# THEORETICAL RESULTS AND APPLICATIONS RELATED TO DIMENSION REDUCTION

A Thesis
Presented to
The Academic Faculty

by

Jie Chen

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology
December 2007

# THEORETICAL RESULTS AND APPLICATIONS RELATED TO DIMENSION REDUCTION

Approved by:

Professor Xiaoming Huo, Advisor
School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Professor Nicoleta Serban
School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Professor Jeff Wu
School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Professor Minqiang Li
College of Management
*Georgia Institute of Technology*

Professor Shijie Deng
School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Date Approved: 29 October 2007

# ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor, Professor Xiaoming Huo for his great help and support during my graduate studies. His patience and insight make it a pleasure to work with him. Thanks are also due to other members of my committee, Professor Jeff Wu, Shijie Deng, Nicoleta Serban and Minqiang Li. I am also very grateful to my parents, Jiqiu Chen and Wenhui Liu, for their love and support over these four and a half years. Finally, I most cordially thank my husband, Jin Liu, who has been a tremendous source of encouragement in the past several years. Without him, the life in the U.S. would have been more tough and pale.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# SUMMARY

To overcome the curse of dimensionality, dimension reduction is important and necessary for understanding the underlying phenomena in a variety of fields. Dimension reduction is the transformation of high-dimensional data into a meaningful representation in the low-dimensional space. It can be further classified into feature selection and feature extraction. In this thesis, which is composed of four projects, the first two focus on feature selection, and the last two concentrate on feature extraction.

The content of the thesis is as follows. The first project presents several efficient methods for the sparse representation of a multiple measurement vector (MMV); some theoretical properties of the algorithms are also discussed. The second project introduces the NP-hardness problem for penalized likelihood estimators, including penalized least squares estimators, penalized least absolute deviation regression and penalized support vector machines. The third project focuses on the application of manifold learning in the analysis and prediction of 24-hour electricity price curves. The last project proposes a new hessian regularized nonlinear time-series model for prediction in time series.

Main contributions in this thesis are the following:

- Several new theorems regarding the sparse representation in MMV are proved. Their implication in computation is demonstrated through simulations.
- NP-hardness regarding penalized likelihood estimators is new.
- The application of manifold learning approach to electricity price prediction is, to the best of our knowledge, the first time.

- A new hessian regularized nonlinear time-series model is proposed, and its advantages over other nonlinear time-series models are illustrated.

# CHAPTER I

# INTRODUCTION

## *1.1 Motivation*

An information overload in most sciences has been caused by the advances in data collection and storage capabilities during the past decades. Researchers working in a variety of domains, e.g., engineering, astronomy, biology, remote sensing, economics, and consumer transactions, face larger and larger amount of observations and simulations on a daily basis. Such data sets, in contrast with traditional smaller data sets studied extensively in the past, present new challenges in data analysis. Traditional statistical methods break down partly because of the increase in the number of observations, but mostly due to the increase in the number of variables associated with each observation. The dimension of the data is the number of variables that are measured on each observation [44].

Therefore, it is useful and necessary to reduce the dimension of the data to a manageable size. Meanwhile, the original information should be kept as much as possible. After that, the reduced-dimensional data are feeded into the processing system.

The problem of dimension reduction can be more formally stated as the transformation of high-dimensional data into a meaningful representation of reduced dimensionality, under the assumption that the sample actually lies, at least approximately, on a manifold (nonlinear in general) of smaller dimensions than the original data space. The dimensionality of the representation in the smaller dimensional space is called the intrinsic dimensionality of the data. There are several good reviews and surveys on dimension reduction in the literature [61, 109, 10, 44].

Dimension reduction is important in many domains, e.g., classification, visualization, and compression of high-dimensional data, as the curse of dimensionality is an undesired property of high-dimensional space. The curse of the dimensionality refers to the fact that the sample size needed to estimate a function of several variables to a given degree of accuracy (i.e., to get a reasonably low-variance estimate) grows exponentially with the number of variables. A way to avoid the curse of the dimensionality is to reduce the input dimension of the function to be estimated [10].

In statistics, dimension reduction is the process of reducing the number of random variables under consideration, and can be divided into feature selection and feature extraction. Feature selection is the technique that is commonly used for selecting a subset of relevant features for building robust learning models. Feature extraction is applying a mapping of the high-dimensional space into a space of fewer dimensions. This means that the original feature space is transformed by applying a linear or nonlinear transformation. In this work, Chapter 2 and Chapter 3 belong to feature selection problems, while Chapter 4 and Chapter 5 present feature extraction problems.

## *1.2 Contributions*

The main contribution of this work is that we studied some important theoretical properties of the sparse representation problem and penalized likelihood estimators. We also explore the applications of dimension reduction in time series analysis and prediction. The contribution of each of the four projects in the thesis is as follows.

- In Chapter 2, the sparse representation of a multiple measurement vector (MMV) is studied. It is a relatively new problem in the sparse representation, and the theoretical analysis is lacking. In this chapter, some known results of SMV are generalized to MMV. Some of these new results take advantages of additional information in the formulation of MMV. Two computational efficient methods,

$\ell_1$-norm approach and orthogonal matching pursuit (OMP), have been proposed to replace the original $\ell_0$-norm problem. Several new theorems regarding finding the sparest representation in MMV are proved. Simulations show that the predictions made by the proved theorems tend to be very conservative; this is consistent with some recent advances in probabilistic analysis based on random matrix theory.

- Chapter 3 presents that for several existing types of penalty functions, the corresponding penalized least squares estimations, penalized least absolute deviation regressions and penalized support vector machines are NP-hard problems. Our NP-hardness results do not oppose the principle of penalized likelihood estimators. Instead, our results forewarn a misuse of penalized likelihood estimators: i.e., one should not attempt to find the *global* extremum(a) in numerical implementations. The correct way to utilize the penalized likelihood estimator is the following: starting with a consistent estimator (e.g., the maximum likelihood estimator), then modifying it via optimizing the penalized likelihood function locally.

- Chapter 4 proposes a novel nonparametric approach for modeling electricity price curves. Analysis on the intrinsic dimensionality of an electricity price curve is offered. The resulting analysis sheds light on identifying major factors governing the price curve dynamics. The forecast accuracy of our model compares favorably against that of the ARIMA type models in one-day ahead prediction, and is much better in prediction over longer horizons such as one week or one month.

- Chapter 5 introduces a new hessian regularized nonlinear time-series model for prediction in time series. The approach is especially powerful when the number of dependent variables is greater than three, which can not be handled by

natural cubic spline and thin plate spline. Moreover, our approach is nonlinear and nonparametric, and does not enforce any specific structure on the model. Compared to local polynomial regression models, which are pure local methods, the great advantage of our method is that the penalty term of hessian functional can also take into account of the global properties of the data. Both the theoretical and simulation results provide a strong verification and support of our model.

## *1.3 Outline of the Thesis*

The thesis is organized by four projects.

- Chapter 2 presents several efficient methods for the sparse representation of a multiple measurement vector (MMV). We consider the uniqueness under both an $\ell_0$-norm like criterion and an $\ell_1$-norm like criterion. The consequent equivalence between the $\ell_0$-norm approach and the $\ell_1$-norm approach indicates a computationally efficient way of finding the sparsest representation in a redundant dictionary. For greedy algorithms, it is proven that under certain conditions, orthogonal matching pursuit (OMP) can find the sparsest representation of an MMV with computational efficiency, just like in SMV. Simulations show that the predictions made by the proved theorems tend to be very conservative.

- Chapter 3 shows that with a class of penalty functions, numerical problems associated with the implementation of the penalized least squares estimators are equivalent to the exact cover by 3-sets problem, which belongs to a class of NP-hard problems. We then extend this NP-hardness result to the cases of penalized least absolute deviation regression and penalized support vector machines. We discuss the practical implication of our results. In particular, we emphasize that the oracle property of a penalized likelihood estimator requires a local extremum, instead of a global extremum. Hence the penalized likelihood

estimators are still favorable; however, one should not attempt to find their global extrema.

- Chapter 4 applies manifold-based dimension reduction to electricity price curve modeling. LLE is demonstrated to be an efficient method for extracting the intrinsic low-dimensional structure of electricity price curves. Using price data taken from the NYISO, we find that there exists a low-dimensional manifold representation of the day-ahead price curve in NYPP, and specifically, the dimension of the manifold is around four. The interpretation of each dimension and the cluster analysis in the low-dimensional space are given to analyze the main factors of the price curve dynamics. The sensitivity of the parameters is also analyzed. Numerical experiments show that our prediction preforms well for the short-term prediction, and our method also facilitates medium-term prediction, which is difficult, even infeasible for other methods.

- Chapter 5 introduces a numerical approach for prediction in time series, for which the underlying model is nonlinear and nonparametric. Comparing to other existing methods in nonlinear time series, our assumption on the underlying model is the most general: we only require that the Hessian of the underlying function is integrable. The main idea in our method is to minimize a functional, which is made by the classical residual sum of squares plus a penalty term which is an algorithmic parameter ($\lambda$) multiplying with the integrated Hessian of the underlying function. This approach adopts the philosophy of regularization. The integrated Hessian can serve as a smoothness measure of the underlying function. We develop a numerical approximation to the above functional, hence the methodology becomes applicable to real data. In simulations, we compare our method with other state-of-the-art algorithms in nonlinear time series, namely threshold autoregressive (TAR), additive autoregressive

(AAR), functional coefficient autoregressive (FAR), local polynomial regression, etc. The result is very satisfactory: if the underlying model is consistent with one of other models, the performance of our algorithm is comparable; if the underlying model is inconsistent with other models, our algorithm can outperform. We also provide some theoretical analysis on when our method performs well. A discussion on fast computation is given as well.

# CHAPTER II

# THEORETICAL RESULTS ON SPARSE REPRESENTATIONS OF MULTIPLE MEASUREMENT VECTORS

## *2.1 Introduction*

The problem of finding sparse representations of multiple measurement vectors (MMV) in a redundant dictionary was motivated by a neuro-magnetic inverse problem that arises in Magnetoencephalography (MEG) – a modality of imaging the possible activation regions in the brain. We refer to Cotter et al., [22, 92], and a historic paper [51] for more details and other potential applications. The problem of MMV can also be considered as how to achieve sparse representations for SMVs *simultaneously* [106, 107, 104]. In this chapter, we focus on the theoretical development of the MMV problem, instead of its applications.

Given a multiple measurement vector $B$ and a dictionary $A$, an MMV problem solves the system of equations,

$$AX = B,$$

where $A \in \mathbb{R}^{m\times n}, X \in \mathbb{R}^{n\times L}$, and $B \in \mathbb{R}^{m\times L}$. Each column of the matrix $A$ is associated with an *atom*. A set of all atoms is called a *dictionary* (see Mallat's book [80]), which is denoted by $\Omega$. A *sparse representation* means that matrix $X$ (or a vector, if one has an SMV: $L = 1$) has a small number of rows that contain nonzero entries. A mathematical definition of the *sparsity* of a matrix $X$ will be provided later.

A redundant dictionary simply means that $m < n$. Usually, we have $m \ll n$ and

$L < m$. As we mentioned earlier, when $L = 1$, we have the case of a single measurement vector (SMV). Matrices $X$ and $B$ can be rewritten as $X = [x^{(1)}, x^{(2)}, \ldots, x^{(L)}]$, $B = [b^{(1)}, b^{(2)}, \ldots, b^{(L)}]$, where $x^{(l)}$'s and $b^{(l)}$'s, $1 \leq l \leq L$, are column vectors. Evidently, the system of equations $AX = B$ can be rewritten as $Ax^{(l)} = b^{(l)}$, where $l = 1, \ldots, L$. For simplicity, we assume that the columns of $A$ have been normalized; hence all the diagonal entries of the Gram matrix $G = A^T A$ are equal to ones and all the off-diagonal entries are in the interval $[-1, 1]$.

In the case of SMV, there are abundant results on the sparsest representations in a redundant dictionary. We refer to [34, 33, 36, 43, 30, 53, 46]. The introduction of Donoho, Elad, and Temlyakov [31] gives a comprehensive depiction on many important applications. In MMV, we replace $x$ and $b$ by the upper-case letters, $X$ and $B$, emphasizing that they are *matrices* instead of *column vectors*.

In SMV, the sparsity of a representation is defined as the $\ell_0$ quasi-norm of the vector $x$, which is denoted by $\|x\|_0$. The quantity $\|x\|_0$ is simply the number of nonzero elements in the vector $x$. Without loss of accuracy, for simplicity, throughout this chapter, we will call the quantity $\|x\|_0$ an $\ell_0$-norm, instead of an $\ell_0$-quasi-norm; similarly, we will say an $\ell_0$-norm like criterion, instead of an $\ell_0$-quasi-norm like criterion. The sparsest representation in SMV is the solution to the following optimization problem:

$$\textbf{(Q0):} \quad \min \|x\|_0, \quad \text{subject to } Ax = b.$$

The above problem can be convexified as a minimizing-the-$\ell_1$-norm problem,

$$\textbf{(Q1):} \quad \min \|x\|_1, \quad \text{subject to } Ax = b,$$

where $\|x\|_1$ is the sum of the absolute values of the entries of vector $x$, i.e., for $x = [x_1, x_2, \ldots, x_n]^T$, we have $\|x\|_1 = \sum_{i=1}^{n} |x_i|$. Readers may compare the objective functions in **(Q0)** and **(Q1)**. Note that **(Q1)** can be solved via *linear programming*.

The problem **(Q0)** is essentially a combinatorial optimization problem, which in

general is difficult to solve. We hope that the solution to problem **(Q1)** is, in some situations, close enough to the solution to **(Q0)**. The equivalence of the solutions between **(Q0)** and **(Q1)** has been proved under various conditions, and the more recent work was done by many researchers including Donoho and Elad [30], Tropp [103], and Fuchs [46]. Evidently, the equivalence between the two solutions is very important in computing the sparsest representation in SMV. In this chapter, we extend the corresponding theorems from SMV to MMV.

Another way to obtain a sparse representation is through a greedy algorithm, e.g., orthogonal matching pursuit (OMP). It has been proved by Donoho, Elad, and Temlyakov [31] and Tropp [103] independently that under certain conditions, the OMP can find the sparsest representation of the signal. In this chapter, we extend this theory to MMV too.

In the present chapter, we consider a *noiseless* case: an SMV, $b$, or an MMV, $B$, is a linear combination of atoms without noise, i.e., $b = Ax$ or $B = AX$. It will be a different mathematical problem when additive noise is considered in the formulation. For *noisy* cases, we refer to [31, 103, 105] for results in SMV and [107, 104] for results in MMV.

In our generalization from SMV to MMV, it is shown that the generalization can be very broad: the inner vector norm can be any vector norm in a Euclidean space. Moreover in the case of minimizing-the-$\ell_0$-norm, less stringent requirements to guarantee uniqueness can be derived, compared to those for SMV.

The rest of the chapter is organized as follows. Section 2.2 describes the uniqueness of the solutions to the minimizing-the-$\ell_0$-norm problems. Section 2.3 describes the situations in which the solutions to the minimizing-the-$\ell_1$-norm problems are identical with the solutions to the minimizing-the-$\ell_0$-norm problems. Section 2.4 describes the property of the sparsest representations that are computed from a greedy algorithm – OMP. Conditions under which OMP gives the sparsest representations are given.

Section 2.5 describes some simulations, which indicate that the theoretical bounds are conservative. Section 2.6 gives the discussion on related works, possible extensions, and future research topics. Section 2.7 makes some concluding remarks.

## *2.2 Minimizing the $\ell_0$ Norm*

### 2.2.1 Formulation

We describe our formulation of MMV. The following quantity is the number of rows (in a matrix $X$) that contain nonzero entry(ies):

$$\mathcal{R}(X) = \|(m(x_i))_{n\times 1}\|_0 \,,$$

where $x_i \in \mathbb{R}^L$ is the transpose of the $i$th row of the matrix $X$, i.e., $X = [x_1, x_2, \cdots, x_n]^T$, $m(\cdot)$ is any vector norm in $\mathbb{R}^L$, and vector $(m(x_i))_{n\times 1}$ has the $i$th entry equal to $m(x_i), 1 \leq i \leq n$. Symbol $\mathcal{R}$ stands for a sparsity *rank*. A noiseless sparse representation problem in MMV can be written as

$$\textbf{(P0):} \quad \min \quad \mathcal{R}(X), \qquad \text{subject to } AX = B.$$

Readers can compare this with **(Q0)**. In fact, if $L = 1$, the above optimization problem becomes **(Q0)**.

In general, solving **(P0)** requires enumerating all the subsets of set $\{1, 2, \ldots, n\}$. The complexity of such a subset-search algorithm grows *exponentially* with the dictionary size $n$.

### 2.2.2 Uniqueness in $\ell_0$-norm Minimization

We restrict our attention to the case when the solution to **(P0)** is unique. It is provable that having sufficient sparsity is a sufficient condition for the solution (i.e. representation) to be the unique sparsest one. We give some conditions under which the solution to the problem **(P0)** is unique. This is a necessary preparation for a subsequent result, i.e., equivalence of solutions between the $\ell_0$-norm minimization problem and the $\ell_1$-norm minimization problem.

The following generalizes the result of Donoho and Elad [30] to MMV. We start with the concept of *Spark* [71].

