IDENTIFICATION OF OVERCOMPLETE DICTIONARIES AND THEIR APPLICATION IN DISTRIBUTED CLASSIFICATION PROBLEMS

BY ZAHRA SHAKERI

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Prof. Waheed U. Bajwa

and approved by

_____

_____

_____

_____

New Brunswick, New Jersey

May, 2016

ABSTRACT OF THE THESIS

Identification of overcomplete dictionaries and their application in distributed classification problems

by Zahra Shakeri

Thesis Director: Prof. Waheed U. Bajwa

The work presented in this thesis aims to study the conditions essential for reliable dictionary recovery based on the maximal response criterion and exploit the application of dictionary learning in classification of distributed data.

The first part of this thesis revisits the problem of recovery of an overcomplete dictionary in a local neighborhood from training samples using the so-called maximal response criterion. While it is known in the literature that the maximal response criterion can be used for asymptotic exact recovery of a dictionary in a local neighborhood, those results do not allow for linear (in the ambient dimension) scaling of sparsity levels in signal representations. The first contribution in this work is introducing a new condition for the sparse representation of signals and leveraging a new proof technique to establish that maximal response criterion can in fact handle linear sparsity (modulo a logarithmic factor) of signal representations. While the focus of this work is on asymptotic exact recovery, the same ideas can be used in a straightforward manner to strengthen the original maximal response criterion-based results involving noisy observations and finite number of training samples.

The second part of this thesis addresses the problem of collaborative training of nonlinear classifiers using big, distributed training data. The proposed supervised learning

strategy corresponds to data-driven joint learning of a nonlinear transformation that maps the (training) data to a higher-dimensional feature space and a ridge regression based linear classifier in the feature space. The key aspect of this work, which distinguishes it from related prior work, is that it assumes:

- The training data are distributed across a number of interconnected sites.
- Sizes of the local training data as well as privacy concerns prohibit exchange of individual training samples between sites.

The main contribution is formulation of an algorithm, termed *cloud D-KSVD*, that reliably, efficiently and collaboratively learns both the nonlinear map and the linear classifier under these constraints. In order to demonstrate the effectiveness of cloud D-KSVD, a number of numerical experiments on the MNIST dataset are also reported.

Acknowledgements

I would like to express my deepest gratitude to my adviser, Prof. Waheed Bajwa, for his guidance, constant support, encouragement, and patience. I would also like to thank him and the Electrical and Computer Engineering department for providing me with financial support for my graduate studies at Rutgers.

I am also grateful to my other thesis committee members, Prof. Anand Sarwate and Prof. Roy Yates, for taking the time to review my work and giving me valuable insights.

I would like to thank my lab members at INSPIRE laboratory for making my studies more enjoyable at Rutgers. I especially would like to thank Haroon Raja for all the discussions we had and for helping me with the second part of this thesis.

Finally, I would like to thank all my friends and family for their love and support. I am truly grateful to my husband Alireza for his moral support and being by my side in hard times.

Dedication

To my parents,

for their endless love, support, and encouragement

Table of Contents

List of Figures

Chapter 1

Introduction

Sparsity-based data processing has gained significant attention in recent years due to the explosion of data. In 2014 alone, the amount of information stored worldwide exceeded 5 ZetaBytes [1]. This number is expected to grow rapidly in the next years, hence, there is an increased demand for approaches to store and process big data. *Dictionary learning* [2, 3] is a powerful tool to obtain sparse representations of signals in computational harmonic analysis. Specifically, the task of dictionary learning corresponds to obtaining an overcomplete basis $\mathbf{D} \in \mathbb{R}^{m \times p}, p \gg m$, such that each sample in the training data is well approximated by no more than $S \ll m$ columns (*atoms*) of $\mathbf{D}$. Such a dictionary, which is a linear map from $\mathcal{F}_S = \{\mathbf{x} \in \mathbb{R}^p : \|\mathbf{x}\|_0 \leq S\}$ to $\mathbb{R}^m$, in turn (under suitable conditions on $\mathbf{D}$ and S) induces a nonlinear map $\Phi_{\mathbf{D}}$ from the *input space* $\mathbb{R}^m$ to the *feature space* $\mathcal{F}_S$ as follows:

$$\Phi_{\mathbf{D}}(\mathbf{y}) = \underset{\mathbf{x} \in \mathcal{F}_S}{\arg\min} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2. \tag{1.1}$$

In the literature, evaluation of nonlinear maps of the form (1.1) for a given $\mathbf{y} \in \mathbb{R}^m$ is termed *sparse coding* [3]. These sparse representations can then be used in a variety of applications, such as denoising [4, 5], classification [6–8], and compressed sensing [9].

The existing literature on dictionary learning ranges from designing dictionary learning algorithms to analyzing the performance of these algorithms. The proposed algorithms are used to obtain dictionaries suitable for specific tasks such as data representation and data classification, or deal with specific data such as distributed data or online data. In this thesis, we touch upon both aspects of dictionary learning.

While initial focus in the literature has been on developing efficient algorithms for dictionary learning, it is important to also understand the performance of such algorithms theoretically. To this end, our focus in Chapter 2 is on *dictionary identifiability*,

i.e. recovering the reference dictionary from generated observations, for a relatively-new maximization criterion proposed in [10, 11] for dictionary learning. The proposed criterion not only leads to an efficient computational algorithm for dictionary learning, but it is also shown in [10] that this new criterion results in provable local recovery of an $m \times p$ dictionary from training signals. Sample complexity results for dictionary learning under both noiseless and noisy settings are also provided in [10]. The common thread underlying these results is a decay constraint on sparse representations of the signals, which is a crucial element in the arguments used throughout [10]. Unfortunately, even in the best setting, the decay condition stated in [10] dictates that if the training signals have S-sparse representations in the dictionary then one must have $S = \mathcal{O}(\sqrt{m})$. Nonetheless, it is suggested in [10] that it may be possible to break this "square-root bottleneck" using different proof techniques (although no formal arguments are provided).

In Chapter 2, we revisit the maximization criterion of [10] for dictionary learning and obtain an alternative decay condition on the coefficients of the sparse representations that is less restrictive than the one obtained in [10]. Specifically, the new decay condition allows us to break the square-root bottleneck in the sense it can allow for asymptotic exact recovery of the true dictionary even if the sparse representations of the signals satisfy $S = \mathcal{O}(\frac{m}{\log p})$. Similar to [10], our focus here is on local analysis, i.e., there exists a neighborhood around the true dictionary in which only the true dictionary maximizes the objective function. Our new condition also results in a larger neighborhood compared to the one given in [10]. Our proofs rely on a new measure of dictionary coherence studied in [12, 13] as well as the *method of bounded differences* [14] and a complex variant of *Azuma's inequality* [15]. Our proof techniques can be used in a straightforward manner to improve the results reported in [10] for both noisy and finite sample settings.

There has been some prior works that focus on the theoretical guarantees of dictionary learning algorithms and required sample complexity for reliable recovery of the true underlying dictionary. Among these works, [16, 17] focus on recovery of square dictionaries, while [10, 11, 18–25] study overcomplete dictionaries. In some works such

as [17–22], global identification results for several algorithms are obtained under various assumptions on noise and the objective function. On the other hand, [16, 23] study local identifiability for objective functions without the presence of noise, while in [10, 11, 24, 25] local identifiability results are obtained for algorithms such as K-SVD [3], ITKM [10, 11], and SPAMS [26]. To the best of our knowledge, our work is the first work in the literature that formally shows $S \approx \mathcal{O}(m)$ is sufficient for reliable dictionary recovery.

Classification is one of the most important information processing tasks. There exists an extensive body of literature on training classifiers from labeled data, but much of that work assumes the training data to be available at a centralized location [27, 28]. On the other hand, many disciplines in the world today—ranging from search engines to medical informatics—are increasingly faced with scenarios in which the training data are geographically distributed across different interconnected locations (sites). While each one of the sites in this setting can rely only on its local data for supervised learning, such an approach can be suboptimal due to issues ranging from noisy local data and labels to local class imbalance. At the same time, it might be infeasible in many of these cases to gather all the distributed data at a centralized location for supervised learning due to the massive nature of these data and/or privacy concerns. The challenge in this setting then is design of a *collaborative* supervised learning framework in which individual sites collaborate with each other to approach centralized classification performance *without* exchange of individual training samples between the sites.

In Chapter 3, we undertake this challenge and develop a framework that collaboratively learns a nonlinear classifier at individual sites from the distributed training data. Our collaborative supervised learning strategy in this regard corresponds to data-driven collaborative and joint learning of a nonlinear transformation that maps (training) data in the input space $\mathbb{R}^m$ to a higher-dimensional feature space and a ridge regression based linear classifier in the feature space. In order to learn the nonlinear mapping, we resort to the framework of dictionary learning. We use the dictionary learning terminology to formally describe the goal of Chapter 3 as follows: *collaborative exploitation of labeled training data distributed across sites for joint learning of a dictionary* $\mathbf{D}$ *(equivalently,*

the nonlinear map $\Phi_{\mathbf{D}} : \mathbb{R}^m \to \mathcal{F}_S$) and a linear classification rule in $\mathcal{F}_S$.

We develop a collaborative supervised learning framework for joint dictionary learning and linear classification rule from distributed training data. Our development in this regard leverages the centralized framework of [8] for joint dictionary and classifier learning, termed *discriminative K-SVD* (D-KSVD), and the collaborative framework of [29] for *reconstructive* dictionary learning from distributed data, termed *cloud K-SVD*. We accordingly term the framework developed in this work as *cloud D-KSVD*. The second main contribution of Chapter 3 is that it evaluates the performance of cloud D-KSVD by carrying out a series of numerical experiments on the MNIST dataset of handwritten digits [30]. The results of these experiments confirm that collaborative supervised learning is superior to local supervised learning, especially in the presence of class imbalance at (some of the) individual sites. These experiments also demonstrate that the classification performance of our proposed framework not only comes very close to that of centralized supervised learning, but is also better than the classification performance of a collaborative framework based on cloud K-SVD alone.

In terms of connections to prior work, a number of dictionary learning based classifiers have been developed in the literature in recent years [7, 8, 26, 31–35]. Some of these works are based on reconstructive dictionary learning [26, 31], while others are based on discriminative dictionary learning [7, 8, 32–35]. To the best of our knowledge, however, all of these works assume the (labeled or unlabeled) training data to be available at a centralized location. Recently, the collaborative framework of cloud K-SVD was proposed in [29] for reconstructive dictionary learning. In this regard, our work can be viewed as a demonstration of the usefulness of some of the principles underlying cloud K-SVD for collaborative discriminative dictionary learning.

While our focus in Chapter 3 has been on combining the ideas in cloud K-SVD and D-KSVD due to the superior classification performance of D-KSVD in a centralized setting, it is plausible that the D-KSVD part of our collaborative framework can be replaced with some of the other (centralized) discriminative dictionary learning approaches in the literature.

Outside the realm of dictionary learning, distributed classification has been studied

in the literature in various guises. Some of the earliest interest in this topic arose in the context of distributed sensor networks [36–40]. But the distributed classification problems studied in works like [36–39] primarily focus on *fusion* of distributed data for classification, rather than collaborative training of classifiers at individual sites from distributed data. Similarly, the focus in works like [40] is on collaborative *decision making*, rather than collaborative training, using related (but different) distributed measurements of the same object. In recent years, there has also been an interest in parallelizing supervised learning algorithms [41–44]. Such works, however, are based on the premise that training (labeled) data is initially available at a centralized location.

In terms of the distribution of labeled training data, our work is most closely related to [45–56]. In [45, 46], the authors collaboratively learn kernel-linear least-squares regression estimators from training data, which can in principle also be used for classification. In [47–56], the focus is on the collaborative training of (linear and/or kernel) *support vector machines* (SVMs). Although works [47, 48] require the sites to be connected in either a fully connected [47] or a ring [48] topology, other works [49–56] can deal with more general topologies. The fundamental difference between these works and our work is that we are interested in collaborative learning of both a nonlinear map and a classifier. In the context of kernel SVM training, this would be akin to joint, collaborative learning of a kernel and an SVM. To the best of our knowledge, however, none of the earlier works address such a problem.

