boldsymbol{b} - \boldsymbol{A}\delta_i(\boldsymbol{x})\|_2^2$, where $\boldsymbol{A} = \boldsymbol{D}$, $\boldsymbol{\delta}_i(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}^n$ returns the coefficients for class i. Its j-th element is given by x_j, if atom $\boldsymbol{a}_j$ is in class i, otherwise 0.

In [16], a kernel extension of a l_1-model is proposed, it is equivalent to $\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{b}' - \boldsymbol{A}'\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1$, where $\boldsymbol{b}' = (\phi(\boldsymbol{A}))^{\mathrm{T}}\phi(\boldsymbol{b})$ and $\boldsymbol{A}' = (\phi(\boldsymbol{A}))^{\mathrm{T}}\phi(\boldsymbol{A})$. $\phi(\cdot)$ is a function that maps a sample from input space into high-dimensional feature space. The essence of their idea is to first map all samples in high-dimensional feature space, and then project them onto n-dimensional space by the transformation matrix $\phi(\boldsymbol{A})$. In the n-dimensional space, basis pursuit is applied.

2.3 My Contributions

Instead of using the l_1-regularized model, I propose the following non-negative sparse coding for classification [11]:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2, \quad \boldsymbol{x} \ge 0. \tag{3}$$

This is inspired by *non-negative matrix factorization* (NMF). In usual circumstance, the optimal solution to Equation (3) is very sparse. The relation between non-negativity and sparsity can be explained by either the active-set theory in optimization, or a Bernoulli prior in Bayesian inference [13]. Combining the l_1-norm and non-negativity I obtain the l_1-non-negative sparse coding model [9, 13]:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{x}, \quad \boldsymbol{x} \geq 0, \tag{4}$$

where $\boldsymbol{\lambda} = \{\lambda\}^n$. In sparse coding, I name the training samples corresponding to nonzero coefficients the *support atoms*. The rational of using non-negative sparse coding is that a unknown sample resides in the conical region of the active atoms. The minimum cone of a unknown sample may be well explained by its vertices (that is the active atoms). The classification methods of using the above two models are called *non-negative least squares* (NNLS) and l_1NNLS approaches. I propose the *k-nearest neighbor* (k-NN) based decision rule, in [13], which can take less time than the NS rule, but obtain similar accuracy. I have demonstrated that NNLS requires very few training samples in order to obtain significant accuracy. Through strict statistical comparison, it has also been shown that NNLS has a performance comparable to that of SVM.

I have extended the l_1LS, NNLS, and l_1NNLS models to kernel versions by applying the dimension-free property in sparse coding. My rational is in the following. Since least squares optimization is a specific *quadratic programming* (QP) problem, we can reformulate Equation (1) to a *l_1-regularized QP* (l_1QP) problem:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{H}\boldsymbol{x} + \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x} + \lambda\|\boldsymbol{x}\|_1, \tag{5}$$

where $\boldsymbol{H} = (\phi(\boldsymbol{A}))^{\mathrm{T}}\phi(\boldsymbol{A})$, and $\boldsymbol{g} = -(\phi(\boldsymbol{A}))^{\mathrm{T}}\phi(\boldsymbol{b})$. Similarly, the non-negative and l_1-non-negative models can be reformulated to the following non-negative QP problem:

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{H}\boldsymbol{x} + \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}, \quad \text{s.t. } \boldsymbol{x} \geq 0, \tag{6}$$

where $\boldsymbol{g} = -(\phi(\boldsymbol{A}))^{\mathrm{T}}\phi(\boldsymbol{b})$ for NNLS, and $\boldsymbol{g} = \lambda - (\phi(\boldsymbol{A}))^{\mathrm{T}}\phi(\boldsymbol{b})$ for l_1NNLS. Thus the optimization of sparse coding models is *dimension-free*. Via replacing inner products with kernel matrices, we can easily obtain the kernel sparse coding. It has been reported that my kernel sparse coding based classifier can obtain good performance [9, 13].

2.4 Future Works

First, the learning bound of sparse coding approaches will be studied under the statistical learning theory. Qualitatively speaking, the first term in Equations (1), (3), and (4) aims to minimize the empirical error, while the sparsity-inducing term is to reduce the *Vapnik-Chervonenkis dimension*. Second, the choice of an appropriate kernel is crucial in order to obtain good classification performance. Thus my future work in this direction will be focused on kernel learning for space coding approaches.