**Definition 2.2.1 (Spark)** *Given a matrix $A$, the quantity Spark, which is denoted by $Spark(A)$ (or $\sigma$), is the smallest possible integer such that there exist $\sigma$ columns of matrix $A$ that are linearly dependent.*

In [30], $Spark(A)/2$ is a threshold of the sparsity: if the signal is made by less than $Spark(A)/2$ atoms, or in other words, if the signal is a linear combination of less than $Spark(A)/2$ columns of matrix $A$, then the solution to **(P0)** is exactly the atoms that are included in this linear combination. For MMV, with the above mentioned $\mathcal{R}(\cdot)$, we can draw the following conclusion. It is interesting that the result holds for any vector norm $m(\cdot)$.

**Theorem 2.2.2** *Matrix $X$ will be the unique solution of the problem* **(P0)**, *if $B = AX$ and*

$$\mathcal{R}(X) < Spark(A)/2.$$

Compared with the SMV cases, the above theorem has the same upper bound.

**Remark 2.2.3** *From the known results in SMV, Theorem 2.2.2 can be proved as a direct extension. The argument is as follows. If $\mathcal{R}(X) < Spark(A)/2$, obviously $\|X^{(j)}\|_0 < Spark(A)/2$, $1 \leq j \leq L$, where $X^{(j)}$ is the $j$th column of matrix $X$. Hence the solution to the following optimization problem*

$$\min \|Y^{(j)}\|_0, \ \textit{subject to: } AY^{(j)} = B^{(j)}, \ j = 1, 2, \ldots, L,$$

*where $Y \in \mathbb{R}^{n \times L}$ and $Y^{(j)}$ is the $j$th column of $Y$, should give exactly the $j$th column of matrix $X$; Recall $B^{(j)}$ is the $j$th column of matrix $B$. This renders the fact that $X$ is the unique solution to* **(P0)**.

Readers can easily derive a rigorous proof by following the above idea. Next, we present a different proof, which we think is more straightforward.

**Proof**. Suppose matrices $X_1$ and $X_2 \in \mathbb{R}^{n\times L}$ are the solutions to **(P0)** with property $\max\{\mathcal{R}(X_1), \mathcal{R}(X_2)\} < Spark(A)/2$. We have

$$\mathcal{R}(X_1 - X_2) \leq \mathcal{R}(X_1) + \mathcal{R}(X_2) < Spark(A). \tag{2.1}$$

On the other hand, because $0 = A(X_1 - X_2)$, if we consider $(X_1 - X_2)^{(1)}$, which is the first column of matrix $X_1 - X_2$, we have $0 = A(X_1 - X_2)^{(1)}$. It leads to $\|(X_1 - X_2)^{(1)}\|_0 \geq Spark(A)$. Therefore, we have $\mathcal{R}(X_1 - X_2) \geq Spark(A)$, which contradicts (2.1). This contradiction proves the theorem. □

If we are willing to consider the additional feature of matrix $B$ – to take advantage of the MMV formulation – a more general condition can be derived. A precedent is Lemma 1 in Cotter et al. [22]. The following result is more general.

**Theorem 2.2.4** *Let $Rank(B)$ denote the rank of matrix $B$. Apparently $Rank(B) \leq L$. Matrix $X$ will be the unique solution to the problem* **(P0)**, *if $B = AX$ and*

$$\mathcal{R}(X) < [Spark(A) - 1 + Rank(B)]/2.$$

**Proof**. Recall $B \in \mathbb{R}^{m\times L}$. Suppose we have $B = AX_1 = AX_2$, where $X_1, X_2 \in \mathbb{R}^{n\times L}$, and $X_1 \neq X_2$. Let $d(Null(B))$ denote the dimension of the (right) null space of matrix $B$: $\{x : Bx = 0\}$. Similarly, $d(Null(X_1))$ and $d(Null(X_2))$ denote the dimensions of the (right) null spaces of matrices $X_1$ and $X_2$. We have

$$d(Null(X_1)) \leq d(Null(B)),$$

and

$$d(Null(X_2)) \leq d(Null(B)).$$

Recall $Rank(B)$ denotes the rank of matrix $B$. Similarly let $Rank(X_1)$ and $Rank(X_2)$ denote the ranks of matrices $X_1$ and $X_2$. We have

$$Rank(X_1) \geq Rank(B), \tag{2.2}$$

and

$$Rank(X_2) \geq Rank(B). \tag{2.3}$$

Consider a matrix $[A_1, A_{12}, A_2]$, where submatrix $[A_1, A_{12}]$ is made by the columns of matrix $A$ that correspond to the nonzero rows of matrix $X_1$ and submatrix $[A_{12}, A_2]$ is made by the columns of matrix $A$ that correspond to the nonzero rows of matrix $X_2$. Let $r_1$ and $r_2$ denote the numbers of nonzero rows in matrices $X_1$ and $X_2$ respectively. Note that matrix $A_{12}$ corresponds to columns where matrices $X_1$ and $X_2$ have nonzero rows simultaneously. Let $r_{12}$ denote the number of columns of matrix $A_{12}$. Matrix $X_{11}$ (resp. Matrix $X_{22}$) consists of the nonzero rows of matrix $X_1$ (resp. $X_2$) corresponding to $A_1$ (resp. $A_2$). Matrix $X_{12}$ (resp. Matrix $X_{21}$) consists of the nonzero rows of $X_1$ (resp. $X_2$) corresponding to the columns in $A_{12}$. We have

$$B = [A_1, A_{12}] \cdot \begin{bmatrix} X_{11} \\ X_{12} \end{bmatrix} = [A_{12}, A_2] \cdot \begin{bmatrix} X_{21} \\ X_{22} \end{bmatrix}$$

and

$$0 = A(X_1 - X_2) = [A_1, A_{12}, A_2] \cdot \begin{bmatrix} X_{11} \\ X_{12} - X_{21} \\ -X_{22} \end{bmatrix}. \tag{2.4}$$

From (2.4), we have

$$d(Null([A_1, A_{12}, A_2])) \geq Rank(\begin{bmatrix} X_{11} \\ X_{12} - X_{21} \\ -X_{22} \end{bmatrix}). \tag{2.5}$$

It is easy to see

$$\begin{aligned} & Rank(\begin{bmatrix} X_{11} \\ X_{12} - X_{21} \\ -X_{22} \end{bmatrix}) \\ \geq & \max\{Rank(X_{11}), Rank(X_{22})\}. \end{aligned} \tag{2.6}$$

Without loss of generality, we consider $X_{11}$ only. It is easy to see that

$$d(Null(X_{11})) \leq d(Null(X_1)) + r_{12}. \tag{2.7}$$

The above is true because if we consider two systems of linear equations: for variable y, $X_1 \cdot y = 0$ or $X_{11} \cdot y = 0$; the former has $r_{12}$ more constrains, so its solution space(the null space of matrix $X_1$) is at most reduced by $r_{12}$ dimensions, which is (2.7). Inequality (2.7) immediately leads to

$$Rank(X_1) - r_{12} \leq Rank(X_{11}). \tag{2.8}$$

Combining (2.5), (2.6), (2.8), and one of (2.2) and (2.3), we have

$$d(Null([A_1, A_{12}, A_2])) \geq Rank(B) - r_{12}. \tag{2.9}$$

By the definition of $Spark$, we have

$$\begin{aligned} & Rank([A_1, A_{12}, A_2]) \\ \geq & Spark([A_1, A_{12}, A_2]) - 1 \\ \geq & Spark(A) - 1. \end{aligned} \tag{2.10}$$

Combining all the above, we have

$$\begin{aligned} r_1 + r_2 - r_{12} &= \#Cols([A_1, A_{12}, A_2]) \\ &= Rank([A_1, A_{12}, A_2]) \\ &\quad + d(Null([A_1, A_{12}, A_2])) \\ &\geq Spark(A) - 1 \\ &\quad + Rank(B) - r_{12}. \end{aligned}$$

Hence $r_1 + r_2 \geq Spark(A) - 1 + Rank(B)$. The last inequality is based on (2.9) and (2.10). It is easy to see that the above proves the theorem. □

It turns out that to study the theoretical property of the $\ell_0$-norm approach, we only need the fact that $m(x) = 0$ if and only if $x = \vec{0}$, where $\vec{0}$ is an all zero vector in $\mathbb{R}^L$. It is evident that Theorem 2.2.2 is a special case of Theorem 2.2.4, as $Rank(B) \geq 1$.

For the upper bound of Theorem 2.2.4, it is conceptually interesting to ask whether the rank of matrix $B$ can be replaced with the rank of matrix $X$. Because $B = AX$, it is evident that $Null(X) \subset Null(B)$. Hence $d(Null(X)) \leq d(Null(B))$. Therefore, $Rank(X) \geq Rank(B)$. Given this, it is not clear how to utilize the existing approach to generate an upper bound that is based on $Rank(X)$. Further exploitation in this direction will be a future research topic.

### 2.2.3 Mutual Incoherence and $\mu_{1/2}(G)$

A difficulty associated with an upper bound with $Spark(A)$ is that the quantity $Spark$ is hard to calculate, as pointed out by Donoho and Elad [30]. Up to now, there is no good algorithm to compute $Spark(A)$ besides enumerating all the possible subsets. For practical use, we introduce other quantities: *mutual incoherence* and $\mu_{1/2}(G)$. These quantities have appeared in previous papers, e.g., [33, 36, 30]. They provide upper bounds that are lower than the one that is built on $Spark$. However, these quantities are easy to compute.

**Definition 2.2.5** *Mutual incoherence (denoted by $M$) is the maximum absolute inner product between two column vectors of matrix $A$, i.e.,*

$$M = M(A) = \max_{1 \leq i,j \leq n, i \neq j} |G(i,j)|,$$

*where $G(i,j)$ is the $(i,j)$th entry of the Gram matrix $G$: $G = A^T A$.*

Note that quantities $M$ and $Spark$ have the following relation, which has been proved in Donoho and Elad [30, Theorem 7]:

$$Spark(A) \geq (1 + 1/M).$$

Therefore, an upper bound with $Spark(A)$ is better. In fact, the above inequality together with Theorem 2.2.4 gives a one-line proof of the following corollary. We omit the proof.

**Corollary 2.2.6** *If* $B = AX$ *and*

$$\mathcal{R}(X) < (M^{-1} + Rank(B))/2,$$

*then matrix* $X$ *is the unique solution to the problem* **(P0)**.

We consider another quantity.

**Definition 2.2.7** *For a Gram matrix* $G$*, which is symmetric, let* $\mu_{1/2}(G)$ *denote the smallest number* $m$*, such that the sum of a collection of* $m$ *off-diagonal magnitudes in a single row or column of the Gram matrix* $G$ *is at least* $1/2$.

In [30, Theorem 6 and Section 4.2], we can find the following relation: $Spark(A) \geq 2\mu_{1/2}(G) + 1$. Combining with Theorem 2.2.4, we immediately have the following.

**Corollary 2.2.8** *If* $B = AX$ *and*

$$\mathcal{R}(X) < \mu_{1/2}(G) + Rank(B)/2,$$

*then matrix* $X$ *is the unique solution to the problem* **(P0)**.

We conclude our analysis of the uniqueness in the minimizing-the-$\ell_0$-norm approach.

## 2.3 Minimizing the $\ell_1$ Norm

### 2.3.1 Formulation

Recall that we have defined a sparsity rank of matrix $X \in \mathbb{R}^{n\times L}$,

$$\mathcal{R}(X) = \|(m(x_i))_{n\times 1}\|_0$$

where $m(x_i)$ is a vector norm in $\mathbb{R}^L$. In this section, we consider a relaxation to the above quantity.

We consider the following function as a relaxation of the quantity $\mathcal{R}(X)$:

$$Relax(X) = \|(m(x_i))_{n\times 1}\|_1.$$

Note that the only difference between $\mathcal{R}(X)$ and $Relax(X)$ is that the outside $\ell_0$ norm is replaced by an $\ell_1$ norm. The corresponding optimization problem becomes

$$\textbf{(P1):} \quad \min \quad Relax(X), \quad \text{subject to } B = AX.$$

The above formulation includes many known works. For example, in Tropp [104], $m(\cdot)$ is the $\ell_\infty$ norm; in Malioutov et al. [79], $m(\cdot)$ is the $\ell_2$ norm.

Besides the $\ell_1$ norm, other functions of $X$ have been proposed as objective functions. In the pioneer works on MMV [22, 93, 69], the following diversity measure on sparsity was proposed:

$$J^{(p,q)}(x) = \sum_{i=1}^{n} (\|x^{(i)}\|_q)^p, \quad 0 \le p \le 1, q \ge 1,$$

where $p$ and $q$ are parameters, vector $x^{(i)}$ is the $i$th row of matrix $X$. The norm of a row is given by $\|x^{(i)}\|_q = (\sum_{j=1}^{L} |x_{ij}|^q)^{1/q}$. An algorithm, which was named M-FOCUSS is proposed to minimize the above objective [22]. The M-FOCUSS, for $q = 2, p \le 1$, is an iterative algorithm that uses the idea of Lagrange multipliers. A disadvantage of the above objective function is that it could have more than one local minima, e.g., when $p < 1$. An iterative algorithm could be trapped by a local minimum. With $p = 1$ in the above objective, we obtain the $\ell_1$-norm minimization problem **(P1)** with $\ell_q$ norm inside.

### 2.3.2 Uniqueness under the $\ell_1$ Norm

We consider an optimal solution to the problem **(P1)**. Let $B = AX^*$, where $X^*$ is the optimal solution to the problem **(P0)**. Let $S$ be an index set that contains the rows of $X^*$ where $m(x_i^*) > 0$. Here $x_i^*$ denotes the $i$th row of matrix $X^*$. Let $A_S$ denote a matrix that is made by the columns of $A$ with indices from $S$. We can write $B = A_S X_S^*$, where matrix $X_S^*$ is made by the nonzero rows of $X^*$. Without loss of generality, we can assume that $A_S$ is of full column rank; otherwise, the number of nonzero rows of $X^*$ can be reduced, which contradicts the optimality. We define the generalized inverse of $A_S$ to be $A_S^+ = (A_S^T A_S)^{-1} A_S^T$. Based on the fact that $A_S$ is of full column rank, the generalized inverse is well defined. We present a sufficient condition of the sparsity of $X^*$ in the following.

**Theorem 2.3.1** *A sufficient condition for $X^*$ to be the unique solution to* **(P1)** *is that*

$$\|A_S^+ A_j\|_1 < 1, \forall j \notin S. \tag{3.11}$$

Note that the above is the Exact Recovery Condition in Tropp's [103]. See also [46]. It turns out that it is also a sufficient condition for the uniqueness under the $\ell_1$-norm for MMV, *with an arbitrary inner vector norm $m(\cdot)$*. Readers may want to revisit the formulation of **(P1)**.

As a preparation for the proof of theorem 2.3.1, the following is a well-known result for norms in the Euclidean space. We present it without a proof.

**Proposition 2.3.2** *For a linear combination $\sum_{i=1}^k c_i x_i$, where $c_i \in \mathbb{R}$, $x_i \in \mathbb{R}^L$, and $k$ is an integer, for any norm $m(\cdot)$ in $\mathbb{R}^L$, we have*

$$m\left(\sum_{i=1}^k c_i x_i\right) \leq \sum_{i=1}^k |c_i| \cdot m(x_i).$$

**Proof of theorem 2.3.1.** Suppose there are two representations: $B = A_S X_S^* = A_{S'} Y_{S'}$, where $S \neq S'$ and set $S'$ includes the indices of the nonzero rows of the matrix $Y \in \mathbb{R}^{n \times L}$. We only need to show that

$$Relax(X^*) < Relax(Y). \tag{3.12}$$

Recall

$$Relax(X^*) = \|(m(x_i^*))_{n\times 1}\|_1 = \sum_{i=1}^{n} m(x_i^*) = \sum_{i \in S} m(x_i^*).$$

Because $X_S^* = (A_S^+ A_{S'}) Y_{S'}$, we have

$$x_i^* = \sum_k (A_S^+ A_{S'})_{ik} (Y_{S'})_k,$$

where$(A_S^+ A_{S'})_{ik}$ is the $(i,k)$th entry of the matrix $A_S^+ A_{S'}$, and $(Y_{S'})_k$ is the $k$th row of $Y_{S'}$. Note that the above is a linear combination. From Proposition 2.3.2, we have

$$m(x_i^*) \leq \sum_k |(A_S^+ A_{S'})_{ik}| m((Y_{S'})_k).$$

Taking $\sum_i$ on both sides, we have

$$\sum_i m(x_i^*) \leq \sum_i \sum_k |(A_S^+ A_{S'})_{ik}| m((Y_{S'})_k) < \sum_k m(Y_k).$$

The last inequality is based on two facts (see Acknowledgement). The first fact is that $S'$ contains at least one column that does not appear in $S$. Otherwise, $S'$ would be a strict subset of S, which contradicts the minimality of S. Therefore, there must exist some k, such that $\sum_i |(A_S^+ A_{S'})_{ik}| < 1$, based on (3.11). The other fact is that $\|A_S^+ A_j\|_1 \leq 1$ for every column j in matrix A. Hence we prove (3.12). □

### 2.3.3 Equivalence

In [103, Theorem B and Corollary 3.6], we know whenever one of the following conditions is satisfied:

$$\mathcal{R}(X^*) < (1 + 1/M)/2 \tag{3.13}$$

or

$$\mathcal{R}(X^*) < \mu_{1/2}(G), \tag{3.14}$$

$\max_{j\notin S} \|A_S^+ A_j\|_1 < 1$ holds for any signal with $\mathcal{R}(X^*)$ atoms in its optimal representation. Therefore, according to Theorem 2.3.1, when (3.13) or (3.14) holds, $X^*$ is the unique solution to **(P1)**.