1.1 Notational Convention

Bold upper-case and lower-case letters are used to denote matrices and vectors, respectively. Lower-case letters denote scalars. We denote the ℓ_0, ℓ_1, and ℓ_2 norm of the vector $\mathbf{v}$ by $\|\mathbf{v}\|_0$ (number of non-zero elements of $\mathbf{v}$), $\|\mathbf{v}\|_1$, and $\|\mathbf{v}\|_2$, respectively and $\|X\|_F$ denotes the Frobenius norm of matrix $\mathbf{X}$. The k-th column of $\mathbf{X}$ is denoted by $\mathbf{x}_k$ and v_i denotes the i-th element of $\mathbf{v}$. $\mathbf{X}_{\mathcal{I}}$ is the matrix consisting of columns of $\mathbf{X}$ with indices $\mathcal{I}$. $\mathbf{e}_j$ denotes the j-th column of the identity matrix. Furthermore, $\mathbf{v}_1 \odot \mathbf{v}_2$ denotes the pointwise product of $\mathbf{v}_1$ and $\mathbf{v}_2$. We write $[K]$ for $\{1, \ldots, K\}$. For

two matrices $\mathbf{A}$ and $\mathbf{B}$ of the same dimensions $m \times p$, we define their distance to be

$$d(\mathbf{A}, \mathbf{B}) = \max_{i \in [p]} \|\mathbf{a}_i - \mathbf{b}_i\|_2. \tag{1.2}$$

For any matrix $\mathbf{X} \in \mathbb{R}^{m \times p}$ consisting of unit-norm columns, we denote the *worst-case coherence* as

$$\mu = \max_{\substack{i,j \in [p] \\ i \neq j}} |\langle \mathbf{x}_i, \mathbf{x}_j \rangle|, \tag{1.3}$$

where $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ denotes the inner product of $\mathbf{x}_i$ and $\mathbf{x}_j$. Also, we define the *average coherence* of $\mathbf{X}$ as

$$\nu = \frac{1}{p-1} \max_{i \in [p]} \Big| \sum_{\substack{j \in [p] \\ j \neq i}} \langle \mathbf{x}_i, \mathbf{x}_j \rangle \Big| \tag{1.4}$$

We use $f(\epsilon) = \mathcal{O}(g(\epsilon))$ if $\lim_{\epsilon \to 0} f(\epsilon)/g(\epsilon) = c < \infty$ for some constant c.

1.2 Thesis Outline

The rest of this thesis is organized as follows. In Chapter 2 we analyze the performance of dictionaries learned using the so-called maximal response criterion. By introducing a condition for the coefficient vector, we provide conditions for which the true dictionary maximizes the objective function, in a local neighborhood. The implications of the conditions are discussed in section 2.3.

In Chapter 3 we study the problem of learning a nonlinear classifier from distributed training data. We assume the training data is distributed among connected sites and the sites collaboratively learn a dictionary and a linear classifier, without exchanging raw data points. We demonstrate the effectiveness of the proposed scheme in numerical experiments in section 3.3.

Chapter 2

Maximal Response-Based Local Identification of Overcomplete Dictionaries

In this chapter, we address the problem of dictionary identifiability. Considering the response maximization criterion proposed in [10] for dictionary learning, we obtain conditions on the sparse representation of signals and the underling dictionary to ensure reliable recovery of the dictionary. We formulate the problem in the next section.

2.1 System Model

In dictionary learning, we assume an observation $\mathbf{y} \in \mathbb{R}^m$ is generated via

$$\mathbf{y} = \mathbf{D}_0\mathbf{x} + \mathbf{n}, \tag{2.1}$$

where $\mathbf{D}_0 \in \mathbb{R}^{m\times p}$ is a fixed dictionary, $\mathbf{x} \in \mathbb{R}^p$ is the signal coefficient vector, and $\mathbf{n} \in \mathbb{R}^m$ is the underlying noise vector. Given a signal matrix $\mathbf{Y}$ consisting of observations $\mathbf{y}_k$, $k \in [N]$, the goal is to find a representative dictionary, $\mathbf{D}$, and a coefficient matrix $\mathbf{X}$ consisting of signal coefficient vectors $\mathbf{x}_k$, $k \in [N]$, such that the representation error is minimized. In other words,

$$(\mathbf{D}^*, \mathbf{X}^*) = \underset{\mathbf{D}\in\mathcal{D},\mathbf{X}\in\mathcal{X}}{\arg\min} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2. \tag{2.2}$$

The dictionary class $\mathcal{D}$ is defined by

$$\mathcal{D} \triangleq \{\mathbf{D}' \in \mathbb{R}^{m\times p}, \|\mathbf{d}'_j\|_2 = 1 : j \in [p], \operatorname{rank}(\mathbf{D}') = m \leq p\}. \tag{2.3}$$

We also assume the non-zero singular values of $\mathbf{D} \in \mathcal{D}$ are in the interval $[\sqrt{A}, \sqrt{B}]$. We assume the coefficient vector is sparse, i.e.

$$\mathcal{X} \triangleq \{\mathbf{X}' \in \mathbb{R}^{p\times N}, \|\mathbf{x}'_j\|_0 \leq S : j \in [N]\}, \tag{2.4}$$

where S denotes the sparsity of the coefficient vector and $S \ll m$.

Similar to [10], to minimize (2.2), we use the *response maximization criterion*

$$\max_{\mathbf{D}\in\mathcal{D}} \sum_{k\in[N]} \max_{|\mathcal{I}|=S} \|\mathbf{D}_{\mathcal{I}}^{*}\mathbf{y}_k\|_1, \tag{2.5}$$

which maximizes the ℓ_1 norm of the S largest responses. We can interpret (2.5) as the generalization of the K-means objective function [10]. The asymptotic version of (2.5) can be stated as

$$\max_{\mathbf{D}\in\mathcal{D}} \mathbb{E}_{\mathbf{y}} \left(\max_{|\mathcal{I}|=S} \|\mathbf{D}_{\mathcal{I}}^{*}\mathbf{y}\|_1 \right). \tag{2.6}$$

In [10], local identifiability results are obtained using the response maximization criterion for dictionaries generated from randomly sparse signal coefficients in the presence of noise.

We now introduce the signal coefficient model. We consider a sequence $\mathbf{c} \in \mathbb{R}^p$ satisfying

$$c_1 \geq c_2 \geq \cdots \geq c_p \geq 0, \ \|\mathbf{c}\|_2 = 1. \tag{2.7}$$

We construct the signal coefficient vectors using the relation

$$\mathbf{x} = \boldsymbol{\sigma} \odot \mathbf{P}\mathbf{c}, \tag{2.8}$$

where $\mathbf{P} \in \mathbb{R}^{p\times p}$ is a random permutation matrix and $\boldsymbol{\sigma} \in \mathbb{R}^p$ is a sign vector with elements taking values ± 1 randomly. In this case, the coefficient vector $\mathbf{x}$ is equal to $\boldsymbol{\sigma} \odot \mathbf{P}\mathbf{c}$ with probability $\frac{1}{2^p p!}$, for permutation matrix $\mathbf{P}$ and the sign vector $\boldsymbol{\sigma}$. While there is no sparsity assumption on $\mathbf{x}$, additional constraints on the decay of the elements of $\mathbf{c}$ can be made to prove identifiablity results for the underlying dictionary.

2.2 Asymptotic Identifiability Results

In this section, we provide a variation of Proposition 6 in [10]. While this case is the most basic setting where noise is not present, the proof technique can be used in all theorems in [10] to improve the results stated in there.

Theorem 1. *Consider observations generated via* (2.1) *with noise variance* $\sigma = 0$*, let* $\mathbf{D} \in \mathcal{D}$ *be a dictionary with worst-case coherence* μ *and average coherence* ν*, where* $\nu \leq \mu\sqrt{\frac{\log p}{p}}$ *holds and let* $\mathbf{x}$ *be the signal coefficient generated according to* (2.8)*. If* $\mathbf{c}$ *satisfies*

$$c_S > c_{S+1} + 26\mu\sqrt{\log p}, \tag{2.9}$$

then there is a local maximum of (2.6) *at* $\mathbf{D}$ *with high probability. Moreover, for any perturbation of the true dictionary,* $\widetilde{\mathbf{D}} = (\widetilde{\mathbf{d}}_1, \ldots, \widetilde{\mathbf{d}}_p)$ *with* $d(\mathbf{D}, \widetilde{\mathbf{D}}) \leq \epsilon$*, we have* $\mathbb{E}_y\left(\max_{|\mathcal{I}|=S} \|\widetilde{\mathbf{D}}_{\mathcal{I}}^* \mathbf{y}\|_1^2\right) < \mathbb{E}_y\left(\max_{|\mathcal{I}|=S} \|\mathbf{D}_{\mathcal{I}}^* \mathbf{y}\|_1^2\right)$ *with high probability as soon as*

$$\epsilon \leq \frac{c_S - c_{S+1} - 26\mu\sqrt{\log p}}{1 + 3\sqrt{\log\left(\frac{25p^2 S\sqrt{B}}{(c_S - c_{S+1} - 26\mu\sqrt{\log p})(\sum_{i\in[S]} c_i)}\right)}}. \tag{2.10}$$

Outline of Proof: The proof of the theorem follows from steps taken in [10]:

- We show that for a fixed permutation, the maximal response is obtained by $\mathbf{D}_{\mathcal{I}_s}$, where $\mathcal{I}_s$ denotes the indices of the coefficient vector elements corresponding to $\{c_i\}_{i\in[S]}$. To this end, we introduce the decaying condition in (2.9), which is less restrictive than the decaying condition in [10] for the decay of elements of $\mathbf{c}$.

- The rest of the proof is similar to the proof of Proposition 6 in [10]. (2.6) is computed for ϵ-perturbations of the original dictionary, i.e. $d(\mathbf{D}, \widetilde{\mathbf{D}}) \leq \epsilon$, and it is shown that for small perturbations of the original dictionary and most sign sequences, the maximal response is obtained by $\widetilde{\mathbf{D}}_{\mathcal{I}_s}$. Using arguments on the loss of $\widetilde{\mathbf{D}}$ over the typical sign sequence of all permutations compared to the maximal gain over approximately atypical sign sequences, it is shown that $\mathbf{D}$ maximizes (2.6) locally.

We introduce a lemma essential to prove Theorem 1.

Lemma 1. *Consider observations generated according to* (2.1) *with noise variance* $\sigma = 0$*, where the dictionary* $\mathbf{D} \in \mathcal{D}$ *has worst-case coherence* μ *and average coherence* ν*, and let the coefficient vector be generated according to* (2.8)*. Then, for any* $i \in [p]$*,*

any $t > 0$, and any p satisfying $\sqrt{p} \leq t/\nu$, we have

$$\mathbb{P}\left(\left| \sum_{\substack{j \in [p] \\ j \neq i}} x_j \langle \mathbf{d}_i, \mathbf{d}_j \rangle \right| > t \right) \leq 4 \exp\left(-\frac{(t - \nu\sqrt{p})^2}{144\mu^2} \right). \tag{2.11}$$

2.3 Discussion

The condition $\nu \leq \mu\sqrt{\frac{\log p}{p}}$ in Theorem 1 is implied by conditions $\frac{p}{\log p} \leq m$ and $\nu \leq \frac{\mu}{\sqrt{m}}$ and according to [12], there exist dictionaries that satisfy $\nu \leq \frac{\mu}{\sqrt{m}}$.