3 Dictionary Learning for Feature Extraction

3.1 Problem Statement

The sparse coding based approach is an instance-based learning. For each new sample, a large QP needs to be solved, it is hence inefficient for large-scale data. We thus need to learn a dictionary to capture the main latent patterns. For classification, dictionary learning is a scheme of dimension reduction. The classification involves three phases. First, a dictionary $\boldsymbol{A}$ is learned from training data $\boldsymbol{D}$ and possibly $\boldsymbol{c}$, that is solving the matrix decomposition $\boldsymbol{D} \approx \boldsymbol{AY}$. Columns of $\boldsymbol{Y}$ are the images of training samples in the feature space. Second, a classifier g is trained over $\boldsymbol{Y}$ and $\boldsymbol{c}$. Third, the images (denoted by $\boldsymbol{X}$) of the new samples in the feature space are obtained as well by solving the sparse coding $\boldsymbol{B} \approx \boldsymbol{AX}$, and their class labels are predicted by the classifier $g(\boldsymbol{X})$.

3.2 Existing Solutions

We can view NMF as a model of unsupervised dictionary learning. It has been used for clustering and feature extraction before my study. For instance, it has been applied to reduce the dimensionality of gene expression data [7]. New samples are usually projected into the feature space by applying pseudo-inverse $\boldsymbol{X} = \boldsymbol{A}^{\dagger}\boldsymbol{B}$. The drawback of this is that the non-negative constraint of $\boldsymbol{X}$ is violated. A kernel solution to a l_1-model was proposed in [4]. It is inefficient because the sparse code of each sample is updated separately and the dictionary atoms are not well-represented in the feature space.

3.3 My Contributions

I present a fast generic unsupervised dictionary learning framework in [13] and [8]. It solves the following two generic models:

$$\min_{\boldsymbol{A},\boldsymbol{Y}} \frac{1}{2}\|\boldsymbol{D} - \boldsymbol{AY}\|_F^2 + \lambda\|\boldsymbol{Y}\|_1 \quad \text{s.t. } \|\boldsymbol{a}_i\|_2 = 1; \text{ if } t = \text{true}, \boldsymbol{Y} \geq 0, \tag{7}$$

$$\min_{\boldsymbol{A},\boldsymbol{Y}} \frac{1}{2}\|\boldsymbol{D} - \boldsymbol{AY}\|_F^2 + \frac{\alpha}{2}\sum_{i=1}^{k}\|\boldsymbol{a}_i\|_2^2 + \lambda\sum_{i=1}^{n}\|\boldsymbol{y}_i\|_1 \quad \text{s.t. if } t = \text{true}, \boldsymbol{Y} \geq 0, \tag{8}$$

where t indicates if non-negative constraint should apply on $\boldsymbol{Y}$. The advantages of this framework are that i) $\boldsymbol{A}$ can be updated analytically; ii) columns of $\boldsymbol{Y}$ are updated in a parallel fashion; and iii) inner products among training data and dictionary are only required in optimization rather than the original data. The inner product $\boldsymbol{A}_{\phi}^{\mathrm{T}}\boldsymbol{A}_{\phi}$, rather than the intractable $\boldsymbol{A}_{\phi}$, is iteratively updated for nonlinear kernel. I also propose a supervised dictionary learning method in [10], where I reveal that the sparse coding of a new sample must be consistent with the dictionary learning model in training phase.

3.4 Future Works

First, unlike PCA and ICA, SR can learn non-orthogonal and redundant basis vectors. Independent basis vectors are selected during the sparse coding of a signal. Hence it

is interesting to investigate how accuracy changes with the number of basis vectors. Second, I plan to enforce sparsity on dictionary as well which is useful for variable selection. Third, spurred by *Bayesian factor regression modeling*, I plan to design a supervised dictionary learning model that combines dictionary learning and Bayesian regression. Finally, kernel supervised dictionary learning models will also be addressed.

4 Optimization for Sparse Representation

4.1 Problem Statement

Fast sparse coding algorithm is crucial in sparse coding and dictionary learning. Unfortunately, as in Equations (5) and (6), sparse coding is a large-scale QP problem. Moreover, the l_1-regularized models are non-smooth. Therefore, solving this QP problem efficiently for huge amount of data is an important topic in sparse representation.

4.2 Existing Solutions

There are two typical sparse coding algorithms for the l_1-regularized model. One is the interior-point method [6], and another the is proximal method [5]. The former approximates the non-smooth l_1-norm by a smooth function. The later is a first-order approach. It has been shown that first-order methods are efficient for non-smooth problems.