On the other hand, according to [30], we have the following relation: $Spark(A)/2 > \mu_{1/2}(G) \geq \frac{1}{2M}$. Thus, according to Theorem 2.2.2, if $B = AX$ and $\mathcal{R}(X) < (1 + 1/M)/2$ or $\mathcal{R}(X) < \mu_{1/2}(G)$, X is the unique sparest solution to **(P0)**, i.e., $X = X^*$.

From all the above, we have the following theorem.

**Theorem 2.3.3 (Equivalence)** *For a dictionary $A$ with Gram matrix $G = A^T A$. If $AX = B$ and*

$$\mathcal{R}(X) < (1 + 1/M)/2$$

*or*

$$\mathcal{R}(X) < \mu_{1/2}(G),$$

*then matrix $X$ is the unique solution to* **(P1)**. *And this solution is identical with the solution to* **(P0)**.

Note that our condition of equivalence in the above theorem is identical with the one in SMV. Recall that by taking into account of the property of matrix $B$, a stronger uniqueness condition is achieved in the $\ell_0$-like norm. The difficulty in getting a stronger equivalence condition for MMV is that the uniqueness of the $\ell_1$-norm approach does not seem to depend on the matrices $B$ or $X$.

It is interesting to realize that the proof of SMV still works for any norm $m(\cdot)$ in $\mathbb{R}^L$.

### 2.3.4 Comparison between SMV and MMV

In the minimizing-the-$\ell_0$-norm problem, by taking advantage of the formulation of MMV, we can raise the upper bound in the uniqueness condition from $Spark(A)/2$ to $[Spark(A) - 1 + Rank(B)]/2$.

There is no evidence that between condition $\mathcal{R}(x) < Spark(A)/2$ and condition $\max_{j \notin S} \|A_S^+ A_j\|_1 < 1$, one is able to dominate the other. In principle, if $\max_{j \notin S} \|A_S^+ A_j\|_1 < 1$ and $Spark(A)/2 < \mathcal{R}(x) < [Spark(A) - 1 + Rank(B)]/2$, we can claim the equivalence between $\ell_0$-norm and $\ell_1$-norm for MMV, and this is not achievable by simply concatenating SMV problems.

Here is another difference between an MMV problem and an SMV problem. Note that if we find the sparsest representation for $B$ under the condition in Theorem 2.3.1, we do *not* have enough evidence that each column of $X^*$ can be obtained by solving an SMV problem for each column of $B$. The reason is that from $\max_{j \notin S} \|A_S^+ A_j\|_1 < 1$, where $S$ consists of the atoms in the optimal representation of matrix $B$, it is not necessary to have $\max_{j \notin S_i} \|A_{S_i}^+ A_j\|_1 < 1$, where $S_i$ consists of the atoms in the optimal representation of vector $b^{(i)}$. This is because the number of the atoms in the optimal representation of vector $b^{(i)}$ may be less than the number of the atoms in the optimal representation of matrix $B$. In summary, the uniqueness conditions under the $\ell_1$-norm differ between in the formulation of an MMV and in the formulation of a combination of several SMVs. (The description here is speculative and consequently raises an open question: What can we say about the relationship between the set $S$ and the group of sets $\{S_i\}$? We leave this question to be studied in the future.)

## *2.4 Orthogonal Matching Pursuit*

Matching pursuit (MP) [81] is proposed as an efficient numerical method to decompose a signal. As an improvement of MP, orthogonal matching pursuit (OMP) [24, 90]

has been introduced. OMP overcomes some drawbacks of MP. Unfortunately, counterexamples show that both methods could be trapped by the initial selection of a 'bad' atom, see Chen et al. [15]. For the MMV problem, many variants of OMP have been proposed. A subset of them are [22, 106, 107, 73, 74, 78, 99].

For this section, we propose our OMP with an $\ell_q (q \geq 1)$ norm of the inner product. Note that in MMV, the inner product becomes a vector. A condition that guarantees the exact recovery of OMP is derived. This condition is identical to the corresponding Exact Recovery Condition in SMV, see [103]. Again, it is interesting to see that an existing condition holds for a large class of vector norms.

### 2.4.1 OMP Algorithm for MMV

An OMP in MMV, which is denoted by OMPMMV, works as follows.

---

Orthogonal Matching Pursuit for MMV (OMPMMV)

1. Initialization: residual $R_0 = B$ and subset $S_0 = \emptyset$.
2. At the $t$th iteration:
   (a) Choose the atom $a_{k_t}$, which satisfies $a_{k_t} = \text{argmax}_{a_k} \|z_k\|_q$, where $z_k = R_{t-1}^T a_k$ and $q \geq 1$;
   (b) Let $S_t = [S_{t-1}, a_{k_t}]$, and $X^* = \text{argmin}_X \|S_t X - B\|_F^2, y_t = S_t X^*$;
   (c) Set $R_t = B - y_t$.

---

Readers can find that except taking the $\ell_q$-norm of the vector $z_k$ in step 2)-a), the remaining components in the above algorithm are standard in an OMP.

In [106, 107], Tropp et al. proposed $\ell_1$ norm in step 2)-a). In [73, 74, 78, 99], $\ell_2$ and $\ell_\infty$ are proposed for weak matching pursuit and weak orthogonal matching pursuit for the MMV problem. In [22], $\ell_2$ norm is applied. We will prove that, when the coefficient matrix of $B$ is very sparse, no matter what the $\ell_q$ norm is, an OMP with the $\ell_q$ norm in 2)-a) can recover the sparsest representation.

### 2.4.2 Matrix Norm Preparation

Before providing the proof, we introduce some necessary notations and results that will be used in this section.

**Definition 2.4.1** *The $(p, q)$ matrix (or operator) norm of $A$ is defined as*

$$\|A\|_{p,q} = \max_{x \neq 0} \frac{\|Ax\|_q}{\|x\|_p} = \max_{\|x\|_p = 1} \|Ax\|_q.$$

Several of the $(p, q)$ matrix norms can be computed easily, see also [49, 107].

**Lemma 2.4.2** *Consider matrix $A$.*

1. *The $(1, q)$ matrix norm is the maximum $\ell_q$ norm of the columns of $A$.*
2. *The $(2, 2)$ matrix norm is the maximum singular value of $A$.*
3. *The $(p, \infty)$ norm is the maximum $\ell_{p'}$ norm of the rows of $A$, where $1/p + 1/p' = 1$.*

The following property regarding $(p, q)$ matrix norm can be easily derived from the definitions or results mentioned above.

**Lemma 2.4.3** *For matrix $A$, we have*

1. *$\|Ax\|_q \leq \|A\|_{p,q} \cdot \|x\|_p$, and*
2. *$\|A^T\|_{\infty,\infty} = \|A\|_{1,1}$.*

In particular, the $(p, \infty)$ matrix norm has the following property.

**Lemma 2.4.4** *For matrices $A$ and $B$, and $p > 0$, we have*

$$\|AB\|_{p,\infty} \leq \|A\|_{\infty,\infty} \|B\|_{p,\infty}.$$

**Proof.** The following are direct applications of some previous results.

$$\begin{aligned}
\|AB\|_{p,\infty} &= \max_{\|x\|_p=1} \|A(Bx)\|_\infty \\
&\leq \max_{\|x\|_p=1} \|A\|_{\infty,\infty} \|(Bx)\|_\infty \\
&= \|A\|_{\infty,\infty} \max_{\|x\|_p=1} \|(Bx)\|_\infty \\
&= \|A\|_{\infty,\infty} \|B\|_{p,\infty}.
\end{aligned}$$

We prove the lemma. □

### 2.4.3 Main Result

Note that OMP never chooses the same atom twice because the residual is orthogonal to the atoms that have already been selected. If at each step, OMP selects the atoms in the optimal representation, after $\mathcal{R}(X^*)$ steps, the residual must become zero, and the algorithm stops. Note that since we only consider the *noiseless* formulation, we are allowed to use such an idealistic argument.

According to our notation, in step 2)-a) in OMPMMV, we have $\max_{a_k} \|z_k\|_q = \max_{a_k} \|a_k^T R_{t-1}\|_q = \|A^T R_{t-1}\|_{p,\infty}$, where $1/p+1/q = 1$. Thus at $(t+1)$th step, we can select the atom in the optimal representation if and only if $\|A_S^T R_t\|_{p,\infty} > \|A_{\bar{S}}^T R_t\|_{p,\infty}$, where $\bar{S}$ is the complement of $S$ in the dictionary $\Omega$. Following this idea, we have the following theorem with the same notations used in the previous sections.

**Theorem 2.4.5** *A sufficient condition for OMPMMV to recover a representation of matrix $B$ associated with atom indices $S$ is*

$$\max_{j \notin S} \|A_S^+ A_j\|_1 < 1.$$

Readers can see that the above is again the Exact Recovery Condition in [103]. In fact, readers can see that the following proof is modified from the corresponding proof in [103].

**Proof**. At each iteration $t$,

$$\begin{aligned}\rho_t &= \frac{\|A_S^T R_t\|_{p,\infty}}{\|A_{\bar{S}}^T R_t\|_{p,\infty}} \\ &= \frac{\|A_S^T R_t\|_{p,\infty}}{\|A_{\bar{S}}^T (A_S^+)^T A_S^T R_t\|_{p,\infty}} \\ &\geq \frac{1}{\|A_{\bar{S}}^T (A_S^+)^T\|_{\infty,\infty}}.\end{aligned}$$

In the above, we use the equality

$$R_t = (A_S^+)^T A_S^T R_t. \tag{4.15}$$

Recall $(A_S^+)^T A_S^T = A_S (A_S^T A_S)^{-1} A_S^T$, which is a projection matrix to the subspace spanned by the columns of matrix $A_S$. Because $A_S$ is the optimal set, the columns of $R_t$ is in the subspace spanned by the columns of $A_S$. Hence we have (4.15).

To pick the atom in the optimal representation, we want $\rho_t > 1$, that is $\|A_{\bar{S}}^T (A_S^+)^T\|_{\infty,\infty} < 1$. Moreover, we have

$$\|A_{\bar{S}}^T (A_S^+)^T\|_{\infty,\infty} = \|(A_S^+) A_{\bar{S}}\|_{1,1} = \max_{j \in \bar{S}} \|A_S^+ A_j\|_1 < 1.$$

This completes the proof. □

Applying the same argument that has been used in the $\ell_1$-norm, we have the following corollary.

**Corollary 2.4.6** *If $AX = B$ and*

$$\mathcal{R}(X) < (1 + 1/M)/2$$

*or*

$$\mathcal{R}(X) < \mu_{1/2}(G),$$

*matrix $X$ is the unique sparsest solution to* **(P0)**, *and OMPMMV can recover this representation exactly.*

Compared with Theorem 2.4.5, the conditions in the above corollary are much easier to check. We can calculate matrix $X$ through OMPMMV first, and then check if such a matrix $X$ satisfies the conditions.

## *2.5 Simulation*

### 2.5.1 Exact Recovery of OMPMMV and (P1)

Simulations are conducted to bring insights on when OMPMMV and **(P1)** can exactly find the *original* signal. Two experiments are conducted. In the first experiment, matrix $A \in \mathbb{R}^{m\times n}$ has dimensions $m = 20$ and $n = 30$. We set $L = 5$ and $m(\cdot) = \ell_1$. The entries of matrix $A$ are independently sampled from the standard normal $N(0, 1)$. We compute the multiple measurement vector, $B$, using $B = AX_0$, where the $N$ nonzero rows of matrix $X_0 \in \mathbb{R}^{30\times 5}$ are randomly chosen, and the values of the nonzero entries of matrix $X_0$ are assigned again by independently sampling from the standard normal distribution. The value of $N$ is ranged from 1 to $\lceil (1+1/M)/2 \rceil + 15$. For each generated pairs of matrices $B$ and $A$, matrix $X$ is solved via both OMPMMV with $\ell_2$ norm and **(P1)** with $\ell_1$ norm. The solution $X$ is compared with the original matrix $X_0$. If $X \equiv X_0$, an 'exact recovery' is obtained. The above simulation is executed for 1000 times for the same matrix A. The proportion of exact recoveries among $1,000$ times of simulation via OMPMMV (resp. **(P1)**) for the N nonzero rows of matrix $X_0$ is reported as 'the empirical probability of exact recovery' for the value N via OMPMMV (resp. **(P1)**) in Figure 1(a). We observed that the OMPMMV performs slightly better. Symbol $*$ indicates where the theoretical upper bound for uniqueness is (i.e., $\lfloor (1 + 1/M)/2 \rfloor$). In the figures, it is not very evident where the above proportions are equal to 1 – the proportions of exact recoveries are very close to 1, but not 1. We introduce two symbols to indicate the positions when the proportion are exactly equal to 1: symbol $\oplus$ indicates the largest value of $N$ while OMPMMV finds the original $X_0$ among all simulations; symbol $\ominus$ indicates the largest value of

$N$ while the solutions of **(P1)** are identical with matrix $X_0$ for all simulations.

In the second experiment, matrix $A$ is generated by concatenating two orthonormal bases: $[I, H]$, where matrix $I$ is an identity matrix and matrix $H$ is a Hadamard matrix. We choose $m = 16$ and $n = 32$. Matrix $X_0$ has $L = 3$ columns. All the other settings are the same as in the first experiment. Again, we observe that the OMPMMV performs slightly better.

In both cases, we observe that the exact recovery can occur when the value of $N$ is above the theoretical threshold ($\lfloor (1 + 1/M)/2 \rfloor$) that is given in this chapter. Based on this, we say that the theoretical upper bound is pessimistic.

### 2.5.2 Comparison of Different Vector Norms in (P1)

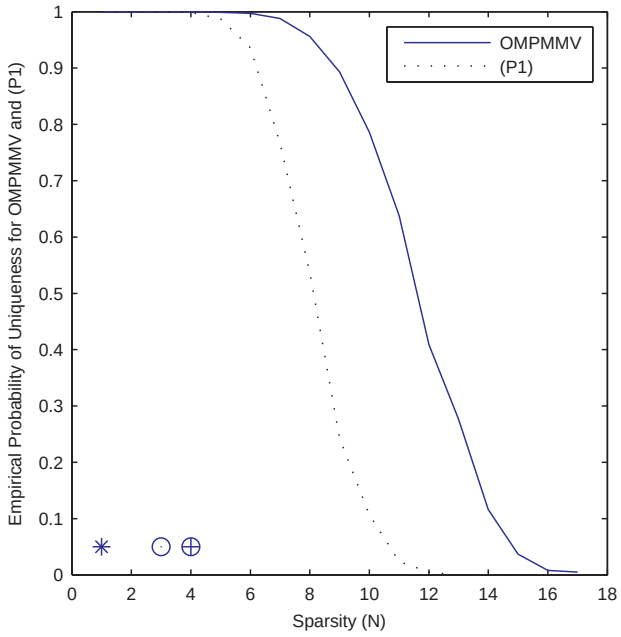
The settings in the subsection are the same as those in the last subsection. In this subsection, we do the simulation for the $\ell_1$ norm method with different $m(\cdot)$ norms. Firstly, for matrix $A$ randomly generated from Normal$(0, 1)$, where $m = 30, n = 20, L = 5$, we do the simulation for $m(\cdot) = \ell_1$ and $m(\cdot) = \ell_\infty$, respectively, and draw their empirical probabilities of exact recoveries on one plot. See the results in Figure 2 (a). Secondly, for $A = [I, H]$, where sub-matrix $H$ is a 16 by 16 Hadamard matrix, matrix $I$ is a 16 by 16 identity matrix, and $L = 3$, we do the same simulation as above. See the results in Figure 2 (b).

From the simulations, we see that the curves in each plot are similar. This might imply that for the method of $\ell_1$ norm , there is no significant difference among different inside vector norms.