To analyze our result and compare it to the analogous result in [10], we study the basic setting where $\mathbf{c}$ is S-sparse and $\{c_i\}_{i=1}^{S} = \frac{1}{\sqrt{S}}$, resulting in $\|\mathbf{c}\|_1 = \sqrt{S}$. According to the decay condition in [10], $c_S > c_{S+1} + 2\mu\|\mathbf{c}\|_1$, we have recovery of the true dictionary as long as $S < \frac{1}{2\mu}$. From the Welch bound [57], this translates to sparsity levels of order $\mathcal{O}(\sqrt{m})$. With the new decay condition $c_S > c_{S+1} + 26\mu\sqrt{\log p}$, we can recover the true dictionary as long as sparsity levels are of order $\mathcal{O}(\frac{m}{\log p})$. Hence, we are able to overcome the fundamental limitations of [10] where, regardless of the dictionary, there is a square-root bottleneck for S, whereas, we get close to a linear scaling. Although we have only studied the noiseless asymptotic case, the decay condition for the coefficient vector can also be used in noisy and finite sample settings.

2.4 Appendix

Lemma 2 (The Complex Azuma's Inequality [12])**.** *Assuming the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, let $\widetilde{M}_1, \ldots, \widetilde{M}_n$ be a complex-valued martingale difference sequence on $(\Omega, \mathcal{F}, \mathbb{P})$ with $|\widetilde{M}_i| \leq c_i$ for $i \in [n]$. Then for any $t > 0$,*

$$\mathbb{P}\left(\left| \sum_{i \in [n]} \widetilde{M}_i \right| \geq t \right) \leq 4 \exp\left(-\frac{t^2}{4 \sum_{i \in [n]} c_i^2} \right). \tag{2.12}$$

2.4.1 Proof of Lemma 1

The proof of the lemma follows similar steps as Lemma 3 in [12]. Assuming the permutation matrix $\mathbf{P} \in \mathbb{R}^{p \times p}$:

$$\mathbf{P} = [\mathbf{e}_{\pi(1)}, \mathbf{e}_{\pi(2)}, \ldots, \mathbf{e}_{\pi(p)}]^T, \tag{2.13}$$

the measurement vector $\mathbf{y}$ can be stated as

$$\mathbf{y} = \mathbf{D}(\boldsymbol{\sigma} \odot \mathbf{P}\mathbf{c}) = \mathbf{D}_{\Pi}(\boldsymbol{\sigma} \odot \mathbf{c}), \tag{2.14}$$

where $\Pi = \{\pi(i)\}_{i=1}^{p}$ and $\mathbf{D}_{\Pi}$ is the column-wise permuted version of $\mathbf{D}$. We now introduce the method of bounded differences (MOBD) [14] that uses Azuma's inequality for bounded martingale difference sequences (BMDS). For a fixed index i, conditioned on the event $\mathcal{A}_{i'} = \{\pi(i) = i'\}$ and the sign vector $\boldsymbol{\sigma}$, writing the coefficient vector elements as $x_j = \sigma_j c_{\pi(j)}, j \in [p]$, we get

$$\mathbb{P}\bigg(\bigg|\sum_{\substack{j\in[p]\\ j\neq i'}} \sigma_j c_{\pi(j)} \langle \mathbf{d}_{\pi(i)}, \mathbf{d}_j\rangle\bigg| > t \big| \mathcal{A}_{i'}, \boldsymbol{\sigma}\bigg) = \mathbb{P}\bigg(\bigg|\sum_{\substack{j\in[p]\\ j\neq i}} \sigma_{\pi^{-1}(j)} c_j \langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)}\rangle\bigg| > t \big| \mathcal{A}_{i'}, \boldsymbol{\sigma}\bigg). \tag{2.15}$$

To obtain an upper bound for (2.15), we define a random $(p-1)$-tuple $\Pi^{-i} = \{\pi(k)\}_{k=1}^{p}, k \neq i$ and construct a Doob Martingale $(M_0, M_1, \ldots, M_{p-1})$:

$$\begin{aligned} M_0 &= \mathbb{E}\bigg[\sum_{\substack{j\in[p]\\ j\neq i}} \sigma_{\pi^{-1}(j)} c_j \langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)}\rangle \big| \mathcal{A}_{i'}, \boldsymbol{\sigma}\bigg], \text{and} \\ M_\ell &= \mathbb{E}\bigg[\sum_{\substack{j\in[p]\\ j\neq i}} \sigma_{\pi^{-1}(j)} c_j \langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)}\rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell}, \mathcal{A}_{i'}, \ \boldsymbol{\sigma}\bigg], \end{aligned} \tag{2.16}$$

for $\ell \in [p-1]$, where $\{\pi_k^{-i}\}_{k=1}^{\ell}$ denotes the first ℓ elements of $\prod^{-i}$. Similar to [12], we can bound $|M_0|$ by

$$\begin{aligned} |M_0| &= \bigg|\mathbb{E}\bigg[\sum_{\substack{j\in[p]\\ j\neq i}} \sigma_{\pi^{-1}(j)} c_j \langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)}\rangle \big| \mathcal{A}_{i'}, \boldsymbol{\sigma}\bigg]\bigg| \\ &\leq \sum_{\substack{j\in[p]\\ j\neq i}} \big|\sigma_{\pi^{-1}(j)} c_j \mathbb{E}\left[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)}\rangle \big| \mathcal{A}_{i'}, \boldsymbol{\sigma}\right]\big| \\ &\leq \sum_{\substack{j\in[p]\\ j\neq i}} c_j \bigg|\sum_{\substack{q\in[p]\\ q\neq i'}} \frac{1}{p-1} \langle \mathbf{d}_{i'}, \mathbf{d}_q\rangle\bigg| \\ &\leq \nu \|\mathbf{c}\|_1 \\ &\leq \nu\sqrt{p}\|\mathbf{c}\|_2 \\ &= \nu\sqrt{p}. \end{aligned} \tag{2.17}$$

To utilize Azuma's Inequality, we have to construct a BMDS from $(M_0, \ldots, M_{p-1})$. Defining $\widetilde{M}_\ell = M_\ell - M_{\ell-1}$ for $\ell \in [p-1]$, it is necessary to find the upper bound $|\widetilde{M}_\ell|$. According to [58], we have $|\widetilde{M}_\ell| \leq \sup_{r,s}[M_\ell(r) - M_\ell(s)]$ where $M_\ell(r)$ is defined as

$$M_\ell(r) \triangleq \mathbb{E}\Big[\sum_{\substack{j\in[p]\\ j\neq i}} \sigma_{\pi^{-1}(j)} c_j \langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = r, \mathcal{A}_{i'}, \boldsymbol{\sigma} \Big], \tag{2.18}$$

for $\ell \in [p-1]$. To find an upper bound for $|M_\ell(r) - M_\ell(s)|$, we have

$$\begin{aligned}
|M_\ell(r) - M_\ell(s)| = \Big| \sum_{\substack{j\in[p]\\ j\neq i}} \sigma_{\pi^{-1}(j)} c_j \big(\mathbb{E}\big[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = r, \mathcal{A}_{i'}, \boldsymbol{\sigma} \big] \\
- \mathbb{E}\big[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = s, \mathcal{A}_{i'}, \boldsymbol{\sigma} \big] \big) \Big| \\
\leq \sum_{\substack{j\in[p]\\ j\neq i}} c_j \Big| \mathbb{E}\Big[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = r, \mathcal{A}_{i'}, \boldsymbol{\sigma} \Big] \\
- \mathbb{E}\Big[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = s, \mathcal{A}_{i'}, \boldsymbol{\sigma} \Big] \Big| \\
= \sum_{\substack{j\leq \ell+1\\ j\neq i}} c_j |d_{\ell,j}| + \sum_{\substack{j> \ell+1\\ j\neq i}} c_j |d_{\ell,j}|,
\end{aligned} \tag{2.19}$$

where

$$d_{l,j} \triangleq \mathbb{E}\big[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = r, \mathcal{A}_{i'}, \boldsymbol{\sigma} \big] - \mathbb{E}\big[\langle \mathbf{d}_{i'}, \mathbf{d}_{\pi(j)} \rangle \big| \{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = s, \mathcal{A}_{i'}, \boldsymbol{\sigma} \big]. \tag{2.20}$$

We consider various cases to upper bound (2.19). For the case where $\ell \notin [p-3]$, Π is deterministic. In this case, if $i \leq \ell$,

$$\begin{aligned}
\sum_{\substack{j\in[\ell+1]\\ j\neq i}} c_j |d_{\ell,j}| &= c_{\ell+1} |\langle \mathbf{d}_{i'}, \mathbf{d}_r \rangle - \langle \mathbf{d}_{i'}, \mathbf{d}_s \rangle| \\
&\leq 2\mu c_{\ell+1}.
\end{aligned} \tag{2.21}$$

Similarly, if $i > \ell$, $\sum_{\substack{j\in[\ell+1]\\ j\neq i}} c_j |d_{\ell,j}| \leq 2\mu c_\ell$.

If $\ell \in [p-3]$, for any $j > \ell+1, j \neq i$, $\pi(j)$ has a uniform distribution over $[p] - \{\{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = r, \mathcal{A}_{i'}\}$ and $[p] - \{\{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = s, \mathcal{A}_{i'}\}$, conditioned on

$\{\{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = r, \mathcal{A}_{i'}\}$ and $\{\{\pi_k^{-i}\}_{k=1}^{\ell-1}, \pi_\ell^{-i} = s, \mathcal{A}_{i'}\}$, respectively and we have

$$\begin{aligned} |d_{\ell,j}| &= \frac{1}{p-\ell-1}|\langle \mathbf{d}_{i'}, \mathbf{d}_r\rangle - \langle \mathbf{d}_{i'}, \mathbf{d}_s\rangle| \\ &\leq \frac{2\mu}{p-\ell-1}. \end{aligned} \tag{2.22}$$

If $\ell \in [p-3]$, for any $j \leq \ell+1$, we study three cases for i. If $i < \ell$, $\sum_{\substack{j\in[\ell+1]\\ j\neq i}} c_j|d_{\ell,j}| \leq 2\mu c_{\ell+1}$. If $i = \ell$, $\sum_{\substack{j\in[\ell+1]\\ j\neq i}} c_j|d_{\ell,j}| \leq 2\mu c_\ell$ and if $i > \ell+1$,

$$\sum_{\substack{j\in[\ell+1]\\ j\neq i}} c_j|d_{\ell,j}| \leq 2\mu(c_\ell + \frac{c_{\ell+1}}{p-\ell-1}). \tag{2.23}$$

Denoting $d_\ell \triangleq \sum_{\substack{j\in[p]\\ j\neq i}} c_j|d_{\ell,j}|$, we have $\sup_{r,s}[M_\ell(r) - M_\ell(s)] \leq 2\mu d_\ell$, where

$$d_\ell = \begin{cases} c_\ell + c_{\ell+1} + \frac{1}{p-\ell-1}\sum_{j=\ell+2}^{p} c_j, & \ell \in [p-3], \\ c_\ell & \ell \notin [p-3]. \end{cases} \tag{2.24}$$

To use the complex Azuma's inequality, it is necessary to upper bound $\sum_{\ell\in[p-1]} d_\ell^2$:

$$\begin{aligned} \sum_{\ell\in[p-1]} d_\ell^2 &= \sum_{\ell\in[p-3]} \big(c_\ell + c_{\ell+1} + \frac{1}{p-\ell-1}\sum_{j=\ell+2}^{p} c_j\big)^2 + \sum_{\ell=p-2}^{p-1} c_\ell^2 \\ &= \sum_{\ell\in[p-3]} \Big(c_\ell^2 + c_{\ell+1}^2 + 2c_\ell c_{\ell+1} + \frac{2(c_\ell + c_{\ell+1})}{p-\ell-1}\sum_{j=\ell+2}^{p} c_j + \big(\frac{1}{p-\ell-1}\sum_{j=\ell+2}^{p} c_j\big)^2\Big) \\ &\quad + c_{p-2}^2 + c_{p-1}^2. \end{aligned} \tag{2.25}$$