4.3 My Contributions

I proposed to use active-set algorithms for various sparse coding models in [8, 13]. I applied the following three properties. First, the optimization is dimension-free, therefore the input of my algorithms are inner products. Second, the active-set method is usually quite efficient for small and medium-sized problems. It thus makes dictionary learning very fast. Third, there are many common but expensive computations among the sparse coding of different signals using active-set method. My algorithms hence allow the sparse coding of multiple signals to share common computations in a parallel fashion.

4.4 Future Works

Inspired by the optimization of SVM, I am working on a *decomposition method* for large-scale sparse coding. The basic idea is in fact an implementation of the block-coordinate-descent scheme. In each iteration, a few coefficients violating the *Karush-Kuhn-Tucker* (KKT) conditions are selected in the working set, and the rest are fixed. Only the coefficients in the working set are updated by a fast QP solver. This procedure iterates until no coefficient violates the KKT conditions. *Sequential minimal optimization* (SMO) is the extreme case of the decomposition method for SVM. I am devising SMO for large-scale sparse coding.

5 Conclusions

The main topic of my thesis dissertation is to devising learning methods which apply the principle of sparse representation. The problems or challenges, and current solutions

are presented in this paper. The future works mentioned above will be finalized and included in my dissertation. Meanwhile, I am developing two open-source toolboxes [1,12] including the implementations of low level optimizations and high level machine learning applications. The purpose is to serve the machine learning community and receive constructive suggestions for my study.

Acknowledgments. I greatly thank my advisor, Dr. Alioune Ngom, and all professors for their helps in my study including Dr. Luis Rueda, Dr. B. John Oommen, Dr. Richard Caron, and Dr. Michael Ochs. My research is supported by IEEE CIS Summer Research Grant 2010, OGS Scholarship 2011-2013, NSERC Grants #RGPIN228117-2006 and #RGPIN228117-2011, CFI Grant #9263, and scholarships from University of Windsor.

References

1. The sparse representation toolbox in matlab, `http://cs.uwindsor.ca/~li11112c/sr`
2. Bruckstein, A.M., Donoho, D.L., Elad, M.: From sparse solutions of systems of equations to sparse modeling of signals and images. SIAM Review 51(1), 34–81 (2009)
3. Elad, M.: Sparse and Redundant Representations. Springer, New York (2010)
4. Gao, S., Tsang, I.W.-H., Chia, L.-T.: Kernel sparse representation for image classification and face recognition. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 1–14. Springer, Heidelberg (2010)
5. Jenatton, R., Mairal, J., Obozinski, G., Bach, F.: Proximal methods for hierarchical sparse coding. JMLR 12(2011), 2297–2334 (2011)
6. Kim, S.J., Koh, K., Lustig, M., Boyd, S., Gorinevsky, D.: An interior-point method for large-scale $l1$-regularized least squares. J-STSP 1(4), 606–617 (2007)
7. Li, Y., Ngom, A.: Non-negative matrix and tensor factorization based classification of clinical microarray gene expression data. In: BIBM, pp. 438–443. IEEE Press, Piscataway (2010)
8. Li, Y., Ngom, A.: Fast kernel sparse representation approaches for classification. In: ICDM, pp. 966–971. IEEE Press, Piscataway (2012)
9. Li, Y., Ngom, A.: Fast sparse representation approaches for the classification of high-dimensional biological data. In: BIBM, pp. 306–311. IEEE Press, Piscataway (2012)
10. Li, Y., Ngom, A.: Supervised dictionary learning via non-negative matrix factorization for classification. In: ICMLA, pp. 439–443. IEEE Press, Piscataway (2012)
11. Li, Y., Ngom, A.: Classification approach based on non-negative least squares. Neurocomputing (in press, 2013)
12. Li, Y., Ngom, A.: The non-negative matrix factorization toolbox for biological data mining. BMC Source Code for Biology and Medicine (2013), `http://cs.uwindsor.ca/~li11112c/nmf` (under revision)
13. Li, Y., Ngom, A.: Sparse representation approaches for the classification of high-dimensional biological data. BMC Systems Biology (in press, 2013)
14. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological) 58(1), 267–288 (1996)
15. Wright, J., Yang, A., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. TPAMI 31(2), 210–227 (2009)
16. Yin, J., Liu, X., Jin, Z., Yang, W.: Kernel sparse representation based classification. Neurocmputing 77, 120–128 (2012)