### 2.5.3 Comparison of Different Vector Norms in OMPMMV

The settings in this subsection are the same as those in the previous subsections. In this subsection, we do the simulation for the OMPMMV method with different $\ell_q$ norms. Firstly, for matrix $A$ whose entries are randomly generated from Normal$(0, 1)$, where $m = 30, n = 20, L = 5$, we do the simulation for $\ell_1, \ell_2$ and $\ell_\infty$, respectively, and

(a)




(b)

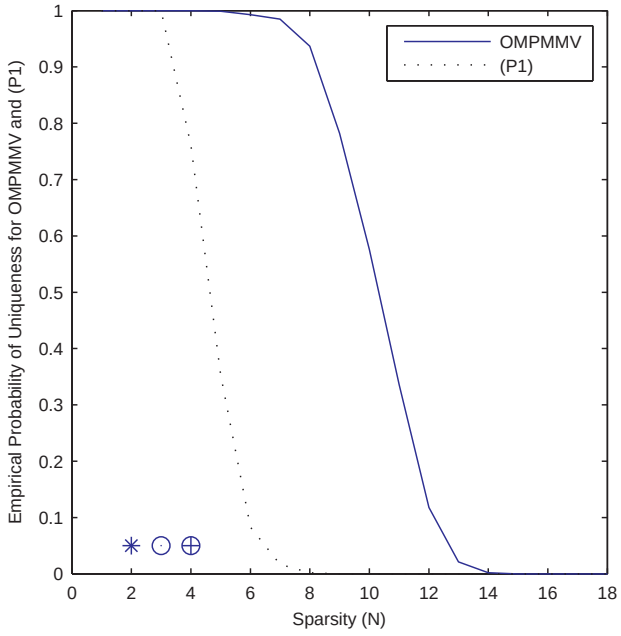
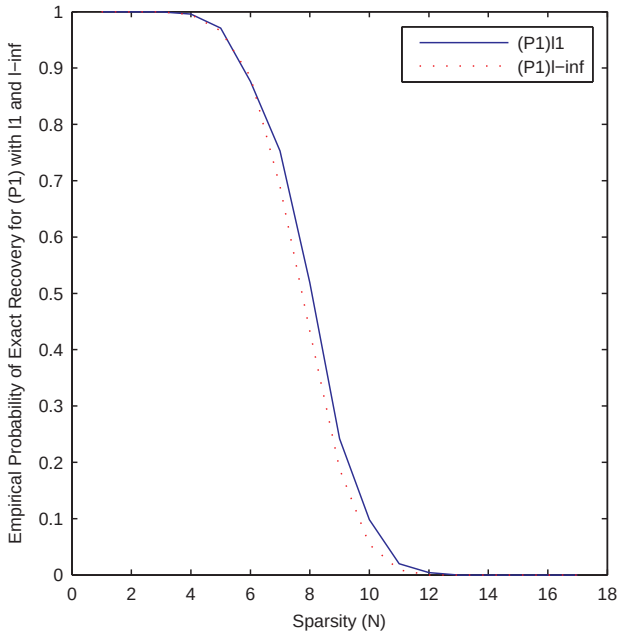



**Figure 1:** **(a)** First experiment of exact recovery, in which $A \in \mathbb{R}^{m\times n}, X_0 \in \mathbb{R}^{n\times L}, m = 20, n = 30, L = 5$, where entries of matrices $A$ and $X_0$ are independently sampled from $N(0,1)$. Symbol $*$ is marked at 1. For the OMPMMV, the $\oplus$ is marked at $N = 4$; while for **(P1)**, $\ominus$ is marked at $N = 3$. **(b)** We now have matrix $A = [I, H]$ where matrix $A \in \mathbb{R}^{16\times 32}$ and sub-matrix $H$ is a 16 by 16 Hadamard matrix. Matrix $I$ is a 16 by 16 identity matrix. Matrix $X_0$ is chosen in the same way, with $L$ equal to 3 and $N$ being the number of nonzero rows. In this case, symbol $*$ is marked at $N = 2$. Symbols $\oplus$ and $\ominus$ are at $N = 4$ and $N = 3$ respectively.

(a)




(b)

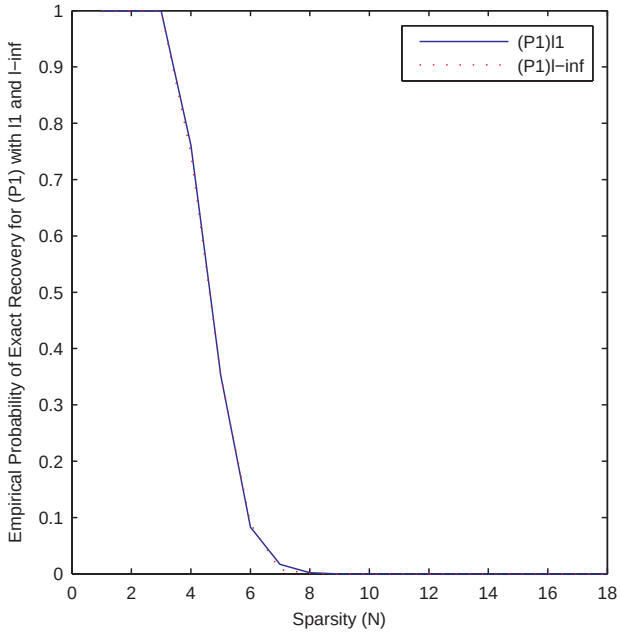



**Figure 2:** **(a)** Consider the case $A \in \mathbb{R}^{m\times n}, X_0 \in \mathbb{R}^{n\times L}$, m=20, n=30, L=5, where entries of matrices $A$ and $X_0$ are independently sampled from $N(0,1)$. The theoretical upper bound for the equivalence is 1. Let $N_i$, $i = 1, \infty$ denote the largest value of $N$ when the solutions of **(P1)** with $m(\cdot)$ being the $\ell_i$ norm are identical with matrix $X_0$ among all of the 1000 simulations. We have $N_1 = N_\infty = 3$. **(b)** We now consider matrix $A = [I, H]$ where submatrix $H$ is a 16 by 16 Hadamard matrix and submatrix $I$ is a 16 by 16 identity matrix. We have $L = 3$. The theoretical upper bound for equivalence is 2. We obtain $N_1 = N_\infty = 3$.

draw their empirical probabilities of exact recoveries on one plot. See the results in Figure 3 (a). Secondly, for $A = [I, H]$, where sub-matrix $H$ is a 16 by 16 Hadamard matrix, $I$ is a 16 by 16 identity matrix, and $L = 3$, we do the same simulation as above. Results are shown in Figure 3 (b).

From the simulations, we see that the curves in each plot are similar. This might demonstrate that for the same method ($\ell_1$ norm or OMP), among different vector norms, there is no significant difference.

## *2.6 Discussion*

### 2.6.1 Better Vector Norms in MMV?

In both sparsity rank $\mathcal{R}(X)$ and its relaxation $Relax(X)$ of matrix $X \in \mathbb{R}^{n\times L}$, we choose an arbitrary norm in $\mathbb{R}^L$. One logic question is whether or not one norm can consistently outperform another norm. To be more specific, we introduce the following dominance concept.

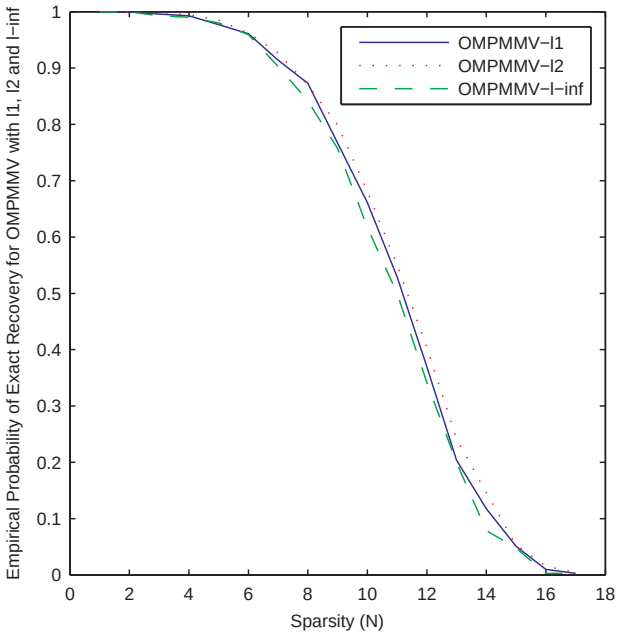
**Definition 2.6.1 (Dominance)** . *We say that a norm $m_1(x)$ is dominated by a norm $m_2(x)$ in $\mathbb{R}^L$, if and only if for any $x, y \in \mathbb{R}^L$, $m_1(x) < m_1(y)$ leads to $m_2(x) < m_2(y)$.*

If norm $m_2$ dominates norm $m_1$, then $m_2$ should always be used. The reason is as following. Denote two relaxations $Relax_1(X) = \sum_{i=1}^n m_1(x_i)$ and $Relax_2(X) = \sum_{i=1}^n m_2(x_i)$. Whenever **(P1)** with relaxation $Relax_1(X)$ finds the unique sparest solution, i.e., $Relax_1(X^*) < Relax_1(Y)$ for any other $Y$, **(P1)** with relaxation $Relax_2(X)$ finds the unique sparsest solution too, i.e., $Relax_2(X^*) < Relax_2(Y)$ for any other $Y$.

For norms in a Euclidean space, the following result demonstrates that no norm can dominate another. The only special case is that they are equivalent.

**Lemma 2.6.2** *If norm $m_2$ dominates norm $m_1$, then there exists a constant $C > 0$, such that $m_1(x) = C \cdot m_2(x), \forall x \in \mathbb{R}^L$.*

(a)




(b)

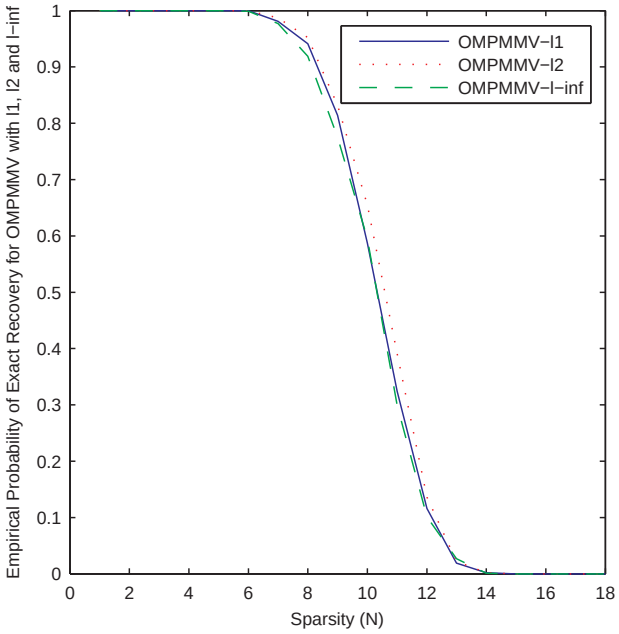



**Figure 3:** **(a)** We consider $A \in \mathbb{R}^{m\times n}, X_0 \in \mathbb{R}^{n\times L}, m = 20, n = 30, L = 5$, where entries of matrices $A$ and $X_0$ are independently sampled from $N(0, 1)$. The theoretical upper bound for equivalence is 1. Notation $N_i$, $i = 1, 2, \infty$, denotes the largest value of $N$ while OMPMMV with $\ell_i$ norm in step 2)-a) finds the original $X_0$ among all the 1000 trials. We have $N_1 = N_2 = N_\infty = 2$. **(b)** We have matrix $A = [I, H]$ where submatrix $H$ is a 16 by 16 Hadamard matrix and submatrix $I$ is a 16 by 16 identity matrix. We have $L = 3$. The theoretical upper bound for equivalence is 2. We obtain $N_1 = N_2 = 6$ and $N_\infty = 5$.

**Proof**. First of all, for any pair $x, y \in \mathbb{R}^L$ satisfying $m_1(x) = m_1(y)$, we prove that $m_2(x) = m_2(y)$. This can be seen from the following. It is easy to see that for $\tau > 0$,

$$m_1((1-\tau)y < m_1(x) < m_1((1+\tau)y).$$

From the dominance, we have

$$m_2((1-\tau)y < m_2(x) < m_2((1+\tau)y).$$

Let $\tau \to 0$, we have $m_2(x) = m_2(y)$.

In the second step, we choose a special $x_0 \in \mathbb{R}^L$, such that $m_1(x_0) = 1$. Because

$$m_1\left(\frac{1}{m_1(y)}y\right) = 1 = m_1(x_0),$$

we have

$$m_2\left(\frac{1}{m_1(y)}y\right) = m_2(x_0).$$

Hence

$$m_2(y) = m_1(y) \cdot m_2(x_0).$$

Note that $m_2(x_0)$ is a constant, we have proved the lemma. □

Note our notion of "equivalent" differs from some uses in the literature, e.g., [57, page 269]. More specifically, the result such as Theorem 5.4.4 in [57] is typically associated with the equivalence. The foregoing lemma is similar to some description in the Section 5.5 of [57]; however, we failed to find a direct reference.

The above demonstrates that there is no optimal relaxation when SMV is generalized to MMV. Note that we consider the optimality in the worst case. If we know some properties about $X$ or $B$, some norms may work better than other norms in function $Relax(X)$, e.g., on statistical average. We leave it as an open question.

### 2.6.2 Simulation

In the simulation, we verify the criterion of 'exact recovery', instead of the sparsest representation as formulated in **(P0)**. *exact recovery* in many applications is a more interesting problem. On the other hand, this approach seems to be adopted by most publications in the field – perhaps due to the numerical difficulty to verify the most sparsity.

### 2.6.3 Other Numerical Approaches

The work of Couvreur and Bresler [23] on *backward elimination* and related analysis has strong similarity with some of the results that we developed here for MMV.

Short papers [55, 54] proposed various heuristics to achieve sparse representations . They give a flavor on algorithms that have been adopted in signal processing.

### 2.6.4 Probability, Random Matrices

Recently, in the case of SMV, some very inspiring new results are obtained. Recall in SMV, we have $b = Ax_0$, where $A \in \mathbb{R}^{m\times n}, x_0 \in \mathbb{R}^n$, and $b \in \mathbb{R}^m$, $m < n$. Donoho in [29] shows that when $\|x_0\|_0 = O(n)$ and the matrix $A$ is random, with the probability nearly equal to 1, the minimizing $\ell_1$ norm approach (i.e., **(Q1)**) gives the solution being equal to $x_0$.

In general, the upper bounds that are given in this chapter are lower than $O(n)$. The cases that are considered here are the *worst cases*. It is shown that these worst-case results are extremely *conservative.*

A similar result regarding noisy data was reported in [28]. At the same time, E. Candès gave several talks with similar results, based on his joint work with T. Tao and J. Romberg [9].

There are interesting developments in the random matrix. Recall that the *mutual incoherence*, $M$, has been used in several upper bounds of underlying sparsity, for both uniqueness and equivalence. Roughly, the upper bounds are $\sim M^{-1}/2$. Historically,

it is of particular interest to study the case when matrix $A$ is a concatenation of two orthogonal square matrices: $A = [O_1, O_2]$, where matrices $O_1, O_2$ are orthogonal. Apparently the *mutual incoherence* is the maximum magnitude of the entries of matrix $O_1^T O_2$. Jiang in [64] derived the asymptotic distribution of this quantity. Basically, if $O_1 \in \mathbb{R}^{n\times n}$, he proved that $M^{-1}/2$ is *almost surely* between $\frac{1}{2\sqrt{6}}\sqrt{\frac{n}{\log n}}$ and $\frac{1}{4}\sqrt{\frac{n}{\log n}}$.

In another work of Jiang [63], the limit distribution of the maximal off diagonal entry in a correlation matrix was derived. It can have similar applications as the above result in analyzing the behavior of $M^{-1}/2$ in other scenarios.

We would like to point out that the worst case analysis (which eventually produces $M^{-1}/2$) is not powerful enough to produce the probabilistic results that are stated at the beginning of this subsection.

### 2.6.5 Related Publications

Some preliminary results in this chapter was reported in a conference paper [11] and a manuscript [12]. The latter was downloadable online. This chapter is an extensive revision of [12].

## *2.7 Conclusion*

We showed that most of the results on sparse representations of *simple measurement vectors* can be generalized to the case of *multiple measurement vectors*. Our generalization is broad: the inside norm $m(\cdot)$ in **(P0)** and **(P1)** can be *any* vector norm.

When additional information is available in *multiple measurement vectors,* better upper bounds for uniqueness in **(P0)** (and hopefully for equivalence, referring to our discussion) become possible. An incarnation of this is Theorem 2.2.4.

We showed that a greedy algorithm – OMP – under certain conditions, can achieve the sparsest representation, just like the result in SMV. We realize that the generalization can be achieved in a broad sense; more specifically, the inner vector norm in

the step 2)-a) of OMPMMV can be $\ell_q$ norm for any $q \geq 1$.

These results provide useful insights in designing numerical solutions to find the sparse representations of *multiple measurement vectors.*

# CHAPTER III

# HARDNESS ON NUMERICAL REALIZATION OF SOME PENALIZED LIKELIHOOD ESTIMATORS

## *3.1 Introduction*

Penalized least squares estimators are well presented and advocated in statistics, see, e.g., [38, 1, 39]. We show that for several existing types of penalty functions, the corresponding penalized least squares problems are equivalent to the exact cover by 3-sets problem, which belongs to an NP-hard class [47]. No polynomial-time numerical solution is available by now. We then extend the NP-hardness results to penalized least absolute deviation regression and a problem that is derived from penalized support vector machines. The proof of NP-hardness is preceded by [85] and some recent works [62, 86]. However, the proofs in this chapter require much more involved techniques.

Our NP-hardness result does not oppose the principle of penalized likelihood estimators. Instead, our results forewarn a misuse of penalized likelihood estimators: i.e., one should not attempt to find the *global* extremum(a) in numerical implementations. It was proven by Fan and Li [38] that the oracle property of a penalized likelihood estimator just requires a *local* extremum. The correct way to utilizing penalized likelihood estimator is the following: starting with a consistent estimator (e.g., the maximum likelihood estimator), then modifying it via optimizing the penalized likelihood function locally.