Since $\mathbf{c}$ is non-negative and non-increasing, $2c_\ell c_{\ell+1} \leq 2c_\ell$ and we can write

$$\sum_{\ell=1}^{p-3} c_\ell^2 + c_{\ell+1}^2 + 2c_\ell c_{\ell+1} \leq 4\|\mathbf{c}\|_2^2 - c_{p-2}^2 - c_{p-1}^2. \tag{2.26}$$

Denoting $\|\mathbf{c}\|_1^{-n} \triangleq \|\mathbf{c}\|_1 - \sum_{i\in[n]} c_i$, which has $p-n$ elements, we have $\|\mathbf{c}\|_1^{-n} \leq (p-n)c_{n+1}$. Therefore,

$$\begin{aligned} \sum_{\ell\in[p-3]} \frac{2(c_\ell + c_{\ell+1})}{p-\ell-1}\sum_{j=\ell+2}^{p} c_j &= \sum_{\ell\in[p-3]} \frac{2(c_\ell + c_{\ell+1})\|\mathbf{c}\|_1^{-(\ell+1)}}{p-\ell-1} \\ &\leq \sum_{\ell\in[p-3]} \frac{4c_\ell(p-\ell-1)c_{\ell+2}}{p-\ell-1} \\ &= \sum_{\ell\in[p-3]} 4c_\ell c_{\ell+2} \\ &\leq 4\|\mathbf{c}\|_2^2. \end{aligned} \tag{2.27}$$

Similarly, we have

$$\sum_{\ell\in[p-3]}\left(\frac{1}{p-\ell-1}\sum_{j=\ell+2}^{p}c_j\right)^2=\sum_{\ell\in[p-3]}\left(\frac{\|\mathbf{c}\|_1^{-(\ell+1)}}{p-\ell-1}\right)^2$$
$$\leq\sum_{\ell\in[p-3]}c_{\ell+2}^2$$
$$\leq\|\mathbf{c}\|_2^2. \tag{2.28}$$

Adding the upper bounds in (2.26), (2.27), and (2.28) for (2.25) results in

$$\sum_{\ell\in[p-1]}d_\ell^2\leq 9\|\mathbf{c}\|_2^2. \tag{2.29}$$

We have established that $(\widetilde{M}_1,\ldots,\widetilde{M}_{p-1})$ is a BDMS with $|\widetilde{M}_\ell|\leq 2\mu d_\ell$ for $\ell\in[p-1]$. We have

$$\mathbb{P}\bigg(\Big|\sum_{\substack{j\in[p]\\j\neq i}}\sigma_{\pi^{-1}(j)}c_j\langle\mathbf{d}_{i'},\mathbf{d}_{\pi(j)}\rangle\Big|>t\big|\mathcal{A}_{i'},\boldsymbol{\sigma}\bigg)$$
$$\overset{(a)}{\leq}\mathbb{P}\left(|M_{p-1}-M_0|>t\|\mathbf{c}\|_2-\nu\sqrt{p}\|\mathbf{c}\|_2\big|\mathcal{A}_{i'},\boldsymbol{\sigma}\right)$$
$$=\mathbb{P}\bigg(\Big|\sum_{i\in[p-1]}\widetilde{M}_i\Big|>t\|\mathbf{c}\|_2-\nu\sqrt{p}\|\mathbf{c}\|_2\big|\mathcal{A}_{i'},\boldsymbol{\sigma}\bigg)$$
$$\overset{(b)}{\leq}4\exp\left(-\frac{(t-\nu\sqrt{p})^2\|\mathbf{c}\|_2^2}{16\mu^2\sum_{\ell=1}^{p-1}d_\ell}\right)$$
$$\leq 4\exp\left(-\frac{(t-\nu\sqrt{p})^2}{144\mu^2}\right), \tag{2.30}$$

where (a) follows from (2.17) and (b) follows from the complex Azuma's inequality for BDMS in Lemma 2. Taking the union bound over all events $\mathcal{A}_{i'}$ and sign sequences, we have

$$\mathbb{P}\bigg(\Big|\sum_{\substack{j\in[p]\\j\neq i}}\sigma_{\pi^{-1}(j)}c_j\langle\mathbf{d}_{i'},\mathbf{d}_{\pi(j)}\rangle\Big|>t\bigg)$$
$$\leq\sum_{j\in[p]}\sum_{i'\in[p]}\mathbb{P}\bigg(\Big|\sum_{\substack{j\in[p]\\j\neq i}}\sigma_{\pi^{-1}(j)}c_j\langle\mathbf{d}_{i'},\mathbf{d}_{\pi(j)}\rangle\Big|>t\|\mathbf{c}\|_2\big|\mathcal{A}_{i'},\boldsymbol{\sigma}\bigg)\mathbb{P}(\mathcal{A}_{i'})\mathbb{P}(\sigma_j)$$
$$\leq 4\exp\left(-\frac{(t-\nu\sqrt{p})^2}{144\mu^2}\right), \tag{2.31}$$

where i' can be replaced with any $i\in[p], i\neq i'$ and the inequality holds for all i.

2.4.2 Proof of Theorem 1

The objective function in (2.6) can be restated as

$$\mathbb{E}_y\left(\max_{|\mathcal{I}|=S}\|\mathbf{D}^*_{\mathcal{I}}\mathbf{y}\|_1\right) = \mathbb{E}_\pi\mathbb{E}_\sigma\left(\max_{|\mathcal{I}|=S}\|\mathbf{D}^*_{\mathcal{I}}\mathbf{D}\mathbf{x}\|_1\right)$$
$$= \mathbb{E}_\pi\mathbb{E}_\sigma\left(\max_{|\mathcal{I}|=S}\sum_{i\in\mathcal{I}}|\langle\mathbf{d}_i,\mathbf{D}\mathbf{x}\rangle|\right). \tag{2.32}$$

We now show that the maximum of (2.32) is obtained via $\mathcal{I}=\mathcal{I}_s$, where $\mathcal{I}_s = \pi^{-1}(\{1,2,\ldots,S\})$.

Selecting $t = 13\mu\sqrt{\log p}$, as long as the condition $\nu \le \mu\sqrt{\frac{\log p}{p}}$ is satisfied, we have $t-\nu\sqrt{p}\ge 0$ and $\exp\left(-\frac{(t-\nu\sqrt{p})^2}{144\mu^2}\right)\le p^{-1}$. Therefore, with high probability, for any $i\in\mathcal{I}_s$, we have

$$|\langle\mathbf{d}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| = \big|\sigma_i c_{\pi(i)} + \sum_{\substack{j\in[p]\\ j\neq i}}\sigma_j c_{\pi(j)}\langle\mathbf{d}_i,\mathbf{d}_j\rangle\big|$$
$$\overset{(c)}{\ge} c_S - \big|\sum_{\substack{j\in[p]\\ j\neq i}}\sigma_j c_{\pi(j)}\langle\mathbf{d}_i,\mathbf{d}_j\rangle\big|$$
$$\overset{(d)}{\ge} c_S - 13\mu\sqrt{\log p}, \tag{2.33}$$

where (c) follows from the triangle inequality and (d) follows from substituting $\epsilon = 13\mu\sqrt{\log p}$ in (2.11). Similarly, for all $i\notin\mathcal{I}_s$, we have

$$|\langle\mathbf{d}_i,\mathbf{D}c_{\pi,\sigma}\rangle| = \big|\sigma_i c_i + \sum_{\substack{j\in[p]\\ j\neq i}}\sigma_j c_{\pi(j)}\langle\mathbf{d}_i,\mathbf{d}_j\rangle\big|$$
$$\le c_{S+1} + \big|\sum_{\substack{j\in[p]\\ j\neq i}}\sigma_j c_{\pi(j)}\langle\mathbf{d}_i,\mathbf{d}_j\rangle\big|$$
$$\le c_{S+1} + 13\mu\sqrt{\log p}, \tag{2.34}$$

with high probability. Thus, the condition $c_S > c_{S+1} + 26\mu\sqrt{\log p}$ ensures that the maximum of the objective function is attained at $\mathcal{I}_s$. The next steps follow similarly

from [10]. We can write (2.32) as

$$\begin{aligned}\mathbb{E}_y\left(\max_{|\mathcal{I}|=S}\|\mathbf{D}_{\mathcal{I}}^*\mathbf{y}\|_1\right) &= \mathbb{E}_\pi\mathbb{E}_\sigma\left(\|\mathbf{D}_{\mathcal{I}_s}^*\mathbf{D}\mathbf{x}\|_1\right)\\ &= \mathbb{E}_\pi\mathbb{E}_\sigma\left(\sum_{i\in\mathcal{I}_s}\left|c_{\pi(i)}+\sigma_i\sum_{\substack{j\in[p]\\ j\neq i}}\sigma_j c_{\pi(j)}\langle\mathbf{d}_i,\mathbf{d}_j\rangle\right|\right)\\ &= c_1+\cdots+c_S. \end{aligned} \tag{2.35}$$

Now, we compute the expectation for the perturbation dictionary $\widetilde{\mathbf{D}}$ with the distance $d(\mathbf{D},\widetilde{\mathbf{D}}) = \epsilon$ from the original dictionary $\mathbf{D}$. It is clear that $\|\mathbf{d}_i-\widetilde{\mathbf{d}}_i\|_2 = \epsilon_i$ and $\max_i \epsilon_i = \epsilon$. We can state $\widetilde{\mathbf{d}}_i$ as

$$\widetilde{\mathbf{d}}_i = \alpha_i\mathbf{d}_i+\beta_i\mathbf{z}_i,\ i\in[p], \tag{2.36}$$

where $\alpha_i \triangleq \left(1-\frac{\epsilon_i^2}{2}\right)$, $\beta_i \triangleq \left(\epsilon_i^2-\frac{\epsilon_i^4}{4}\right)^{\frac{1}{2}}$, and $\mathbf{z}_i$ satisfies $\langle\mathbf{d}_i,\mathbf{z}_i\rangle = 0$, $\|\mathbf{z}_i\|_2 = 1$. Hence,

$$\mathbb{E}_y\left(\max_{|\mathcal{I}|=S}\|\widetilde{\mathbf{D}}_{\mathcal{I}}^*\mathbf{y}\|_1\right) = \mathbb{E}_\pi\mathbb{E}_\sigma\left(\max_{|\mathcal{I}|=S}\sum_{i\in\mathcal{I}}|\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle|\right). \tag{2.37}$$

We show that for perturbed dictionary $\widetilde{\mathbf{D}}$ and most sign sequences, the maximum of (2.37) is also attained by $\mathcal{I}_s$. For all $i\in\mathcal{I}_s$ we have

$$|\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| \overset{(e)}{\geq} \alpha_i(c_S-13\mu\sqrt{\log p})-\beta_i|\langle\mathbf{z}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle|, \tag{2.38}$$

where (e) follows from (2.33). For all $i\notin\mathcal{I}_s$ we have

$$|\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| \overset{(f)}{\leq} \alpha_i(c_{S+1}+13\mu\sqrt{\log p})+\beta_i|\langle\mathbf{z}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle|, \tag{2.39}$$

where (f) follows from (2.34). Using Hoeffding's inequality, we get

$$\mathbb{P}\left(\beta_i|\langle\mathbf{z}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle|\geq t\right) \leq 2\exp\left(-\frac{t^2}{2\epsilon_i^2}\right). \tag{2.40}$$

Therefore, except with probability $2\exp\left(-\frac{t^2}{2\epsilon_i^2}\right)$, we have

$$\begin{aligned}|\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| &\geq \alpha_i(c_S-13\mu\sqrt{\log p})-t, \forall i\in\mathcal{I}_s,\\ |\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| &\leq \alpha_i(c_{S+1}+13\mu\sqrt{\log p})+t, \forall i\notin\mathcal{I}_s.\end{aligned} \tag{2.41}$$

Setting $t \triangleq \frac{1}{2}(c_S-c_{S+1}-26\mu\sqrt{\log p}-\frac{\epsilon^2}{2})$, we ensure that

$$\max_{|\mathcal{I}|=S}\sum_{i\in\mathcal{I}}|\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| = \sum_{i\in\mathcal{I}_s}|\langle\widetilde{\mathbf{d}}_i,\mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle|. \tag{2.42}$$

The next step is to compute the expectation of (2.42) over $\boldsymbol{\sigma}$. For this purpose, for each permutation π, we define two sets. One set is the set of all sign sequences that results in $\beta_i|\langle \mathbf{z}_i, \mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| \geq t$, while the second set is the set of all sign sequences that results in $\beta_i|\langle \mathbf{z}_i, \mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle| < t$. Following similar steps as [10], we get

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\sigma}}\left(\sum_{i\in\mathcal{I}_s}|\langle\widetilde{\mathbf{d}}_i, \mathbf{D}\mathbf{c}_{\pi,\sigma}\rangle|\right) &= \mathbb{E}_{\boldsymbol{\sigma}}\left(\sum_{i\in\mathcal{I}_s}\left|\alpha_i c_{\pi(i)} + \sigma_i\langle\alpha_i\mathbf{d}_i + \beta_i\mathbf{z}_i, \sum_{\substack{j\in[p]\\ j\neq i}}\sigma_j c_{\pi(j)}\mathbf{d}_j\rangle\right|\right)\\ &= \sum_{i\in\mathcal{I}_s}\alpha_i c_{\pi(i)} \end{aligned} \tag{2.43}$$

and the form of the objective function for the perturbed dictionary becomes

$$\begin{aligned}&\mathbb{E}_{\mathbf{y}}\left(\max_{|\mathcal{I}|=S}\|\widetilde{\mathbf{D}}_{\mathcal{I}}^*\mathbf{y}\|_1\right)\\ &\qquad\leq 4\epsilon S\sqrt{B}\sum_{\substack{i\in[p]\\ \epsilon_i\neq 0}}\exp\left(-\frac{(c_S - c_{S+1} - 26\mu\sqrt{\log p} - \frac{\epsilon^2}{2})^2}{8\epsilon_i^2}\right) + \frac{c_1+\cdots+c_S}{p}\sum_{i\in[p]}\alpha_i.\end{aligned} \tag{2.44}$$

To ensure $\mathbb{E}_{\mathbf{y}}\left(\max_{|\mathcal{I}|=S}\|\mathbf{D}_{\mathcal{I}}^*\mathbf{y}\|_1\right) > \mathbb{E}_{\mathbf{y}}\left(\max_{|\mathcal{I}|=S}\|\widetilde{\mathbf{D}}_{\mathcal{I}}^*\mathbf{y}\|_1\right)$, the following condition arises:

$$\epsilon > \frac{8Sp^2\sqrt{B}}{c_1+\cdots+c_S}\exp\left(-\frac{(c_S - c_{S+1} - 26\mu\sqrt{\log p} - \frac{\epsilon^2}{2})^2}{8\epsilon_i^2}\right), \tag{2.45}$$

which is ensured by (2.10).

Chapter 3

Dictionary Learning Based Nonlinear Classifier Training from Distributed Data

In this Chapter, dictionary learning is employed to design a non-linear classifier from distributed training data. We consider a distributed setting where training data is distributed among sites. Our goal is for the sites to collaboratively learn a joint dictionary that transforms the data to a higher dimension and a linear classifier to classify transformed data. We formulate the problem formally in the next section.

3.1 Problem Formulation

Consider a collection of K interconnected sites. We express this collection through an undirected, connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = [K]$ and $\mathcal{E} = \{(i,j) \in \mathcal{V} \times \mathcal{V} : \text{sites } i \text{ and } j \text{ are connected}\}$. Each of these K sites is interested in classifying p-dimensional data into one of L possible classes. In order to facilitate this classification task, we assume each site i has access to N_i labeled training samples $\{(\mathbf{y}_i^j, \ell_i^j)\}_{j=1}^{N_i}$, where $\mathbf{y}_i^j \in \mathbb{R}^m$ denotes a training sample, $\ell_i^j \in \mathcal{L}$ denotes the label of $\mathbf{y}_i^j$, and $\mathcal{L} = [L]$. Given these $N = \sum_{i \in \mathcal{V}} N_i$ labeled training samples distributed across different sites, we are interested in collaboratively and jointly learning a nonlinear (feature) map $\Phi_{\mathbf{D}}$ and a linear classifier $\mathcal{C}$ such that (ideally) $\mathcal{C}(\Phi_{\mathbf{D}}(\mathbf{x})) = \ell_{\mathbf{x}}$ for any sample $\mathbf{x} \in \mathbb{R}^p$ that belongs to class $\ell_{\mathbf{x}} \in \mathcal{L}$. Note that the composition $\mathcal{C} \circ \Phi_{\mathbf{D}} : \mathbb{R}^m \to \mathcal{L}$ in this case is a nonlinear classifier in the input space.

In order to solve this problem, we resort to the framework of discriminative dictionary learning in which the nonlinear map $\Phi_{\mathbf{D}}$ is induced by a dictionary $\mathbf{D} \in \mathbb{R}^{m \times p}$ according to (1.1). We motivate that framework by collecting the training samples $\{\mathbf{y}_i^j\}_{j=1}^{N_i}$ into a matrix $\mathbf{Y}_i \in \mathbb{R}^{m \times N_i}$ and writing $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_K]$. In addition, we

associate with each label ℓ_i^j a *label unit-vector* $\mathbf{h}_i^j = \mathbf{e}_{\ell_i^j} \in \mathbb{R}^L$, where $\mathbf{e}_{\ell_i^j}$ denotes the ℓ_i^j-th column of the $L \times L$ identity basis. Then, collecting the label vectors $\{\mathbf{h}_i^j\}_{j=1}^{N_i}$ into a matrix $\mathbf{H}_i \in \mathbb{R}^{L \times N_i}$ and writing $\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_K]$, the problem of joint learning of a dictionary $\mathbf{D} \in \mathbb{R}^{m \times p}$ and a linear classifier $\mathcal{C}$ can be posed in terms of the following optimization problem [8]:

$$(\mathbf{D}^*, \mathbf{W}^*, \mathbf{X}^*) = \underset{\mathbf{D},\mathbf{W},\mathbf{X} \in \mathcal{X}}{\arg\min} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \gamma \|\mathbf{H} - \mathbf{W}\mathbf{X}\|_F^2 + \beta \|\mathbf{W}\|_F^2, \tag{3.1}$$

where

$$\mathcal{X} \triangleq \{\mathbf{X}' \in \mathbb{R}^{p \times N}, \|\mathbf{x}'_j\|_0 \leq S : j \in [N]\}. \tag{3.2}$$

Here, $\mathbf{X} \in \mathbb{R}^{p \times N}$ denotes the *coefficient matrix*, $\mathbf{W} \in \mathbb{R}^{L \times p}$ denotes the classification matrix, and the final classification rule $\mathcal{C}$ is defined in terms of the matrix $\mathbf{W}$ as $\mathcal{C}(\Phi_{\mathbf{D}}(\mathbf{x})) = \arg\max_{\ell \in \mathcal{L}} |[\mathbf{W}\Phi_{\mathbf{D}}(\mathbf{x})]_\ell|$. Note that the regularization parameters γ and β in (3.1) control the discriminative power and the complexity of the classifier, respectively.

While (3.1) is a non-convex problem, [8] provides a solution to this problem under the rubric of *discriminative K-SVD* (D-KSVD). But the D-KSVD framework, which relies on the K-SVD algorithm of [3] for dictionary learning, assumes $(\mathbf{Y}, \mathbf{H})$ to be available at one location. In contrast, our goal is to collaboratively solve (3.1) at each individual site when the training data is split across K sites (see Fig. 3.1) and sites do now want to gather all the distributed data at a centralized location for supervised learning due to the massive size of these data or privacy concerns. Given the nature of this problem, we can in fact only learn K different dictionary–classifier pairs $(\widetilde{\mathbf{D}}_i, \widetilde{\mathbf{W}}_i)$, one pair at each site, but our goal is to ensure that the classification performances of these pairs remain close to each other.

3.2 Proposed Collaborative Framework

In this section, we present our approach to collaborative learning of $(\widetilde{\mathbf{D}}_i, \widetilde{\mathbf{W}}_i)$ at each individual site from distributed training data. We term our proposed approach *cloud D-KSVD*, which is based on the centralized D-KSVD solution to (3.1) proposed in [8].

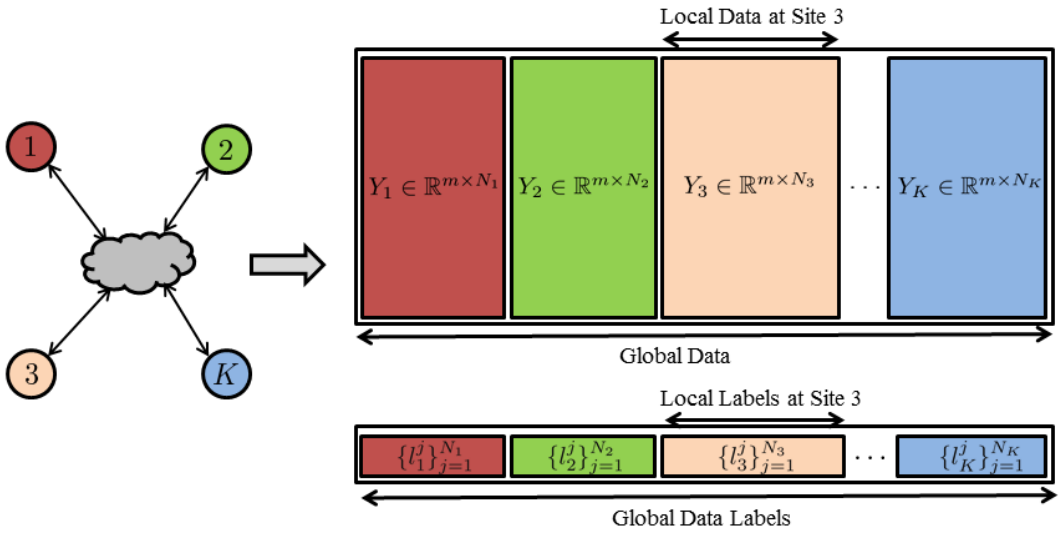


Figure 3.1: An illustration of the distribution of labeled training data across sites.

Before discussing cloud D-KSVD, however, we first provide a brief review of (centralized) D-KSVD for discriminative dictionary learning.

3.2.1 Centralized D-KSVD

The key to the D-KSVD solution of [8] is transformation of the discriminative dictionary learning problem (3.1) into the classical reconstructive dictionary learning problem [3]. Specifically, notice that (3.1) can be rewritten in the following form:

$$(\mathbf{D}^*, \mathbf{W}^*, \mathbf{X}^*) = \underset{\mathbf{D},\mathbf{W},\mathbf{X}\in\mathcal{X}}{\arg\min} \left\| \begin{pmatrix} \mathbf{Y} \\ \sqrt{\gamma}\mathbf{H} \end{pmatrix} - \begin{pmatrix} \mathbf{D} \\ \sqrt{\gamma}\mathbf{W} \end{pmatrix} \mathbf{X} \right\|_F^2 + \beta\|\mathbf{W}\|_F^2. \tag{3.3}$$

Next, define $\widehat{\mathbf{Y}} \in \mathbb{R}^{(m+L)\times N} \triangleq \begin{bmatrix} \mathbf{Y}^T & \sqrt{\gamma}\mathbf{H}^T \end{bmatrix}^T$ as "training data" and $\widehat{\mathbf{D}} \in \mathbb{R}^{(m+L)\times p} \triangleq \begin{bmatrix} \mathbf{D}^T & \sqrt{\gamma}\mathbf{W}^T \end{bmatrix}^T$ as "reconstructive dictionary". $\beta\|\mathbf{W}\|_F^2$ term in (3.3) can be removed as $\widehat{\mathbf{D}}$ is normalized column-wise. In other words, [8] promotes the use of the following optimization program as a surrogate for (3.3):

$$(\widehat{\mathbf{D}}^*, \mathbf{X}^*) = \underset{\widehat{\mathbf{D}},\mathbf{X}\in\mathcal{X}}{\arg\min} \|\widehat{\mathbf{Y}} - \widehat{\mathbf{D}}\mathbf{X}\|_F^2. \tag{3.4}$$

Training Algorithm

The formulation in (3.4) reduces the problem of learning $(\mathbf{D}, \mathbf{W})$ from the training data to that of learning a reconstructive dictionary $\widehat{\mathbf{D}}$ from $\widehat{\mathbf{Y}}$. In the D-KSVD formulation, (3.4) is solved using the K-SVD dictionary learning algorithm [3]. This involves initialization with some $\widehat{\mathbf{D}}^{(0)}$, followed by an alternate-minimization procedure that alternates between solving (3.4) first for $\mathbf{X}$ by fixing $\widehat{\mathbf{D}}$ and then for $\widehat{\mathbf{D}}$ by fixing $\mathbf{X}$. Specifically, assuming K-SVD has started iteration $t > 0$, it estimates $\mathbf{X}^{(t)}$ by carrying out *sparse coding* as follows:

$$\mathbf{X}^{(t)} = \arg\min_{\mathbf{X} \in \mathcal{X}} \|\widehat{\mathbf{Y}} - \widehat{\mathbf{D}}^{(t-1)}\mathbf{X}\|_F^2. \tag{3.5}$$

Note that (3.5) can be efficiently solved using a number of greedy or optimization-based algorithms [3].