The rest of the chapter is organized as follows. Section 3.2 presents a formulation of the penalized least squares estimation. Section 3.3 describes some well-adopted

penalty functions and known NP-hardness results. Section 3.4 establishes the NP-hardness for penalized least squares estimators in a general way. Specific results regarding each class of penalty functions are given as corollaries. Section 3.5 extends the NP-hardness to regression with least absolute deviations. Section 3.6 extends the NP-hardness to penalized support vector machines. Section 3.7 discuss other possible penalized likelihood estimators and conjecture on their NP-hardness. Section 3.8 gives some literature points on the oracle property of penalized likelihood estimators, emphasizing their requirement of local extremum (instead of a global extremum). We discuss the practical implication of the NP-hardness of the penalized likelihood estimators in Section 3.8.

## *3.2 Problem Formulation*

Consider linear regression model, $y = \Phi x + \varepsilon$, where $y \in \mathbb{R}^m, \Phi \in \mathbb{R}^{m\times n}, x \in \mathbb{R}^n$, and $\varepsilon \in \mathbb{R}^m$. Vectors $y, x$, and $\varepsilon$ are called responses, coefficients, and random errors respectively. Matrix $\Phi$ is called the model matrix. The penalized least squares estimator is the solution to the following optimization problem:

$$\text{(PLS)} \qquad \min_x \quad \|y - \Phi x\|_2^2 + \lambda_0 \sum_{i=1}^{n} p(|x_i|),$$

where term $\|\cdot\|_2^2$ corresponds to the residual sum of squares, $\lambda_0$ is a prescribed algorithmic parameter, penalty function $p(\cdot)$ maps nonnegative value to nonnegative value ($p : \mathbb{R}^+ \to \mathbb{R}^+$), and $x_i$ is the $i$th entry of the coefficient vector $x$.

## *3.3 Penalty Functions and Known NP-hardness Results*

Some choices of $p$ are listed below. We always assume $x \geq 0$.

- $\ell_0$ penalty: $p(x) = I(x \neq 0)$, where $I(\cdot)$ is the indicator function.
- $\ell_1$ penalty: $p(x) = |x|$; Lasso [101] and its variants utilize this penalty function.
- Ridge regression: $p(x) = x^2$.

- More generally, for $0 < c < 2$, bridge regression [45] takes $p(x) = x^c$.
- Hard-threshold penalty [35]: $p(x;\lambda) = \lambda^2 - [(\lambda - |x|)_+]^2$, where $\lambda$ ($\lambda > 0$) is another algorithmic parameter. This penalty function is smoother than the $\ell_0$ penalty function.
- Nikolova penalty [87]: $p(x) = \frac{x}{1+x}$.
- Finally, the smoothly clipped absolute deviation (SCAD) penalty [38]: for $\lambda > 0, a > 1$,

$$p(x) = \begin{cases} \lambda x, & \text{if } 0 \le x < \lambda; \\ -(x^2 - 2a\lambda x + \lambda^2)/[2(a-1)], & \text{if } \lambda \le x < a\lambda; \\ (a+1)\lambda^2/2, & \text{if } x \ge a\lambda. \end{cases}$$

As one can see, penalized least squares covers many problems in model selection and estimation. It is well-known that when the model matrix $\Phi$ is orthogonal, the solutions to the above problems are trivial: just apply some univariate operators. It is shown in [62, 86] that for generic model matrix $\Phi$, when the $\ell_0$ penalty is chosen, the problem is NP-hard. The proof of Huo and Ni [62] utilizes the result of Natarajan [85], which states that sparse approximate solutions (SAS) to linear system are equivalent to the exact cover by 3-sets (X3C) problem, which is known to be NP-hard [47]. Huo and Ni [62] apply the principle of Lagrange multiplier to establish the relation between the $\ell_0$ penalized PLS and SAS; hence prove the NP-hardness. It remained open whether a more general PLS is NP-hard.

## *3.4 General NP-Hardness for PLS Estimators*

In this chapter, we establish the following theorem.

**Theorem 3.4.1 (NP-Hardness of PLS)** *For general model matrix $\Phi$, problem (PLS) is NP-hard if the penalty function $p(\cdot)$ ($p : \mathbb{R}^+ \to \mathbb{R}^+$) satisfies the following four conditions.*

*C1.* $p(0) = 0$ *and function* $p(x), x \geq 0$, *is monotone increasing:* $\forall 0 \leq x_1 < x_2, p(x_1) \leq p(x_2)$.

*C2. There exists* $\tau_0 > 0$ *and a constant* $c > 0$, *such that* $\forall 0 \leq x < \tau_0$, *we have*

$$p(x) \geq p(\tau_0) - c(\tau_0 - x)^2.$$

*C3. For the aforementioned* $\tau_0$, *if* $x_1, x_2 < \tau_0$, *then* $p(x_1) + p(x_2) \geq p(x_1 + x_2)$.

*C4.* $\forall 0 \leq x < \tau_0, p(x) + p(\tau_0 - x) > p(\tau_0)$.

A proof is given in Section 3.9.1. Note in most cases of PLS, we have $p(0) = 0$ and function $p(\cdot)$ is monotone increasing. Hence C1 is satisfied. Condition C3 is satisfied if function $p(x)$ is concave in $[0, 2\tau_0]$. Condition C4 holds if function $p(x)$ is strictly concave for point 0 and point $\tau_0$. See Section 3.9.2 for a brief justification.

For the penalty functions in $\ell_0$ penalty, bridge regression with $0 < c < 1$, Hard-threshold, Nikolova penalty, and SCAD, one can easily see that C3 and C4 hold.

Condition C2 is less intuitive. However, it is important to ensure the NP-hardness. Recall that in Fan and Li [38], $p'(|x|) = 0$ for large $|x|$ is a sufficient condition for the unbiasedness of a PLS estimator. On the other hand, if $p'(|x|) = 0$ for $|x|$ larger than a positive value, it is possible to find a quadratic function, $y = p(\tau_0) - c(\tau_0 - x)^2$, which is below penalty function $y = p(x)$ for $0 \leq x < \tau_0$ with positive constants $\tau_0$ and $c$. For $\ell_0$, hard-threshold, and SCAD penalties, one can verify C2 with the following values for $\tau_0$ and $c$.

- For the $\ell_0$ penalty, one takes $\tau_0 = 1$ and $c = 1$.
- For the hard-threshold penalty, one may take $\tau_0 = \lambda$ and $c = 1$.
- For the SCAD penalty, one takes $\tau_0 = a\lambda$ and $c = \frac{1}{2(a-1)}$.

From the above, we immediately have the following.

**Corollary 3.4.2** *For the penalties in $\ell_0$, hard-threshold, and SCAD, the implementation of PLS lead to NP-hard problems.*

Note that the result of NP-hardness in Huo and Ni [62] becomes a special case.

It is well known that the $\ell_1$ penalty leads to linear programming problems. It is also well know that ridge regression and bridge regression with $c \geq 1$ lead to convex optimization problems, hence they have polynomial time solutions.

Solely based on Theorem 3.4.1, we can not establish the NP-hardness for the PLS problem with Nikolova penalty function or bridge regression with $0 < c < 1$. One can not establish C2 for the Nikolova penalty; neither can we for the bridge regression with $0 < c < 1$. The derivatives of both penalty functions converge to zero as the variable goes to the positive infinity. In Section 3.9.3, it is shown that if C2 holds, then $p'(\tau_0) = 0$.

For bridge regression with $0 < c < 1$, we conjecture that the related PLS problem is NP-hard. However, we do not establish a proof in the present chapter. In part, it is found that $p'(x) = c \cdot x^{c-1} \to +\infty$, as $x \to 0$. In all the cases that we have proved so far, we have $p'(x)$ upper bounded in interval $[0, \tau_0)$. At the same time, the gradient of the objective function in (PLS) becomes instable (going to infinities) as some elements of $x$ converge to 0. Hence we believe it is hard to numerically realize a bridge regression with $0 < c < 1$. Furthermore, it is shown in [1] and [38] that such a bridge regression is *not* continuous.

The following theorem will be used to establish the NP-hardness related to the Nikolova penalty. A proof is given in Section 3.9.4.

**Theorem 3.4.3** *Assume the model matrix $\Phi$ is of full row rank. For continuous penalty function $p(x)$ that satisfies condition C1 and is strictly concave within interval $(0, \infty)$. Suppose penalty function $p(x)$ satisfies the Lipschitz condition: there exists a constant $C_1 > 0$ such that $|p(x_1) - p(x_2)| \leq C_1|x_1 - x_2|$ for any $0 < x_1, x_2 < \infty$. Then the corresponding PLS problem is NP-hard.*

For the PLS estimators with Nikolova penalty, it is observed that

$$p'(x) = (1+x)^{-2} \Rightarrow 0 < p'(x) \leq 1, \forall x \in [0, +\infty).$$

Evidently, the Lipschitz condition holds. We immediately have the following.

**Corollary 3.4.4** *Assume the model matrix $\Phi$ is of full row rank. For a PLS estimator with Nikolova penalty, the corresponding optimization problem is NP-hard.*

## *3.5 Least Absolute Deviation Regression*

It is interesting to know that when the quadratic term $\|y - \Phi x\|_2^2$ in (PLS) is replaced by a sum of the absolute values of the residuals (i.e., $\|y - \Phi x\|_1$), for several penalty functions, the corresponding optimization problems are NP-hard. The proof of NP-hardness is nearly identical with the proof of Theorem 3.4.1. Recall that these problems are associated with the least absolute deviations (LAD) regression [48, 4].

We consider

$$\text{(PLAD)} \qquad \min_x \quad \|y - \Phi x\|_1 + \lambda_0 \sum_{i=1}^{n} p(|x_i|),$$

where all the notations are predefined. We have the following theorem.

**Theorem 3.5.1 (NP-hardness for Penalized LAD)** *For general model matrix $\Phi$, problem (PLAD) is NP-hard if the penalty function $p(\cdot)$ ($p : \mathbb{R}^+ \to \mathbb{R}^+$) satisfies the following three conditions:*

*D1.* $p(0) = 0$ *and function* $p(x), x \geq 0$, *is monotone increasing:* $\forall 0 \leq x_1 < x_2, p(x_1) \leq p(x_2)$.

*D2. There exists a constant $\tau_0 > 0$, such that function $p(x)$ is concave in the interval $[0, 2\tau_0]$.*

*D3.* $\forall 0 \leq x < \tau_0, p(x) + p(\tau_0 - x) > p(\tau_0)$.

It is easy to verify that for the function $p(x)$ in $\ell_0$ penalty, bridge regression with $0 < c < 1$, hard-threshold, Nikolova penalty, and SCAD, the conditions in the above theorem are satisfied. Hence we immediately have the following.

**Corollary 3.5.2** *If the penalty function is chosen according to the $\ell_0$ penalty, bridge regression with $0 < c < 1$, hard-threshold, Nikolova penalty, or SCAD, the resulting problem as in (PLAD) is NP-hard.*

We explain why the proof of Theorem 3.5.1 will be an easy extension from the proof of Theorem 3.4.1. First of all, note that conditions D1 and D3 in Theorem 3.5.1 are identical with the conditions C1 and C4 in Theorem 3.4.1. Moreover, given D2, it is easy to verify that a condition like C3 is satisfied, referring to the discussion in Section 3.9.2. Finally, given D2, for $0 < x < \tau_0$, we have

$$\begin{aligned} p(x) &\geq \frac{x}{\tau_0}p(\tau_0) + \frac{\tau_0 - x}{\tau_0}p(0) \\ &= \frac{x}{\tau_0}p(\tau_0) \\ &= p(\tau_0) - \frac{p(\tau_0)}{\tau_0}(\tau_0 - x); \end{aligned}$$

i.e., a condition like C2 is satisfied. As an exercise, readers can verify that the proof of Theorem 3.4.1 in Section 3.9.1 can be modified to prove the Theorem 3.5.1.

## *3.6 A Problem Related to Machine Learning and Data Mining*

The following problem is rooted in machine learning and data mining [39, Section 6.4]. We consider

$$\text{(PSVM)} \qquad \min_{\beta} \quad \sum_{i=1}^{n}[1 - y_i(\mathbf{x}_i^T\beta)]_+ + \lambda_0 \sum_{j=1}^{d} p(|\beta_j|),$$

where $(\mathbf{x}_i, y_i), \mathbf{x}_i \in \mathbb{R}^d, y_i \in \{-1, 1\}, i = 1, 2, \ldots, n$, are training data; coefficient vector $\beta \in \mathbb{R}^d$ has elements $\beta_j, j = 1, 2, \ldots, d$; function $[\cdot]_+$ corresponds to the hinge

loss and only takes nonnegative value:

$$[x]_+ = \begin{cases} x, & \text{if } x \geq 0, \\ 0, & \text{if } x < 0; \end{cases}$$

constant $\lambda_0$ is an algorithmic parameter; function $p(\cdot)$ is the previously mentioned penalty function. In 1-norm support vector machine ([116] and references therein), we have $p(\beta) = |\beta|$; while in ordinary support vector machine, we have $p(\beta) = \beta^2$.

We will show that for a class of penalty function $p(\cdot)$, the problem (PSVM) is NP-hard. The proof of this NP-hardness result bears strong similarity with the proof of Theorem 3.4.1. However, it is not a direct extension. In the proof, several steps require slightly different treatment. We provide a complete proof in Section 3.9.5.

**Theorem 3.6.1 (Penalized Support Vector Machines)** *For a general training data* $(\mathbf{x}_i, y_i), i = 1, 2, \ldots, n$, *the problem (PSVM) is NP-hard if there exists a constant* $\tau_0 > 0$, *such that*

$$\lambda_0 \leq 3/p(\tau_0) \tag{6.16}$$

*and (for this* $\tau_0$*) the penalty function* $p(\cdot)$ $(p : \mathbb{R}^+ \to \mathbb{R}^+)$ *satisfies the three conditions (D1-D3) in Theorem 3.5.1.*

We now discuss when we can apply the above theorem. Recall the penalty functions that are described in Section 3.3.

- For the $\ell_0$ penalty, due to the concave condition D3, we must choose $\tau_0 > 0$. Hence $p(\tau_0) \equiv 1$. Hence the above theorem only applies when $\lambda_0 \leq 3$.

- For the bridge regression with $0 < c < 1$, we can choose $\tau_0$ to be any value in interval $(0, \infty)$. Hence $p(\tau_0)$ takes any value between 0 and $+\infty$. Hence Theorem 3.6.1 applies for any $\lambda_0 > 0$.

- For hard-threshold, one can choose $\tau_0 > 0$. Because $p(\tau_0)$ takes any value in interval $(0, \lambda^2)$, the above theorem applies for any $\lambda_0 > 0$.

- For the Nikolova penalty, one can choose $\tau_0 > 0$. The possible values of $p(\tau_0)$ form interval $(0, 1)$. Hence Theorem 3.6.1 applies for any $\lambda_0 > 0$.

- For SCAD, we must choose $\tau_0 > \lambda$. We have $\lambda^2 < p(\tau_0) \leq \frac{a+1}{2}\lambda^2$. Hence Theorem 3.6.1 applies when $\lambda_0 < 3/\lambda^2$.

**Corollary 3.6.2** *For a penalty function and its corresponding domain of the parameter $\lambda_0$ that is specified in the foregoing list, the problem (PSVM) is NP-hard.*

## *3.7 Other Penalized Likelihood Estimators*

One can see that when the penalty functions, $p_{\lambda_{t,j}}(\cdot)$, are identical, the sparse covariance matrix estimation problem [39, Equation (6.7)] is a PLS problem. Hence the NP-hardness result for PLS applies.

We have considered a subset of penalized likelihood estimators. There are other penalized likelihood estimators, e.g., a penalized likelihood estimator in logistic regression [39, Example 1] and a penalized likelihood estimator in Poisson log-linear regression [39, Example 2]. We conjecture that the corresponding optimization problems are NP-hard. However, this chapter does not provide a proof.

## *3.8 Oracle Property and Local Minimizers*

It would be wrong to think that the NP-hardness results in this chapter are against the use of penalized likelihood estimators. In fact, in [38], the authors have pointed out that the corresponding optimization problems are hard. Moreover, authors of [38] showed that a *local* extremum of the penalized likelihood possess the oracle property. The oracle property is one of the most interesting theoretical properties. The NP-hardness results of this chapter simply demonstrate that it will be a misinterpretation of the penalized likelihood principle if someone tries to find the global extremum(a). The computational issue of penalized likelihood estimators are further discussed in

[58], in which the authors correctly pointed out the difficulty of optimizing the penalized likelihood function.

The account of oracle property (for a local minimizer) in [38] is inspiring. Further technical analysis is reported in [40]. Another recent discovery on the oracle property of Lasso (in fact, a modified version) is reported in [117]. They have a common flavor: starting from a consistent estimator, adjust the estimator by either optimizing the penalized likelihood function locally or modifying the weights in the penalty function of Lasso. It will be interesting to exploit the connection between these approaches. In particular, can we provide a unified condition on when such a local adjustment will lead to oracle property? We leave it as a topic of future research.

## *3.9 Proofs Associated with Chapter III*

### 3.9.1 Proof of Theorem 3.4.1

It is known that the exact cover by 3-sets (X3C) is NP-hard [47]. Let $S$ denote a set with $m$ elements. Let $C$ denote a collection of 3-element subsets of $S$. The X3C is [85]: Does $C$ contain an exact cover for $S$; i.e., a subcollection $\hat{C}$ of $C$ such that every element of $S$ occurs exactly once in $\hat{C}$. Without loss of generality, we assume that $m$ is divisible by 3; otherwise X3C can never be done.

Let $f(x)$ denote the objective function in (PLS); i.e., $f(x) = \|y-\Phi x\|_2^2+\lambda_0 \sum_{i=1}^n p(|x_i|)$. We will show that for a pair of $(\Phi, y)$, there exists a constant $M$, such that $f(x) \leq M$ if and only if there is a solution to X3C. Hence if (PLS) is not NP-hard, then X3C is not either; such a contradiction leads to the NP-hardness of (PLS).

We now construct $\Phi$ and $y$. The number of columns in $\Phi$ is equal to the number of subsets in $C$. Let $\phi_j$ $(1 \leq j \leq |C|)$ denote the $j$th column of the matrix $\Phi$. We assign: for $1 \leq i \leq m$, $(\phi_j)_i = \sqrt{c(\lambda_0+1)/3}$ if the $i$th element of $S$ appears in the $j$th 3-subset in $C$; and $(\phi_j)_i = 0$, otherwise. Here $(\phi_j)_i$ is the $i$th entry of vector $\phi_j$. Apparently, we have $n = |C|$, the size of $C$. Let $y = \tau_0\sqrt{c(\lambda_0+1)/3} \cdot \mathbf{1}_{m\times 1}$, where

$\mathbf{1}_{m\times 1}$ is an all-one vector.

Suppose X3C has a solution. We create vector $x^*$ as: $(x^*)_i = \tau_0$, if the $i$th 3-element subset in $C$ is used in the solution to X3C; and $(x^*)_i = 0$, otherwise. Here $(x^*)_i$ denotes the $i$th entry of vector $x^*$. One can easily verify that $y = \Phi x^*$. Hence we have $f(x^*) = \frac{m}{3}\lambda_0 p(\tau_0)$.

Now assign $M = \frac{m}{3}\lambda_0 p(\tau_0)$. We show that if there exists $x'$ satisfying $f(x') \leq M$, then we must have $x'$ to be a solution of the X3C problem. Recalling that $x^*$ corresponds to a solution to X3C as described above, if the solution to the X3C problem is not unique, it is possible that $x'$ and $x^*$ are not identical.

For $1 \leq k \leq m$, Let $\Omega_k$ denote a set of indices (of $C$) corresponding to the nonzero entries in the $k$th row of matrix $\Phi$. Given $y = \Phi x^*$, it is evident that there is exactly one $j \in \Omega_k$, such that $x^*_j = \tau_0$; while for other $j \in \Omega_k$, we should have $x^*_j = 0$. We will need the following lemma.

**Lemma 3.9.1** *Suppose $p(\cdot)$ satisfies condition C1-C4. For $1 \leq k \leq m$, the following strict inequality holds if at least one side of it is not equal to zero:*

$$\frac{1}{3}\sum_{i\in\Omega_k}[p(|x^*_i|) - p(|x'_i|)] < \lambda_0^{-1}\cdot\frac{(\lambda_0+1)c}{3}\left[\sum_{i\in\Omega_k}(x^*_i - x'_i)\right]^2. \tag{9.17}$$

Before giving the proof of the above lemma, we first introduce the following lemma which will be used in the proof of Lemma 3.9.1 and the proofs of other theorems in the chapter.

**Lemma 3.9.2** *If $\tau_0 \leq \sum_{i\in\Omega_k}|x'_i|$ and $|\Omega_k| > 1$ , we have $p(\tau_0) < \sum_{i\in\Omega_k} p(|x'_i|)$.*

**Proof of Lemma 3.9.2.** Let $c_1 = \sum_{i\in\Omega_k}|x'_i| \geq \tau_0$, we have

$$p(\tau_0) = p(\frac{\tau_0}{c_1}\sum_{i\in\Omega_k}|x'_i|) \overset{C3,C4}{<} \sum_{i\in\Omega_k} p(\frac{\tau_0}{c_1}|x'_i|) \leq \sum_{i\in\Omega_k} p(|x'_i|).$$

Hence, Lemma 3.9.2 holds. □

**Proof of Lemma 3.9.1.** We prove this lemma in two cases.

In the first case, we suppose that the right hand side of (9.17) is nonzero. Hence the right hand side is always positive. The (9.17) holds trivially if the left hand side is nonpositive. If the left hand side of (9.17) is positive, we have

$$\begin{aligned}
\frac{\frac{1}{3}\sum_{i\in\Omega_k}[p(|x_i^*|)-p(|x_i'|)]}{\frac{(\lambda_0+1)c}{3}[\sum_{i\in\Omega_k}(x_i^*-x_i')]^2} &= \frac{p(\tau_0)-\sum_{i\in\Omega_k}p(|x_i'|)}{(\lambda_0+1)c(\tau_0-\sum_{i\in\Omega_k}x_i')^2} \\
&\overset{C3}{\leq} \frac{p(\tau_0)-p(\sum_{i\in\Omega_k}|x_i'|)}{(\lambda_0+1)c(\tau_0-\sum_{i\in\Omega_k}x_i')^2} \\
&\leq \frac{p(\tau_0)-p(\sum_{i\in\Omega_k}|x_i'|)}{(\lambda_0+1)c(\tau_0-\sum_{i\in\Omega_k}|x_i'|)^2} \\
&\overset{C2}{\leq} (1+\lambda_0)^{-1}<\lambda_0^{-1}.
\end{aligned}$$

Note in all the three inequalities, we implicitly utilizes

$$p(\tau_0) > \sum_{i\in\Omega_k} p(|x_i'|). \tag{9.18}$$

By using Lemma 3.9.2, we have $\tau_0 > \sum_{i\in\Omega_k}|x_i'|$. Then the first equality holds by condition C3, and the last inequality holds by condition C2.

In the second case, we assume that the right hand side of (9.17) is zero, we have $\tau_0 = \sum_{i\in\Omega_k} x_i'$. From the assumption of our lemma, the left hand side can *not* be zero simultaneously. Condition C1 and Lemma 3.9.2 demonstrate that the left hand side can only be negative: first, we have

$$p(\tau_0) = p(|\sum_{i\in\Omega_k} x_i'|) \overset{C1}{\leq} p(\sum_{i\in\Omega_k}|x_i'|);$$

Thus, $\tau_0 \leq \sum_{i\in\Omega_k}|x_i'|$. With Lemma 3.9.2, it is easy to see (9.17) holds. □

Given the definition of $\Omega_k$, it is not hard to see that

$$\sum_{i=1}^{n}[p(|x_i^*|)-p(|x_i'|)] = \sum_{k=1}^{m}\frac{1}{3}\sum_{i\in\Omega_k}[p(|x_i^*|)-p(|x_i'|)]. \tag{9.19}$$

From the construction of $\Phi$, it is not hard to verify the following:

$$\begin{aligned} \|y - \Phi x'\|_2^2 &= \|\Phi x^* - \Phi x'\|_2^2 \\ &= \|\Phi(x^* - x')\|_2^2 \\ &= \sum_{k=1}^{m} \frac{(\lambda_0 + 1)c}{3} [\sum_{i \in \Omega_k} (x_i^* - x_i')]^2. \end{aligned} \tag{9.20}$$

Combine Lemma 3.9.1 with (9.19) and (9.20), we have

$$\sum_{i=1}^{n} p(|x_i^*|) - \sum_{i=1}^{n} p(|x_i'|) \leq \lambda_0^{-1} \|y - \Phi x'\|_2^2;$$

and the equality holds if and only if the two sides of (9.17) are equal to zero for every $k$, $1 \leq k \leq m$. Note the above is equivalent to $M = f(x^*) \leq f(x')$. Recall $f(x') \leq M$, we must have $f(x') = M$ and $\forall k, p(\tau_0) = \sum_{i \in \Omega_k} p(|x_i'|)$ and $\tau_0 = \sum_{i \in \Omega_k} x_i'$. Utilizing Lemma 3.9.2, one can show that within set $\{x_i' : i \in \Omega_k\}$, we must have exactly one element that is equal to $\tau_0$ and the rest are zeros. Given the design of $x^*$, it is not hard to see that $x'$ corresponds to another solution to X3C. (Note the solutions to X3C is not necessarily unique.)

From all the above, the theorem is proved.

### 3.9.2 Justifications Related to C3 and C4

Recall function $p(x)$ is concave in $[0, 2\tau_0]$. Hence for $x_1, x_2 < \tau_0$, we have, for $0 \leq \lambda \leq 1$,

$$p[\lambda x_1 + (1 - \lambda) x_2] \geq \lambda p(x_1) + (1 - \lambda) p(x_2),$$

Therefore, we have

$$p(x_i) \geq \frac{x_i}{x_1 + x_2} p(x_1 + x_2) + \frac{x_{\{3-i\}}}{x_1 + x_2} p(0), \quad i = 1, 2.$$

Adding the above two and using $p(0) = 0$, we have

$$p(x_1) + p(x_2) \geq p(x_1 + x_2).$$

The above is condition C3. The justification regarding C4 is nearly identical.

### 3.9.3 Proof of "First Derivative is Zero"

Recall $p(x), x \geq 0$, is nondecreasing; hence $p'(x) \geq 0$. On the other hand, it is obvious that when C2 holds, we have

$$f(x) \stackrel{\Delta}{=} p(x) - p(\tau_0) + c(\tau_0 - x)^2 \geq 0, \text{ for } 0 \leq x \leq \tau_0,$$

and $f(\tau_0) = 0$; hence $f'(\tau_0) \leq 0$, which leads to $p'(\tau_0) \leq 0$. From all the above, we have that $p'(\tau_0) = 0$.

### 3.9.4 Proof of Theorem 3.4.3

Let

$$f(x) = \|y - \Phi x\|_2^2 + \lambda_0 \sum_{i=1}^{n} p(|x_i|),$$

where $p(x)$ is the penalty function as in (PLS). Consider a truncated version of the penalty function:

$$p_t(x; N) = \begin{cases} p(x), & 0 \leq x \leq N, \\ p(N), & x > N, \end{cases}$$

where $N$ is a positive constant. Correspondingly, we define

$$f_t(x; N) = \|y - \Phi x\|_2^2 + \lambda_0 \sum_{i=1}^{n} p_t(|x_i|; N).$$

Minimizing the $f(x)$ in $\mathbb{R}^n$ is the original PLS optimization problem; while minimizing $f(x; N)$ with a prefixed $N$ is its truncated version. Applying Theorem 3.4.1, it is not hard to see that the latter is NP-hard. We omit some obvious details here.

If we can show that for a constant $N'$ that is large enough, the two problems have identical solutions, then we prove that the PLS problem with the original penalty is NP-hard.

We will show below that the solutions to the aforementioned problems are upper bounded by a constant; hence by choosing $N'$ as this upper bound, the two problems are identical.

Let $x_0$ be the minimizer of either objective $f(x)$ or $f_t(x;N)$. Without loss of generality, assume $x_0$ is the minimizer of $f_t(x;N)$. Note the same argument will apply to objective $f(x)$ as well. We have $\forall a \in \mathbb{R}^n$,

$$f_t(x_0 + a; N) \geq f_t(x_0; N).$$

From the definition of $f_t(\cdot; N)$, we have

$$\|y - \Phi(x_0 + a)\|_2^2 + \lambda_0 \sum_{i=1}^{n} p_t[|(x_0 + a)_i|; N] \geq \|y - \Phi x_0\|_2^2 + \lambda_0 \sum_{i=1}^{n} p_t[|(x_0)_i|; N],$$

where $(\cdot)_i$ denotes the $i$th element of a vector. Simplifying the above, we have

$$\begin{aligned} a^T\Phi^T\Phi a + 2(x_0^T\Phi^T\Phi - y^T\Phi)a \\ +\lambda_0 \sum_{i=1}^{n} \{p_t[|(x_0 + a)_i|; N] - p_t[|(x_0)_i|; N]\} \quad \geq \quad 0. \end{aligned} \tag{9.21}$$

We will need the following inequality, which is presented in a lemma.

**Lemma 3.9.3** *For $1 \leq i \leq n$, we have*

$$|(\Phi^T\Phi x_0 - \Phi^T y)_i| \leq \frac{1}{2}\lambda_0 C_1,$$

*where $C_1$ is the Lipschitz constant that is given in the theorem statement.*

**Proof.** For $1 \leq i \leq n$, within vector $a$, we set every entry except $a_i$ to be equal to 0. From (9.21), we have $\forall a_i$,

$$T_1 a_i^2 + 2T_2 a_i + \lambda_0[p_t(|T_3 + a_i|; N) - p_t(|T_3|; N)] \geq 0, \tag{9.22}$$

where $T_1 = (\Phi^T\Phi)_{ii}, T_2 = (\Phi^T\Phi x_0 - \Phi^T y)_i$, and $T_3 = (x_0)_i$. Without loss of generality, in the following argument, we assume that $a_i > 0$. Readers can verify that a trivially modified argument holds when $a_i < 0$. Replacing $a_i$ with $-a_i$ in (9.22), we have

$$T_1 a_i^2 - 2T_2 a_i + \lambda_0[p_t(|T_3 - a_i|; N) - p_t(|T_3|; N)] \geq 0. \tag{9.23}$$

Given the definition of $p_t(\cdot, N)$ and the Lipschitz property of $p(\cdot)$, it is easy to see that function $p_t(\cdot; N)$ also satisfies the Lipschitz condition: for $a_i \neq 0$,

$$\left| \frac{p_t(|\alpha + a_i|; N) - p_t(|\alpha|; N)}{a_i} \right| < C_1,$$

where $\alpha$ is an arbitrary real number. From (9.22), we have

$$\begin{aligned} T_2 &\geq -\frac{1}{2} T_1 a_i - \frac{1}{2} \lambda_0 \frac{p_t(|T_3 + a_i|; N) - p_t(|T_3|; N)}{a_i} \\ &\geq -\frac{1}{2} T_1 a_i - \frac{1}{2} \lambda_0 C_1. \end{aligned} \tag{9.24}$$

Similarly from (9.23), we have

$$\begin{aligned} T_2 &\leq \frac{1}{2} T_1 a_i + \frac{1}{2} \lambda_0 \frac{p_t(|T_3 - a_i|; N) - p_t(|T_3|; N)}{a_i} \\ &\leq \frac{1}{2} T_1 a_i + \frac{1}{2} \lambda_0 C_1. \end{aligned} \tag{9.25}$$

Letting $a_i \to 0$, combining (9.24) and (9.25), we have $|T_2| \leq \frac{1}{2} \lambda_0 C_1$. □

Let $v = \Phi^T \Phi x_0 - \Phi^T y$, the above leads to the following:

$$\begin{aligned} \|x_0\|_\infty &\leq \|x_0\|_2 \\ &\leq \|(\Phi^T \Phi)^{-1} \Phi^T y\|_2 + \sup_{\|v\|_\infty < \frac{1}{2} \lambda_0 C_1} \|(\Phi^T \Phi)^{-1} v\|_2 \\ &\leq \|(\Phi^T \Phi)^{-1} \Phi^T y\|_2 + \frac{\sqrt{n} \frac{1}{2} \lambda_0 C_1}{\mu_{\min}(\Phi^T \Phi)}. \end{aligned}$$

where $\mu_{\min}(\cdot)$ is the smallest eigenvalue of the matrix. Note the last term is a constant, which is determined by $\Phi$, $y$, $\lambda_0$, and $C_1$. By taking $N'$ to be the above constant, the above establishes the equivalence between the PLS problem with the original penalty function and the PLS problem with the truncated penalty function. Because the PLS problem with the truncated penalty function is NP-hard, we conclude that the PLS problem with the original penalty function is NP-hard as well. The theorem is proved.

### 3.9.5 Proof of Theorem 3.6.1

Similar to the proof in Section 3.9.1, we will show that if problem (PSVM) is *not* NP-hard, neither is the exact cover by 3-sets problem. (Recall that the exact cover by 3-sets problem is denoted by X3C.) Because we know that X3C is NP-hard, so is the (PSVM). The proof is again constructive.

Let matrix $\Phi = \mathrm{diag}(y_1, y_2, \ldots, y_n)\mathbf{X}$, where $\mathrm{diag}(y_1, y_2, \ldots, y_n)$ is a diagonal matrix with diagonal entries $y_1, y_2, \ldots, y_n$ and that

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{pmatrix}.$$

Note $\Phi \in \mathbb{R}^{n\times d}$. For any given matrix $\Phi$, one can find a (non-unique) set of data $(\mathbf{x}_i, y_i), i = 1, 2, \ldots, n$, such that the above holds.