Next, K-SVD estimates $\widehat{\mathbf{D}}^{(t)}$ by carrying out *dictionary update* as:

$$\widehat{\mathbf{D}}^{(t)} = \arg\min_{\mathbf{D}} \|\widehat{\mathbf{Y}} - \mathbf{D}\mathbf{X}^{(t-1)}\|_F^2. \tag{3.6}$$

The main novelty of K-SVD lies in the manner it efficiently solves (3.6). To this end, K-SVD fixes all but the k-th column $\widehat{\mathbf{d}}_k^{(t)}$, $k \in [p]$, of $\widehat{\mathbf{D}}^{(t)}$ and then (dropping the iteration count for ease of notation) defines the representation error matrix $\mathbf{E}_k = \widehat{\mathbf{Y}} - \sum_{j \neq k} \widehat{\mathbf{d}}_j \mathbf{x}_T^j$, where $\mathbf{x}_T^j$ denotes the j-th row of $\mathbf{X}^{(t)}$. Next, it obtains a column submatrix $\mathbf{E}_k^R$ of the matrix $\mathbf{E}_k$ by retaining those columns of $\mathbf{E}_k$ whose indices match the indices of the samples in $\widehat{\mathbf{Y}}$ that utilize $\widehat{\mathbf{d}}_k^{(t)}$. For this purpose, $\boldsymbol{\omega}_k$ is defined as

$$\boldsymbol{\omega}_k = \{i | i \in [p], \mathbf{x}_T^k(i) \neq 0\} \tag{3.7}$$

and $\boldsymbol{\Omega}_k$ is defined to be a matrix of size $N \times |\boldsymbol{\omega}_k|$ with ones on the $(\boldsymbol{\omega}_k(i), i)$ entries and zeros elsewhere. Then, $\mathbf{x}_R^k = \mathbf{x}_T^k \boldsymbol{\Omega}_k$ denotes the row vector consisting of only non-zero entries of $\mathbf{x}_T^k$ and $\mathbf{Y}_k^R = \mathbf{Y}\boldsymbol{\Omega}_k$ denotes a matrix consisting of samples that utilize column $\mathbf{d}_k$. Similarly, $\mathbf{E}_k^R = \mathbf{E}_k\boldsymbol{\Omega}_k$ denotes the error columns corresponding to $\mathbf{Y}_k^R$. It then attempts to minimize $\|\mathbf{E}_k^R - \mathbf{d}_k\mathbf{x}_R^k\|_F^2$ using SVD. It updates $\widehat{\mathbf{d}}_k^{(t)}$ by setting it equal to the dominant left singular vector of $\mathbf{E}_k^R$. In addition, it is advocated in [3] to simultaneously update the k-th row of $\mathbf{X}^{(t)}$ at this point by setting its nonzero entries

equal to $\sigma_1 \mathbf{v}_1^T$, where σ_1 and $\mathbf{v}_1$ denote the largest singular value and right singular vector of $\mathbf{E}_k^R$, respectively.

Classification Algorithm

Since K-SVD is guaranteed to converge under appropriate conditions [3], the D-KSVD algorithm obtains $\widehat{\mathbf{D}}$ from (3.4). The next challenge then becomes splitting $\widehat{\mathbf{D}} = \begin{bmatrix}\mathbf{D}^T & \sqrt{\gamma}\mathbf{W}^T\end{bmatrix}^T$ into a desired discriminative dictionary $\widetilde{\mathbf{D}}$ and a classification matrix $\widetilde{\mathbf{W}}$. One of the main contributions of [8] is establishing this relationship between the desired $(\widetilde{\mathbf{D}}, \widetilde{\mathbf{W}})$ and the $(\mathbf{D}, \mathbf{W})$ learned using (3.4). Specifically, [8] shows that

$$\widetilde{\mathbf{D}} = \begin{bmatrix}\frac{\mathbf{d}_1}{\|\mathbf{d}_1\|_2} & \frac{\mathbf{d}_2}{\|\mathbf{d}_2\|_2} & \cdots & \frac{\mathbf{d}_p}{\|\mathbf{d}_p\|_2}\end{bmatrix}, \text{ and} \tag{3.8}$$

$$\widetilde{\mathbf{W}} = \begin{bmatrix}\frac{\mathbf{w}_1}{\|\mathbf{d}_1\|_2} & \frac{\mathbf{w}_2}{\|\mathbf{d}_2\|_2} & \cdots & \frac{\mathbf{w}_p}{\|\mathbf{d}_p\|_2}\end{bmatrix}. \tag{3.9}$$

Once the pair $(\widetilde{\mathbf{D}}, \widetilde{\mathbf{W}})$ is obtained, the classification proceeds as follows. Given a test sample $\widetilde{\mathbf{y}} \in \mathbb{R}^n$ that belongs to one of the L classes in $\mathcal{L}$, we first obtain

$$\widetilde{\mathbf{x}} = \Phi_{\widetilde{\mathbf{D}}}(\widetilde{\mathbf{y}}) = \arg\min_{\mathbf{x}\in\mathcal{X}} \|\widetilde{\mathbf{y}} - \widetilde{\mathbf{D}}\mathbf{x}\|_2^2. \tag{3.10}$$

Next, we define $\widetilde{\mathbf{h}} = \widetilde{\mathbf{W}}\widetilde{\mathbf{x}}$ and then use the classification rule $\mathcal{C}(\Phi_{\widetilde{\mathbf{D}}}(\widetilde{\mathbf{y}})) = \arg\max_{\ell\in\mathcal{L}} |[\widetilde{\mathbf{h}}]_\ell|$, where $[\widetilde{\mathbf{h}}]_\ell$ is the ℓ-th entry of $\widetilde{\mathbf{h}}$.

3.2.2 Cloud D-KSVD

We are now ready to discuss our proposed collaborative framework for discriminative dictionary learning. Similar to D-KSVD, we are interested in solving (3.4) for $\mathbf{D}$ at each site. But the major difference is that $\widehat{\mathbf{Y}} = [\widehat{\mathbf{Y}}_1, \widehat{\mathbf{Y}}_2, \ldots, \widehat{\mathbf{Y}}_K]$ is now distributed across K sites, where $\widehat{\mathbf{Y}}_i = \begin{bmatrix}\mathbf{Y}_i^T & \sqrt{\gamma}\mathbf{H}_i^T\end{bmatrix}^T$.

Initialization

Unlike D-KSVD, initialization of $\widehat{\mathbf{D}}^{(0)}$ in (3.4) is also a function of the training data at individual sites. In cloud D-KSVD, we proceed with the initialization of the dictionary $\widehat{\mathbf{D}}_i^{(0)}$ locally at the i-th site as follows. First, we initialize a dictionary $\mathbf{D}_i^{(0)} \in \mathbb{R}^{m\times p}$

and carry out local sparse coding using $\mathbf{D}_i^{(0)}$, i.e.,

$$\mathbf{X}_i = \underset{\mathbf{X}\in\mathcal{X}_i}{\arg\min} \|\mathbf{Y}_i - \mathbf{D}_i^{(0)}\mathbf{X}\|_F^2, \tag{3.11}$$

where

$$\mathcal{X}_i \triangleq \{\mathbf{X}' \in \mathbb{R}^{p\times N_i}, \|\mathbf{x}'_j\|_0 \leq S : j \in [N_i]\}. \tag{3.12}$$

Next, we initialize a local classifier matrix $\mathbf{W}_i^{(0)}$ by solving

$$\mathbf{W}_i^{(0)} = \underset{\mathbf{W}}{\arg\min} \|\mathbf{H}_i - \mathbf{W}\mathbf{X}_i\|_F^2 + \beta\|\mathbf{W}\|_F^2. \tag{3.13}$$

Note that (3.13) is simply a multivariate ridge regression problem, with the closed-form solution given by

$$\mathbf{W}_i^{(0)} = (\mathbf{X}_i\mathbf{X}_i^T + \beta\mathbf{I})^{-1}\mathbf{X}_i\mathbf{H}_i^T. \tag{3.14}$$

Finally, we set the initial dictionary at the i-th site, $i \in \mathcal{V}$, as follows: $\widehat{\mathbf{D}}_i^{(0)} = \begin{bmatrix}\mathbf{D}_i^{(0)^T} & \sqrt{\gamma}\mathbf{W}_i^{(0)^T}\end{bmatrix}^T$.

Training Algorithm

After initialization, each site $i \in \mathcal{V}$ has access to $\widehat{\mathbf{D}}_i^{(0)}$ that is obtained using local data only. Our next goal is to solve (3.4) at each site for $\widehat{\mathbf{D}}_i \in \mathbb{R}^{(m+L)\times p}$ by relying on a collaborative variant of K-SVD that alternates between solving (3.4) first for (global) $\mathbf{X}$ by fixing $\widehat{\mathbf{D}}_i$ at each site and then for $\widehat{\mathbf{D}}_i$ by fixing $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_K]$, which will always be partitioned across the K sites. In a recent work [29], it is proposed such a collaborative variant using the moniker of *cloud K-SVD*. Specifically, assuming cloud K-SVD has started iteration $t > 0$ in the network, each site only updates the sparse representation of its local $\widehat{\mathbf{Y}}_i$ through sparse coding as follows:

$$\mathbf{X}_i^{(t)} = \underset{\mathbf{X}\in\mathcal{X}_i}{\arg\min} \|\widehat{\mathbf{Y}}_i - \widehat{\mathbf{D}}_i^{(t-1)}\mathbf{X}\|_F^2. \tag{3.15}$$