Let $f(\beta) = \|\mathbf{1}_n - \Phi\beta\|_+ + \lambda_0 \sum_{j=1}^d p(|\beta_j|)$, where $\mathbf{1}_n \in \mathbb{R}^{n\times 1}$ is an all-one vector, and for vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$, we have $\|\mathbf{x}\|_+ = \sum_{i=1}^n (x_i)_+$. It is evident that problem (PSVM) is equivalent to

$$\min_{\beta} \quad f(\beta). \tag{9.26}$$

We now construct a matrix $\Phi$. Let $S$ be a set with $n$ elements. Without loss of generality, we assume that $n$ is divisible by 3. Let $C$ denotes a collection of 3-element subsets of $S$. (Recall that the $S$ and $C$ are used in Section 3.9.1.) Each column of matrix $\Phi$ (denoted by $\phi_k, 1 \le k \le |C|$) corresponds to a subset in collection $C$. Moreover, we have $(\phi_k)_i = 1/\tau_0$ if and only if the $i$th element of $S$ is in the $k$th subset of $C$. Assume $\widehat{C} \subset C$ corresponds to an exact cover of $S$ by 3-sets. For vector $\beta^* \in \mathbb{R}^{|C|}$, we have $\beta_k^* = \tau_0$ if and only if $k \in \widehat{C}$; the rest of $\beta_k^*$'s are equal to zero. Evidently, we have $\mathbf{1}_n = \Phi\beta^*$ and $f(\beta^*) = \lambda_0 \sum_{j=1}^{|C|} p(|\beta_j^*|) = \lambda_0 \frac{n}{3} p(\tau_0)$. We will show that $f(\beta^*)$ is a global minimum of $f(\beta)$. Moreover, any global solution of (9.26)

corresponds to an exact cover by 3-sets. Hence the NP-hardness of X3C will lead to the NP-hardness of (9.26), and then (PSVM).

Let $\Omega_k, 1 \leq k \leq n$, to be the same subset of indices of $C$ that is defined in Section 3.9.1 (right before Lemma 3.9.1). For any other $\beta' \in \mathbb{R}^{|C|}$, we establish the following lemma.

**Lemma 3.9.4** *For any $k, 1 \leq k \leq n$, we have*

$$\frac{1}{3} \sum_{j \in \Omega_k} \lambda_0 [p(|\beta_j^*|) - p(|\beta_j'|)] \leq \frac{1}{\tau_0} \| \sum_{j \in \Omega_k} (\beta_j^* - \beta_j') \|_+; \tag{9.27}$$

*and the inequality is strict unless both sides of the inequality are equal to zero.*

**Proof.** We consider two cases:

- *Case 1:* when the right hand side of (9.27) is positive, and
- *Case 2:* when the right hand side of (9.27) is equal to zero.

Note the right hand side of (9.27) is nonnegative, the above two cases cover all possibilities.

In *case 1,* we have

$$\| \tau_0 - \sum_{j \in \Omega_k} \beta_j' \|_+ > 0 \Rightarrow \tau_0 > \sum_{j \in \Omega_k} \beta_j'.$$

Using Lemma 3.9.2, we can show that we only need to consider the case when any partial sum of quantities $|\beta_j'|, j \in \Omega_k$, must be less than $\tau_0$. Because otherwise, the left hand side of (9.27) is no more than zero; hence the lemma holds.

Note condition D3 is identical with condition C4. We will show that the following is true: $\forall 0 < x < \tau_0$,

$$p(x) > p(\tau_0) - \frac{p(\tau_0)}{\tau_0}(\tau_0 - x) = \frac{p(\tau_0)}{\tau_0} x. \tag{9.28}$$

To see the above, recall that at the end of Section 3.5, we have proved that when condition D2 holds, we have

$$p(x) \geq p(\tau_0) - \frac{p(\tau_0)}{\tau_0}(\tau_0 - x).$$

Without loss of generality, let us assume that $x < \frac{1}{2}\tau_0$. Recall condition D3 ($p(x) + p(\tau_0 - x) > p(\tau_0)$). If $p(x) = p(\tau_0) - \frac{p(\tau_0)}{\tau_0}(\tau_0 - x) = \frac{p(\tau_0)}{\tau_0}x$, we have

$$p(\tau_0 - x) > p(\tau_0) - \frac{p(\tau_0)}{\tau_0}x = \frac{p(\tau_0)}{\tau_0}(\tau_0 - x).$$

The three points $0 < x < \tau_0 - x$ form a counterexample of the concavity condition. Similarly, when $x > \frac{1}{2}\tau_0$, a counterexample will be found (with three points $\tau_0 - x < x < \tau_0$). The counterexample demonstrates that (9.28) holds.

Utilizing the above results, we have

$$\begin{aligned}
\frac{\text{left hand side of (9.27)}}{\text{right hand side of (9.27)}} &= \frac{\lambda_0\tau_0}{3}\frac{p(\tau_0) - \sum_{j\in\Omega_k} p(|\beta'_j|)}{\|\tau_0 - \sum_{j\in\Omega_k}\beta'_j\|_+} \\
&\overset{\text{C3}}{\leq} \frac{\lambda_0\tau_0}{3}\frac{p(\tau_0) - p(\sum_{j\in\Omega_k}|\beta'_j|)}{\tau_0 - \sum_{j\in\Omega_k}|\beta'_j|} \\
&\overset{(9.28)}{<} \frac{\lambda_0\tau_0}{3}\frac{p(\tau_0)}{\tau_0} \\
&\overset{(6.16)}{\leq} 1.
\end{aligned}$$

Hence (9.27) holds with strict inequality.

For *case 2*, we have $\tau_0 - \sum_{j\in\Omega_k}\beta'_j \leq 0$. Hence we have

$$\tau_0 \leq \sum_{j\in\Omega_k}\beta'_j \leq \sum_{j\in\Omega_k}|\beta'_j|.$$

Using Lemma 3.9.2, we have

$$p(\tau_0) \leq \sum_{j\in\Omega_k} p(|\beta'_j|).$$

The above indicates that the left hand side of (9.27) is no more than zero. Hence (9.27) holds. □

We show that $\beta^*$ minimizes the function $f(\beta)$. Similar to the argument in Section 3.9.1, we add up inequalities (9.27) for all $k, 1 \leq k \leq n$. We have

$$f(\beta^*) \leq f(\beta').$$

The above is true for every $\beta'$; hence $\beta^*$ is a minimizer.

Now we show that the minimum of function $f(\beta)$ is achieved when $\beta$ corresponds to an exact cover by 3-sets. Suppose we have $f(\beta^*) = f(\beta')$. Based on Lemma 3.9.4, both sides of inequality (9.27) must be equal to zero for any $k, 1 \leq k \leq n$. That is, $\forall 1 \leq k \leq n$, we have

$$p(\tau_0) = \sum_{j \in \Omega_k} p(|\beta'_j|), \tag{9.29}$$

and

$$\tau_0 \leq \sum_{j \in \Omega_k} \beta'_j. \tag{9.30}$$

From (9.30), we have $\tau_0 \leq \sum_{j \in \Omega_k} |\beta'_j|$. By Lemma 3.9.2, we have $p(\tau_0) < \sum_{j \in \Omega_k} p(|\beta'_j|)$ when $|\Omega_k| > 1$. Thus, the equality in (9.29) holds if and only if for each $k, 1 \leq k \leq n$, there is exactly one $j, j \in \Omega_k$, such that $\beta'_j = \tau_0$ and the rest of $\beta'_{j'}$'s ($j' \in \Omega_k$) are equal to zero. Evidently, $\beta'$ corresponds to another solution to X3C.

From all the above, we prove the theorem.

# CHAPTER IV

# ELECTRICITY PRICE CURVE MODELING BY MANIFOLD LEARNING

## *4.1 Introduction*

In the competitive electricity wholesale markets, market participants, including power generators and merchants alike, strive to maximize their profits through prudent trading and effective risk management against adverse price movements. A key to the success of market participants is to model the electricity price dynamics well and capture their characteristics realistically. One strand of research on modeling electricity price processes focuses on the aspect of derivative pricing and asset valuation which investigates the electricity spot and forward price models in a risk-neutral world (e.g., [65, 26, 77]). Another research strand concerns the modeling of electricity prices in the physical world, which offers price forecasts for assisting with physical trading and operational decision-making. An accurate short-term price forecast over a time horizon of hours helps market participants to devise their bidding strategies in the auction-based pool-type markets and to allocate generation capacity optimally among different markets. The medium-term forecast with a time horizon spanning days to months is useful for balance sheet calculations and risk management applications [82].

In the second research strand of power price modeling, there is an abundant literature on forecasting spot or short-term electricity prices, especially the day-ahead prices ([25, 88, 21, 20, 50, 67, 27, 83]). Typically, the electricity prices are treated as hourly univariate time series and then modeled by parametric models, including ARIMA processes and their variants ([88, 21, 20]), regime-switching or hidden

Markov processes ([50, 83]), Levy processes [27], hybrid price models combining statistical modeling with fundamental supply-demand modeling ([25]), or nonparametric models such as the artificial neural networks ([91, 98, 114]). While spot price modeling is important, successful trading and risk management operations in electricity markets also require knowledge on an electricity price curve consisting of prices of electricity delivered at a sequence of future times instead of only at the spot. For instance, in order to maximize the market value of generation assets, power generators would need to base their physical trading decisions over how much power to sell in the next day and in the long-term contract markets on both the short-term price forecast for electricity delivered in the next 24 hours and the electricity forward price with maturity ranging from weeks to years. The non-storable nature of electricity makes the electrons delivered at different time points essentially different commodities. The current market price (or spot price) of electricity may have little correlation with that of electricity delivered a few months in the future. Thus, it is imperative to be able to model the electricity price curve as a whole. There is not much literature on modeling electricity price curves. Paper [2] proposes a parametric forward price curve model for the Nordic market, which does not model the movements of the expected future level of a forward curve. A recent paper [76] employs a weighted average of nearest neighbors approach to model and forecast the day-ahead price curve. These works offer little insight on understanding the main drivers of the price curve dynamics. This chapter contributes to this strand of research by proposing a novel nonparametric approach for modeling electricity price curves. Analysis on the intrinsic dimension of an electricity price curve is offered, which sheds light on identifying major factors governing the price curve dynamics. The forecast accuracy of our model compares favorably against that of the ARX and ARIMA model in one-day-ahead price predictions. In addition, our model has a great advantage on the predictions in a longer horizon from days to weeks over other models.

In general, the task of analytically modeling the dynamics of such a price curve is daunting, because the curve is a high-dimensional subject. Each price point on the curve essentially represents one dimension of uncertainty. To reduce the dimension of modeling a price curve and identify the major random factors influencing the curve dynamics, Principle Component Analysis (PCA) is proposed and has been widely applied in the real-world data analysis for industrial practices. As PCA is mainly suited for extracting the linear factors of a data set, it does not appear to perform well in fitting electricity price curves with a linear factor model in a low-dimensional space. However, the following intuition suggests that there shall exist a low-dimensional structure capturing the majority of randomness in the electricity price curve dynamics. Take the day-ahead electricity price curve as an example. While electricity delivered in the next 24 hours are different commodities, the corresponding prices all result from equilibrating the fundamental supply and demand for electricity. The common set of demand and supply conditions in all 24 hours hints a possible nonlinear representation of the 24-dimensional price curve in a space of lower dimension. A natural extension to the PCA approach is to consider the manifold learning methods, which are designed to analyze intrinsic nonlinear structures and features of high-dimensional price curves in the low-dimensional space. After obtaining the low-dimensional manifold representation of price curves, price forecasts are made by first predicting each dimension coordinate of the manifold and then utilizing a reconstruction method to map the forecasts back to the original price space. The conceptual flowchart of our modeling approach is illustrated by Fig. 4. Our major contribution is to establish an effective approach for modeling energy forward price curves, and set up the entire framework in Fig. 4. The other major contribution is to identify the nonlinear intrinsic low-dimensional structure of price curves. The resulting analysis reveals the primary drivers of the price curve dynamics and facilitates accurate price forecasts. This work also enables the application of standard

times series models such as Holt-Winters in the forecast step from box 1 to box 2.

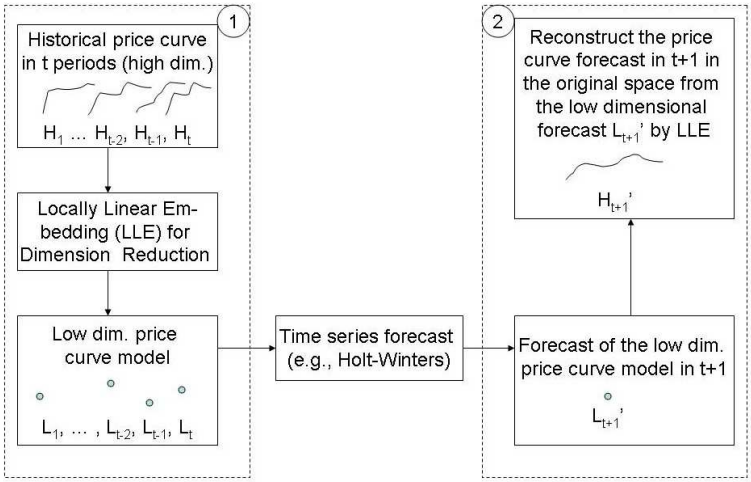



**Figure 4:** The conceptual flowchart of the model.

In this chapter, locally linear embedding (LLE) and LLE reconstruction are adopted for manifold learning and reconstruction. The study of the intrinsic dimension and embedded manifold indicates that there does exist a low-dimensional manifold with the intrinsic dimension around four for day-ahead electricity price curves in the New York Power Pool (known as NYPP).

The rest of the chapter is organized as follows. Section II describes a manifold based method LLE and the corresponding reconstruction method. In Section III, LLE and LLE reconstruction are applied to model and analyze the day-ahead electricity price curves in NYPP. Section IV presents the results of the electricity price curve predictions based on manifold learning. Section V discusses about the extensions and restrictions of our modeling and prediction. Section VI concludes.

## *4.2 Manifold Learning Algorithm*

### 4.2.1 Introduction to Manifold Learning

Manifold learning is a new and promising nonparametric dimension reduction approach. Many high-dimensional data sets that are encountered in real-world applications can be modeled as sets of points lying close to a low-dimensional manifold.

Given a set of data points $x_1, x_2, \cdots, x_N \in \mathbb{R}^D$, we can assume that they are sampled from a manifold with noise, i.e.,

$$x_i = f(y_i) + \varepsilon_i, i = 1, \cdots, N \tag{2.31}$$

where $y_i \in \mathbb{R}^d$, $d \ll D$ and $\varepsilon_i$ is noise. Integer $d$ is also called the intrinsic dimension. The manifold based methodology offers a way to find the embedded low-dimensional feature vectors $y_i$ from the high-dimensional data points $x_i$.

Many nonparametric methods are created for nonlinear manifold learning, including multidimensional scaling (MDS) [6, 70], locally linear embedding (LLE) [95, 96], Isomap [100], Laplacian eigenmaps [3], Hessian eigenmaps [32], local tangent space alignment (LTSA) [115], and diffusion maps [84]. Survey [61] gives a review on the above methods.

Among various manifold based methods, we find that locally linear embedding (LLE) works well in modeling electricity price curves. Our purpose is two-fold: to analyze the features of electricity price curves and predict the price curve at a future time. The reconstruction of high-dimensional forecasted price curves from low-dimensional predictions is a significant step for forecasting. Through extensive computational experiments, we conclude that LLE reconstruction is more efficient relative to other reconstruction methods for our purpose. Moreover, LLE and LLE-reconstruction are fast and easy to implement. In next two subsections, we introduce the algorithms of LLE and LLE reconstruction, respectively.

### 4.2.2 Locally Linear Embedding (LLE)

Given a set of data points $x_1, x_2, \cdots, x_N \in \mathbb{R}^D$ in the high-dimensional space, we are looking for the embedded low-dimensional feature vectors $y_1, y_2, \cdots, y_N \in \mathbb{R}^d$. LLE is a nonparametric method that works as follows [95, 96]:

1. Identify the $k$ nearest neighbors based on Euclidean distance for each data point

$x_i, 1 \leq i \leq N$. Let $N_i$ denote the set of the indices of the $k$ nearest neighbors of $x_i$.

2. Find the optimal local convex combination of the $k$ nearest neighbors to represent each data point $x_i$. That is, the following objective function (2.32) is minimized and the weights $w_{ij}$ of the convex combinations are calculated.

$$E(w) = \sum_{i=1}^{N} \|x_i - \sum_{j \in N_i} w_{ij} x_j\|^2. \tag{2.32}$$

where $\|\cdot\|$ is the $l_2$ norm and $\sum_{j \in N_i} w_{ij} = 1$.

The weight $w_{ij}$ indicates the contribution of the $j$th data point to the representation of the $i$th data point. The optimal weights can be solved as a constrained least squares problem, which is finally converted into a problem of solving a linear system of equation.

3. Find the low-dimensional feature vectors $y_i, 1 \leq i \leq N$, which have the optimal local convex representations with weights $w_{ij}$ obtained from the last step. That is, $y_i$'s are computed by minimizing the following objective function:

$$\Phi(y) = \sum_{i=1}^{N} \|y_i - \sum_{j \in N_i} w_{ij} y_j\|^2. \tag{2.33}$$

It can be shown that solving the above minimization problem (2.33) is equivalent to solving an eigenvector problem with a sparse $N \times N$ matrix. The $d$ eigenvectors associated with the $d$ smallest nonzero eigenvalues of the matrix comprise the $d$-dimensional coordinates of $y_i$'s. Thus, the coordinates of $y_i$'s are orthogonal.