Next, sites focus on collaboratively updating their individual dictionary estimates $\{\widehat{\mathbf{D}}_i^{(t)}\}_{i\in\mathcal{V}}$. In this regard, cloud K-SVD takes its cue from K-SVD and fixes all but the k-th column $\widehat{\mathbf{d}}_{i,k}^{(t)}$ of $\widehat{\mathbf{D}}_i^{(t)}$ at each site. The next challenge then is defining the *global, reduced* representation error matrix $\mathbf{E}_k^R$ (we are once again dropping the iteration count

for ease of notation), since there are K different versions of dictionaries in the network. In order to address this challenge, cloud K-SVD first defines *local* representation error matrices $\mathbf{E}_{i,k} = \widehat{\mathbf{Y}}_i - \sum_{j \neq k} \widehat{\mathbf{d}}_{i,j} \mathbf{x}_{i,T}^j$, where $\mathbf{x}_{i,T}^j$ denotes the j-th row of $\mathbf{X}_i^{(t)}$. It then obtains a submatrix $\mathbf{E}_{i,k}^R$ of $\mathbf{E}_{i,k}$ by retaining the columns of $\mathbf{E}_{i,k}$ whose indices match the indices of the samples in $\widehat{\mathbf{Y}}_i$ that utilize $\widehat{\mathbf{d}}_{i,k}^{(t)}$. Finally, it defines the global, reduced representation error matrix as $\mathbf{E}_k^R = \left[\mathbf{E}_{1,k}^R, \mathbf{E}_{2,k}^R, \ldots, \mathbf{E}_{K,k}^R\right]$, which is distributed across the network. Cloud K-SVD then advocates to update $\widehat{\mathbf{d}}_{i,k}^{(t)}$ by setting it equal to the dominant left singular vector $\mathbf{u}_1$ of $\mathbf{E}_k^R$. Note that $\mathbf{u}_1$ is also equal to the dominant eigenvector of $\mathbf{M} = \mathbf{E}_k^R \mathbf{E}_k^{R^T} = \sum_{i \in \mathcal{V}} \mathbf{M}_i$, where $\mathbf{M}_i$ denotes $\mathbf{E}_{i,k}^R \mathbf{E}_{i,k}^{R^T}$. One of the main novelties of cloud K-SVD in this regard is formulation of a collaborative variant of the classical power method [59] for estimation of the dominant eigenvector of $\mathbf{M}$. This variant, which is described and rigorously analyzed in [29], relies on a finite number of iterations of distributed consensus averaging [60]. While more details of this part of cloud K-SVD can be found in [29], including a discussion of the doubly-stochastic mixing matrix needed for distributed consensus, the end result is that each site obtains an updated $\widehat{\mathbf{d}}_{i,k}^{(t)}$ that can come arbitrarily close to $\mathbf{u}_1$. Finally, cloud K-SVD also simultaneously updates the k-th row of $\mathbf{X}_i^{(t)}$ at this point by setting its nonzero entries equal to $\widehat{\mathbf{d}}_{i,k}^T \mathbf{E}_{i,k}^R$.

Classification Algorithm

The classification algorithm in cloud D-KSVD is identical to that in D-KSVD. Specifically, each site at this point obtains a dictionary $\widehat{\mathbf{D}}_i = \left[\mathbf{D}_i^T \quad \sqrt{\gamma}\mathbf{W}_i^T\right]^T$, which is then transformed into the final pair $(\widetilde{\mathbf{D}}_i, \widetilde{\mathbf{W}}_i)$ according to (3.8) and (3.9). Using this pair, each site can then individually classify any test sample $\widetilde{\mathbf{y}} \in \mathbb{R}^m$ according to the procedure described in Sec. 3.2.1.

3.3 Numerical Results

In this section, we demonstrate the effectiveness of cloud D-KSVD. We use the MNIST database [30], which consists of 70,000 28×28 pixel images of handwritten digits. Due

to the large number of samples in the MNIST database, dictionary learning yields better representations of samples compared to other learning techniques such as SVD or principle component analysis [61], as the data is represented by union of subspaces in this method.

For simplicity, each image is down-sampled to have only 256 features. We work on digits 0 to 4 in experiments ($L = 5$) and we consider a total of 10 sites. We perform 5-fold cross validation on the database by treating $\frac{1}{5}$ of the data as test data and the rest as training data in each fold. In the first set of experiments, we divide the training data uniformly, between the 10 sites. We train dictionaries using centralized D-KSVD (assuming all data is available at a single location), cloud D-KSVD, local D-KSVD (assuming each site performs training on local training data only) and cloud K-SVD (sites collaboratively learn purely representative dictionaries, one for each class). We also train a linear SVM for the centralized data for comparison with cloud D-KSVD.

To initialize the discriminative dictionaries, we first perform 10 iterations of K-SVD for the centralized and local setting and cloud K-SVD in the distributed setting. We then initial the classifiers according to (3.14) using these initial dictionaries. Then, we perform 50 iterations of D-KSVD for centralized and local setting and cloud D-KSVD for distributed setting. The parameters selected in these experiments correspond to a sparsity constraint of $S = 10$, $\gamma = 0.83$ and $p = 500$ number of dictionary columns (100 columns for each class).

For the representative dictionary, we train a separate dictionary for each data label by performing 60 iterations of cloud K-SVD. We set $S = 10$ and $p = 100$ for each dictionary (total of 500 atoms for 5 dictionaries). To classify a test data sample, the coefficient vector of the test sample is obtained for each dictionary using sparse coding. The assigned class to the sample is the index of the dictionary that best represents the sample (has the least representation error).

The test data classification results are shown in Fig. 3.2(a) where the sites' average classification error is plotted along with the worst case and best case error for each label for cloud D-KSVD, local D-KSVD and cloud K-SVD. The results demonstrate that cloud D-KSVD outperforms local D-KSVD and has a performance close to the

centralized D-KSVD and centralized linear SVM. Also, the classification performance of various sites is approximately identical when using cloud D-KSVD due to the fact that they are collaborating with one another. Note that non-linear SVM will likely outperform linear SVM, but we do not make the comparison with non-linear SVM here as our parameters are not optimally chosen. Finally, observing the classification error of cloud D-KSVD and cloud K-SVD, it is evident that cloud D-KSVD outperforms cloud K-SVD for all the class labels.

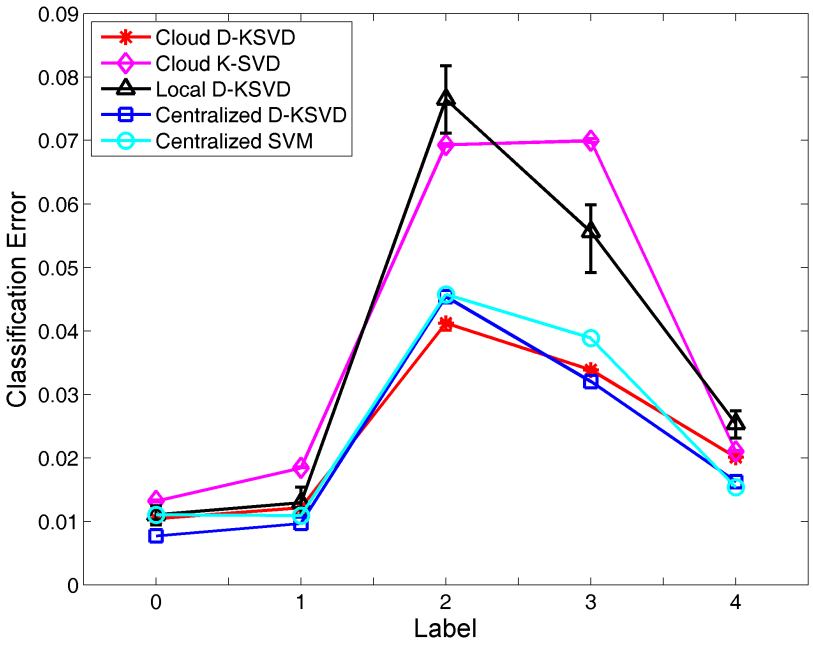
In the second set of experiments, we consider the case of the sites not having the same number of training data. In real world applications, some sites may have access to a smaller number of training data and there may be class imbalance in some sites (different class sizes). We consider that 80% of the labeled data is distributed among half of the sites, while the other 20% is distributed among the other half of the sites. The chosen parameters are similar to the previous simulations. The classification errors for this case are plotted in Fig. 3.2(b). It is apparent that distributed learning of the dictionary and classifier has a great advantage over training based on locally available data for sites with a smaller number of training data.

In the case of balanced data across sites, the normalized distance of the dictionary learned by centralized D-KSVD, $\widehat{\mathbf{D}}_C$, and the one learned by cloud D-KSVD at site i, $\widehat{\mathbf{D}}_{D,i}$, as a function of the number of dictionary learning iterations, is defined as

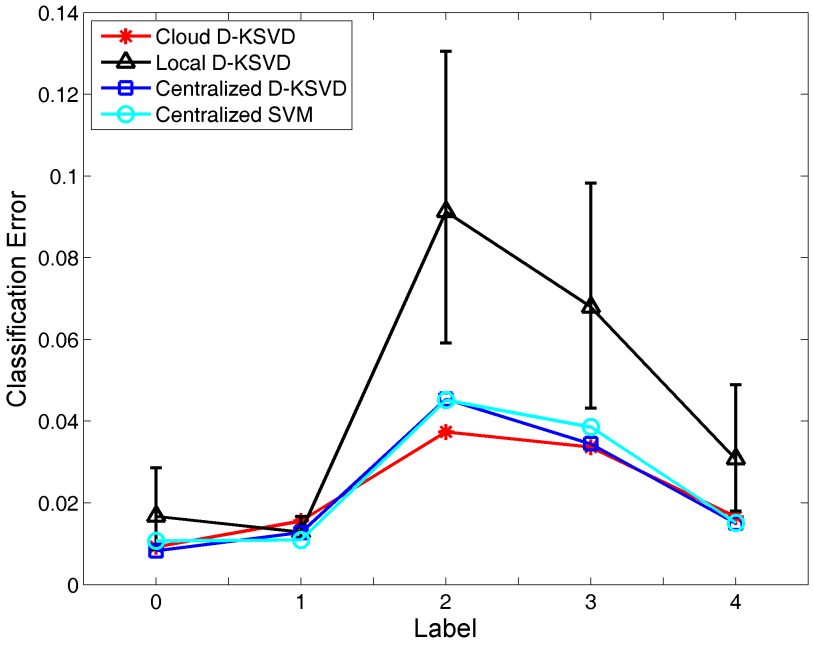
$$d^{(t)} = \frac{1}{p}\left\|\widehat{\mathbf{D}}_C^{(t)} - \widehat{\mathbf{D}}_{D,i}^{(t)}\right\|_F^2,\ t = [50],\ i \in \mathcal{V}. \tag{3.16}$$

The dictionary $\widehat{\mathbf{D}}_C$ is equivalent to $p!2^p$ other dictionaries that consist of column-wise permuted versions of $\widehat{\mathbf{D}}_C$ with all possible sign flips for atoms. Due to different initialization of dictionaries, the normalized distance between $\widehat{\mathbf{D}}_C$ and $\widehat{\mathbf{D}}_{D,i}$ is an upper bound for the normalized distance between the equivalent class of $\widehat{\mathbf{D}}_C$ and $\widehat{\mathbf{D}}_{D,i}$.

Fig. 3.3 plots this normalized distance averaged over 10 sites along with the least and most normalized distance as a function of the number of iterations. It is evident that the average normalized distance does not vary significantly across different iterations and sites obtain similar dictionaries.



(a) Balanced distributed training data



(b) Distributed training data with class imbalance

Figure 3.2: Performance summary of cloud D-KSVD. (a) and (b) compare the classification performance of cloud D-KSVD with that of centralized and local D-KSVD, centralized linear SVM, and cloud K-SVD. The results for cloud D-KSVD, local D-KSVD and cloud K-SVD are displayed using bars to highlight the best, worst, and average error across sites.

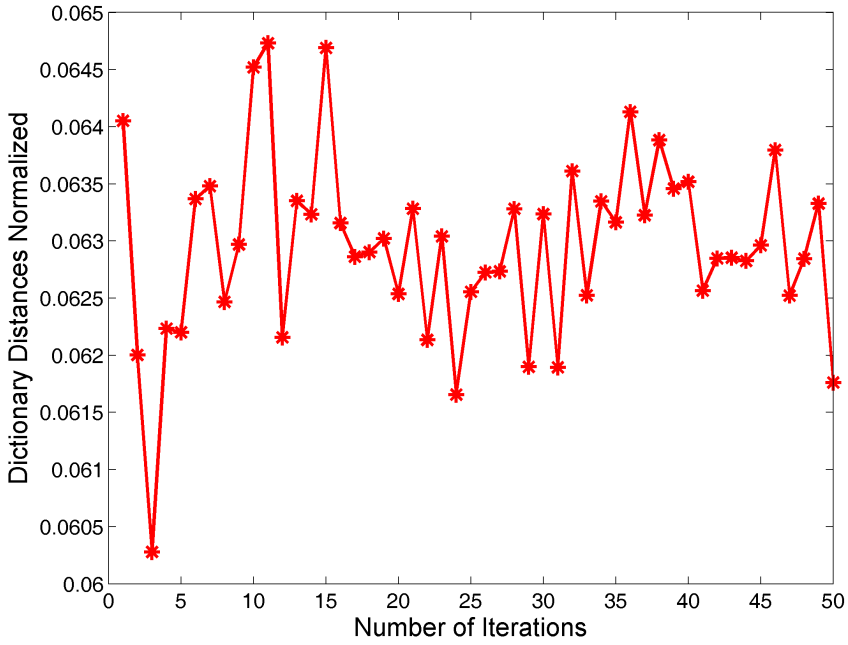


Figure 3.3: The average normalized distance along with the least and most normalized distance between the dictionaries obtained using cloud D-KSVD and centralized D-KSVD as a function of the number of dictionary learning iterations.

Chapter 4

Conclusion and Future Work

In the first part of this thesis, we focused on the problem of dictionary identifiability. Considering the maximal response criterion, we obtained conditions on the underlying dictionary and the coefficient vector to ensure reliable recovery of the true dictionary. Future directions of this work include extension of this proof technique to other dictionary learning objective functions, developing minimax lower bounds for dictionary learning, and analyzing dictionary identifiability for structured signals such as tensors.

In the second part, we developed a collaborative framework for learning a nonlinear classifier from distributed data. Our framework corresponded to joint learning of a dictionary and a linear classifier by leveraging recent results on discriminative and collaborative dictionary learning. In order to verify the effectiveness of our approach, we carried out numerical experiments that showed that the performance of our framework comes very close to that of centralized methods. Further aspects of this work that can be further explored are providing rigorous analysis for the convergence of local dictionaries and linear classifiers and their deviations from centralized counterparts and replacing D-KSVD part of our collaborative framework with some of the other (centralized) discriminative dictionary learning approaches for enhanced efficiency and performance.

Bibliography

[1] The magical world of hadoop, outsystems and big data - nextstep 2014. `http://www.slideshare.net/OutSystems`. Published: 06-11-2014.

[2] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. Lee, and T. J. Sejnowski. Dictionary learning algorithms for sparse representation. *Neural computation*, 15(2):349–396, 2003.

[3] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.

[4] D. L. Donoho, M. Elad, and V. N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1):6–18, 2006.

[5] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

[6] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of 24th international conference on Machine learning*, pages 759–766. ACM, 2007.

[7] J. Mairal, J. Ponce, G. Sapiro, A. Zisserman, and F. Bach. Supervised dictionary learning. In *Proceedings of Advances in Neural Information Processing Systems*, pages 1033–1040, 2009.

[8] Q. Zhang and B. Li. Discriminative K-SVD for dictionary learning in face recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010)*, pages 2691–2698, 2010.

[9] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

[10] K. Schnass. Local identification of overcomplete dictionaries. *arXiv preprint arXiv:1401.6354*, 2014.

[11] Karin Schnass. Convergence radius and sample complexity of itkm algorithms for dictionary learning. *arXiv preprint arXiv:1503.07027*, 2015.

[12] W. U. Bajwa, R. Calderbank, and S. Jafarpour. Why gabor frames? two fundamental measures of coherence and their role in model selection. *Journal of Communications and Networks*, 12(4):289–307, 2010.

[13] W. U. Bajwa, R. Calderbank, and D. G. Mixon. Two are better than one: Fundamental parameters of frame coherence. *Applied and Computational Harmonic Analysis*, 33(1):58–78, July 2012.

[14] C. McDiarmid. On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188, 1989.

[15] K. Azuma. Weighted sums of certain dependent random variables. *Tohoku Math. J.*, 19(3):357–367, 1967.

[16] R. Gribonval and K. Schnass. Dictionary identification-sparse matrix-factorisation via ℓ_1-minimisation. *IEEE Transactions on Information Theory*, 56(7):3523–3539, 2010.

[17] D. A. Spielman, H. Wang, and J. Wright. Exact recovery of sparsely-used dictionaries. *arXiv preprint arXiv:1206.5882*, 2012.

[18] P. Georgiev, F.J. Theis, and A. Cichocki. Sparse component analysis and blind source separation of underdetermined mixtures. *IEEE Transactions on Neural Networks*, 16(4):992–996, 2005.

[19] M. Aharon, M. Elad, and A. M. Bruckstein. On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Linear algebra and its applications*, 416(1):48–67, 2006.

[20] A. Agarwal, A. Anandkumar, P. Jain, and P. Netrapalli. Learning sparsely used overcomplete dictionaries via alternating minimization. *arXiv preprint arXiv:1310.7991*, 2013.

[21] A. Agarwal, A. Anandkumar, and P. Netrapalli. Exact recovery of sparsely used overcomplete dictionaries. *arXiv preprint arXiv:1309.1952v1*, 2013.

[22] S. Arora, R. Ge, and A. Moitra. New algorithms for learning incoherent and overcomplete dictionaries. In *Proceedings of 27th Conference on Learning Theory*, pages 779–806, 2014.

[23] Q. Geng, H. Wang, and J. Wright. On the local correctness of ℓ_1 minimization for dictionary learning. *arXiv preprint arXiv: 1101: 5672*, 2011.

[24] K. Schnass. On the identifiability of overcomplete dictionaries via the minimisation principle underlying K-SVD. *Applied and Computational Harmonic Analysis*, 37 (3):464–491, 2014.

[25] R. Gribonval, R. Jenatton, and F. Bach. Sparse and spurious: dictionary learning with noise and outliers. *IEEE Transactions on Information Theory*, 61(11):6298–6319, 2015.

[26] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research*, 11:19–60, 2010.

[27] D. Michie, D. J. Spiegelhalter, and C. C. Taylor. Machine learning, neural and statistical classification. *Ellis Horwood Series in Artificial Intelligence, Ellis Horwood*, 1994.

[28] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*, volume 2. Springer, 2009.

[29] H. Raja and W.U. Bajwa. Cloud K-SVD: Computing data-adaptive representations in the cloud. In *Proceedings of 51st Annual Allerton Conference on Communication, Control, and Computing*, pages 1474–1481, 2013.

[30] Y. LeCun and C. Cortes. The MNIST database of handwritten digits. `http://yann.lecun.com/exdb/mnist/`, 1998.

[31] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. In *Proceedings of IEEE Conference Computer Vision and Pattern Recognition (CVPR 2008)*, pages 1–8, 2008.

[32] F. Rodriguez and G. Sapiro. Sparse representations for image classification: Learning discriminative and reconstructive non-parametric dictionaries. Technical report, DTIC Document, 2008. `http://www.dsp.ece.rice.edu/cs/`.

[33] D. Pham and S. Venkatesh. Joint learning and dictionary construction for pattern recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVRP 2008)*, pages 1–8, 2008.

[34] Z. Jiang, Z. Lin, and L. S. Davis. Learning a discriminative dictionary for sparse coding via label consistent K-SVD. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, pages 1697–1704, 2011.

[35] J. Mairal, F. Bach, and J. Ponce. Task-driven dictionary learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):791–804, 2012.

[36] A. D'Costa and A. M. Sayeed. Data versus decision fusion for classification in sensor networks. In *Proceedings of International Conference on Information fusion*, 2003.

[37] A. D'Costa, V. Ramachandran, and A. M. Sayeed. Distributed classification of Gaussian space-time sources in wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1026–1036, 2004.

[38] J. H. Kotecha, V. Ramachandran, and A. M. Sayeed. Distributed multitarget classification in wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):703–713, April 2005.

[39] E. Kokiopoulou and P. Frossard. Distributed SVM applied to image classification. In *Proceedings of IEEE International Conference on Multimedia and Expo*, pages 1753–1756, July 2006.

[40] E. Kokiopoulou and P. Frossard. Distributed classification of multiple observation sets by consensus. *IEEE Transactions on Signal Processing*, 59(1):104–114, Jan 2011.

[41] H. P. Graf, E. Cosatto, L. Bottou, I. Dourdanovic, and V. Vapnik. Parallel support vector machines: The cascade SVM. In *Advances in neural information processing systems*, pages 521–528, 2004.

[42] T. Do and F. Poulet. Classifying one billion data with a new distributed SVM algorithm. In *Proceedings of International Conference on Research, Innovation and Vision for the Future*, pages 59–66, 2006.

[43] K. Zhu, H. Wang, H. Bai, J. Li, Z. Qiu, H. Cui, and E. Y. Chang. Parallelizing support vector machines on distributed computers. In *Advances in Neural Information Processing Systems*, pages 257–264, 2008.

[44] D. Mahajan, S. S. Keerthi, and S. Sundararajan. A distributed algorithm for training nonlinear kernel machines. *arXiv preprint arXiv:1405.4543*, 2014.

[45] C. Guestrin, P. Bodik, R. Thibaux, M. Paskin, and S. Madden. Distributed regression: An efficient framework for modeling sensor network data. In *Proceedings of 3rd International Symposium on Information Processing in Sensor Networks*, pages 1–10, 2004.

[46] J. B. Predd, S. R. Kulkarni, and H. V. Poor. A collaborative training algorithm for distributed learning. *IEEE Transactions on Information Theory*, 55(4):1856–1871, April 2009.

[47] A. Navia-Vazquez, D. Gutierrez-Gonzalez, E. Parrado-Hernandez, and J.J. Navarro-Abellan. Distributed support vector machines. *IEEE Transactions on Neural Networks*, 17(4):1091–1097, July 2006.

[48] K. Flouri, B. Beferull-Lozano, and P. Tsakalides. Training a SVM-based classifier in distributed sensor networks. In *Proceedings of 14th European Signal Processing Conference*, pages 1–5, 2006.

[49] K. Flouri, B. Beferull-Lozano, and P. Tsakalides. Distributed consensus algorithms for SVM training in wireless sensor networks. In *Proceedings of 16th European Signal Processing Conference*, pages 25–29, 2008.

[50] K. Flouri, B. Beferull-Lozano, and P. Tsakalides. Optimal gossip algorithm for distributed consensus SVM training in wireless sensor networks. In *Proceedings of 16th International Conference on Digital Signal Processing*, pages 1–6, July 2009.

[51] P. A. Forero, A. Cano, and G. B. Giannakis. Consensus-based distributed linear support vector machines. In *Proceedings of 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 35–46, 2010.

[52] P. A. Forero, A. Cano, and G. B. Giannakis. Consensus-based distributed support vector machines. *Journal of Machine Learning Research*, 11:1663–1707, August 2010.

[53] M. R. Guarracino, A. Irpino, N. Radziukyniene, and R. Verde. Supervised classification of distributed data streams for smart grids. *Energy Systems*, 3(1):95–108, 2012.

[54] S. Lee and A. Nedić. DrSVM: Distributed random projection algorithms for SVMs. In *Proceedings of IEEE 51st Annual Conference on Decision and Control (CDC)*, pages 5286–5291, Dec 2012.

[55] S. Lee and A. Nedić. Distributed random projection algorithm for convex optimization. *IEEE Journal of Selected Topics in Signal Processing*, 7(2):221–229, April 2013.

[56] Y. Lu, V. Roychowdhury, and L. Vandenberghe. Distributed parallel support vector machines in strongly connected networks. *IEEE Transactions on Neural Networks*, 19(7):1167–1178, July 2008.

[57] L. R. Welch. Lower bounds on the maximum cross correlation of signals. *IEEE Transactions on Information Theory*, 20(3):397–399, 1974.

[58] R. Motwani and P. Raghavan. *Randomized algorithms*. Chapman & Hall/CRC, 2010.

[59] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. Baltimore MD: John Hopkins University Press, 2012.

[60] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

[61] Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.