LLE does not impose any probabilistic model on the data; However, it implicitly assumes the convexity of the manifold. It can be seen later that this assumption is satisfied by the electricity price data.

### 4.2.3 LLE Reconstruction

Given a new feature vector in the embedded low-dimensional space, the reconstruction method is used to find its counterpart in the high-dimensional space based on the calibration data set. Reconstruction accuracy is critical for the application of manifold learning in the prediction. There are a limited number of reconstruction methods in the literature. For a specific linear manifold, the reconstruction can be easily made by PCA. For a nonlinear manifold, LLE reconstruction, which is derived in the similar manner as LLE, is introduced in [95]. LTSA reconstruction and nonparametric regression reconstruction are introduced in [115]. Among all these reconstruction methods, LLE reconstruction has the best performance for the electricity data. This is an important reason for us to choose LLE and LLE reconstruction in this chapter.

Suppose low-dimensional feature vectors $y_1, y_2, \cdots, y_N$, have been obtained through LLE in the previous subsection. Denote the new low-dimensional feature vector as $y_0$. LLE reconstruction is applied to find the approximation $\hat{x}_0$ of the original data point $x_0$ in the high-dimensional space based on $x_1 \cdots, x_N$ and $y_1, \cdots, y_N$. There are three steps for LLE reconstruction:

1. Identify the $k$ nearest neighbors of the new feature vector $y_0$ among $y_1, \cdots, y_N$. Let $N_0$ denote the set of the indices of the $k$ nearest neighbors of $y_0$.

2. The weights of the local optimal convex combination $w_j$ are calculated by minimizing
$$E(w) = \|y_0 - \sum_{j \in N_0} w_j y_j\|^2. \tag{2.34}$$
subject to the sum-to-one constraint, $\sum_{j \in N_0} w_j = 1$.

3. Date point $\hat{x}_0$ is reconstructed by $\hat{x}_0 = \sum_{j \in N_0} w_j x_j$.

*Remark:* Solving optimization problems (2.32) and (2.34) is equivalent to solving a linear system of equations. When there are more neighbors than the high dimension

or the low dimension, i.e., $k > D$ or $k > d$, the coefficient matrix associated with the system of linear equations is singular, which means that the solution is not unique. This issue is solved by adding an identity matrix multiplied with a small constant to the coefficient matrix [95]. We adopt this approach here.

Suppose $x_0^{(j)}, 1 \leq j \leq D$, is the $j$th component of vector $x_0$. The *reconstruction error* (RE) of $x_0$ is defined as

$$RE(x_0) = \frac{1}{D} \sum_{j=1}^{D} \frac{|x_0^{(j)} - \hat{x}_0^{(j)}|}{x_0^{(j)}} \tag{2.35}$$

The *reconstruction error of the entire calibration data set* (TRE)[1] is defined as

$$TRE = \frac{1}{N \times D} \sum_{i=1}^{N} \sum_{j=1}^{D} \frac{|x_i^{(j)} - \hat{x}_i^{(j)}|}{x_i^{(j)}} \tag{2.36}$$

by regarding each $y_i$ as a new feature vector $y_0$.

## *4.3 Modeling of Electricity Price Curves with Manifold Learning*

The data of the day-ahead market locational based marginal prices (LBMPs) and integrated real-time actual load of electricity in the Capital Zone of the New York Independent System Operator (NYISO) are collected and predicted in this chapter. The data are available online (*www.nyiso.com/public/market_data/pricing_data.jsp*). In this section, two years (731 days) of price data from Feb 6, 2003 to Feb 5, 2005 are used as an illustration of modeling the electricity price curves by manifold based methodology. Fig. 5(a) plots the hourly day-ahead LBMPs during this period, where the electricity prices are treated as a univariate time series with $24 \times 731$ hourly prices. Fig. 5(b), 5(c) and 5(d) illustrates the mean, standard deviation and skewness of 24 hourly log prices in each day after outlier processing.

The section is organized as follows. First, the data are preprocessed with log transform, outlier processing and LLP smoothing, and then the results of the manifold

[1]When the TRE is calculated, $y_i$ itself is not included in its $k$ nearest neighbors.

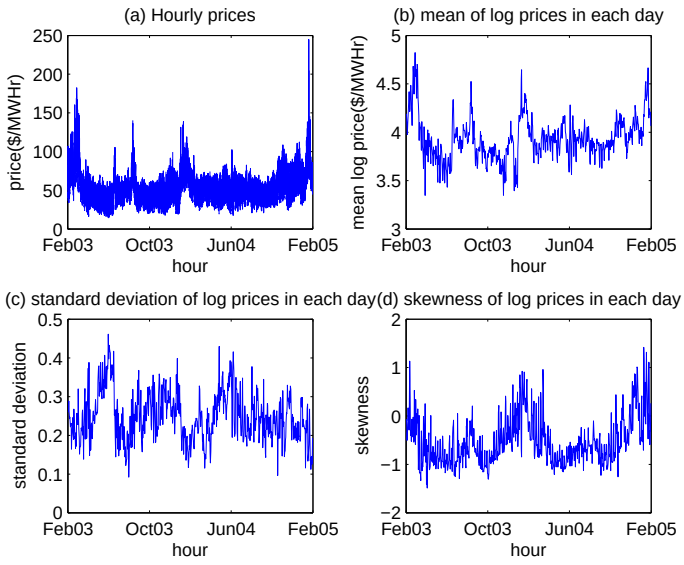



**Figure 5:** Day-ahead LBMPs from Feb 6, 2003 to Feb 5, 2005 in the Capital Zone of NYISO.

learning and reconstruction are illustrated. Next, the major factors of electricity price curve dynamics are analyzed with low-dimensional feature vectors. Finally, the parameter selections and the sensitivity of reconstruction error to those parameters are analyzed.

### 4.3.1 Preprocessing

#### *4.3.1.1 Log Transform*

The logarithmic (log) transforms of the electricity prices are taken before the manifold learning. There are several advantages to deal with the log prices. First, the electricity prices are well known to have the non-constant variance, and log transform can make the prices less volatile. The log transform also enhances the efficiency of manifolding learning, by making the embedded manifold more uniformly distributed in the low-dimensional space and the *reconstruction error of the entire calibration data set* (TRE) reduced. Moreover, the log transform has the interpretation of the returns to someone holding the asset.

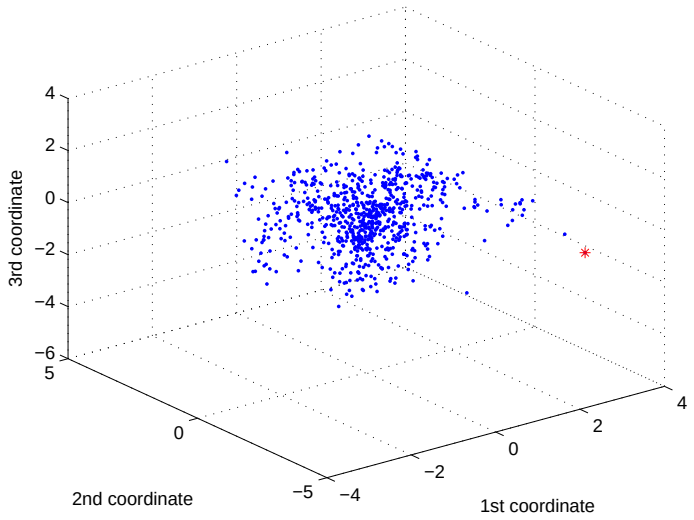



**Figure 6:** Embedded three-dimensional manifold without any outlier preprocessing(but with log transform and LLP smoothing). "∗" indicates the day with outliers—Jan 24, 2005.

#### *4.3.1.2 Outlier Processing*

Outliers in this chapter are defined as the electricity price spikes that are extremely different from the prices in the neighborhood. To deal with the outliers, we replace the prices in the day with outliers by the average of the prices in the days right before and right after. We remove the outliers because the embedded low-dimensional manifold is supposed to extract the primary features of the entire data set, rather than the individual and local features such as extreme price spikes. The efficiency of manifold learning is improved after outlier processing. Moreover, outliers, which represent rarely occurring phenomena in the past, often have very small probability to occur in the near future, so the processing of outliers does not severely affect the prediction of the near-term regular prices.

In the illustrated data set, only one extreme spike is identified on the right of Fig. 5(a), which belongs to Jan 24, 2005. In the low-dimensional manifold, the days of outliers can also be detected by the points that stand far away from the other points. Fig. 6 shows that the point corresponding to Jan 24, 2005 lies out of the main cloud

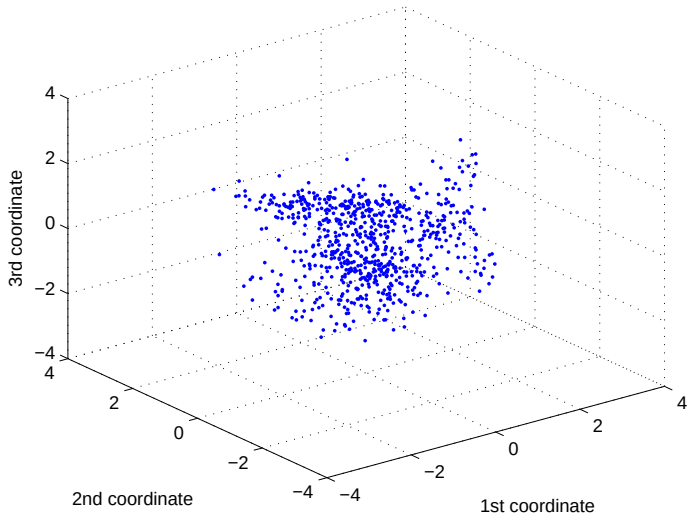



**Figure 7:** Embedded three-dimensional manifold after log transform, outlier preprocessing and LLP smoothing.

of the points on the embedded three-dimensional manifold. Thus, we regard Jan 24, 2005 as a day with outliers. Fig. 7 shows that the low-dimensional manifold after removing the outliers is more uniformly distributed.

#### *4.3.1.3 LLP Smoothing*

The noise in (2.31) can contaminate the learning of the embedded manifold and the estimation of the intrinsic dimension. Therefore, locally linear projection (LLP) ([60, 59, 61]) is recommended to smooth the manifold and reduce the noise. The description of the algorithm is given as follows:

---

ALGORITHM: LLP

For each observation $x_i, i = 1, 2, \cdots, N$,

1. Find the $k$-nearest neighbors of $x_i$. The neighbors are denoted by $\widetilde{x}_1, \widetilde{x}_2, \cdots, \widetilde{x}_k$.
2. Use PCA or SVD to identify the linear subspace that contains most of the information in the vectors $\widetilde{x}_1, \widetilde{x}_2, \cdots, \widetilde{x}_k$. Suppose the linear subspace is $A_i$. Let $k_0$ denote the assumed dimension of the embedded manifold. Then subspace $A_i$

can be viewed as a linear subspace spanned by the singular vectors associated with the largest $k_0$ singular values.

3. Project $x_i$ into the linear subspace $A_i$ and let $\check{x}_i, i = 1, \cdots, N$, denote the projected points.

---

After denoising, the efficiency of manifold learning is enhanced, and the TRE is reduced. For the illustrated data set with the intrinsic dimension being four, the TRE is 3.89% after LLP smoothing, compared to 4.41% without LLP smoothing. The choice of the two parameters in LLP, the dimension of the linear space and the number of the nearest neighbors, will be discussed in detail in subsection D.

### 4.3.2 Manifold Learning by LLE

Each price curve with 24 hourly prices in a day is considered as an observation, so the dimension of the high-dimensional space $D$ is 24. The intrinsic dimension $d$ is set to be four. The number of the nearest neighbors $k$ for LLP smoothing, LLE, and LLE reconstruction is selected to be a common number 23 for all the numerical studies. The details of the parameter selections are discussed in subsection D. Due to the ease of visualization in a three-dimensional space, all the low-dimensional manifolds are plotted with the intrinsic dimension being three. We apply LLE to the denoised data $\check{x}_i, i = 1, \cdots, N$, which are obtained after LLP smoothing. Fig. 7 provides the plot of the embedded three-dimensional manifold. As the low-dimensional manifold is nearly convex and uniformly distributed, LLE is an appropriate manifold based method. Fig. 8 plots the time series of each coordinates of the feature vectors in the embedded four-dimensional manifold.

Table 1 shows the TRE of different reconstruction methods. LLE reconstruction has the minimum reconstruction error among all the methods. LTSA reconstruction has a very large TRE, because it is an extrapolation-like method, and the reconstruction of some of the price curves has very large errors. Therefore, LLE and LLE

**Table 1:** The TRE of different reconstruction methods

| Reconstruction method | TRE(%) |
|---|---|
| LLE and LLE reconstruction | 3.89 |
| PCA and PCA reconstruction | 4.26 |
| LTSA and LTSA reconstruction | $4.55 \times 10^6$ |
| LLE and nonparametric regression reconstruction | 4.77 |

reconstruction are selected to model the electricity price dynamics.

### 4.3.3 Analysis of Major Factors of Electricity Price Curve Dynamics with Low-Dimensional Feature Vectors

The interpretation of each dimension in the low-dimensional space and the cluster analysis to the low-dimensional feature vectors reveal the major drivers of the price curve dynamics, which suggests that our prediction methods in the next section based on the modeling of price curves with manifold learning are reasonable.

#### *4.3.3.1 Interpretation of Each Dimension in the Low-Dimensional Space*

There are some interesting interpretations for the first three coordinates of the feature vectors in the low-dimensional space. For each price curve, we can calculate the mean, standard deviation, range, skewness and kurtosis of the 24 hourly log prices. The sequence of each coordinates of the low-dimensional feature vectors comprises a time series. The correlation between each time series and mean log prices (standard deviation, range, skewness and kurtosis) is calculated. Table 2 shows the one of the four-dimensional coordinates, which have the maximum absolute correlation with mean log prices (standard deviation, range, skewness and kurtosis), and the corresponding correlation coefficients. The comparison between Fig. 5 and Fig. 8 gives more intuition about the correlations. It is found that the first coordinates have a very high correlation coefficient 0.9964 with the mean log prices within each day, and the second coordinates are highly correlated with the standard deviation of

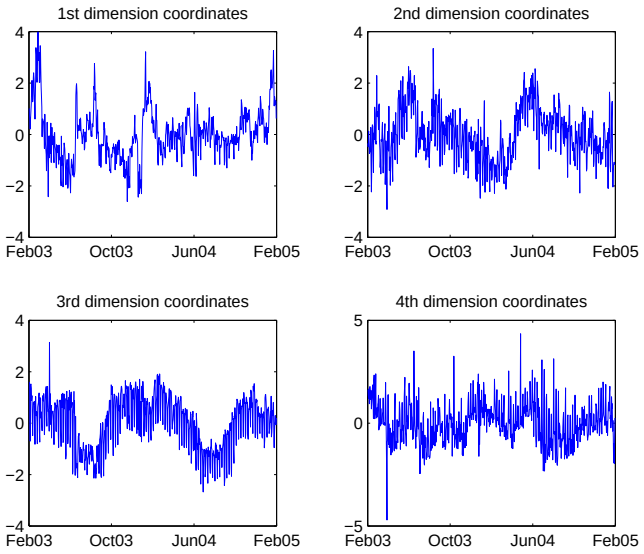



**Figure 8:** Coordinates of the embedded 4-dim manifold.

**Table 2:** The one of the four-dimensional coordinates which has the maximum absolute correlation coefficient with the mean (standard deviation, range, skewness and kurtosis) of log prices in a day in embedded four-dimensional space.

| | Mean | Std. Dev. | Range | Skewness | Kurtosis |
|---|---|---|---|---|---|
| Coordinate | 1st | 2nd | 2nd | 2nd | 3rd |
| Correlation Coefficient | 0.9964 | 0.7073 | 0.5141 | -0.5646 | 0.2611 |

the log prices in a day with a correlation coefficient 0.7073. This also means that the second coordinates contain some other information besides standard deviation, and Table 2 demonstrates that the second coordinates are also correlated, but not significantly, with range and skewness. The third coordinates show both weekly and yearly seasonality in Fig. 8. Weekly seasonality is well known for electricity prices. Yearly seasonality may be caused by the shape change of the price curves over the year. The shape of price curves is often unimodal in the summer and bimodal in the winter.

#### *4.3.3.2 Cluster Analysis*

The yearly seasonality of the electricity price curves can be clearly demonstrated by the cluster analysis of low-dimensional feature vectors.

Cluster analysis [56] (also known as data segmentation) groups or segments a collection of objects into subsets (i.e., clusters), such that those within each cluster are more closely related to each other than those assigned to different clusters.

The K-means clustering algorithm is one of the mostly used iterative clustering methods. Assume that there are $K$ clusters. The algorithm begins with a guess of the $K$ cluster centers. Then, the algorithm iterates between the following two steps until convergence. The first step is to identify the closest cluster center for each data point based on some distance metric. The second step is to replace each cluster center with the coordinate-wise average of all the data points that are the closest to it.

For the electricity price data, we apply K-means clustering with Euclidean distance to the low-dimensional feature vectors that are obtained from manifold learning. The number of clusters is set to be three, as the yearly seasonality can be clearly illustrated with three clusters. The coordinate-wise average of price curves in each cluster is plotted in Fig. 9. The distribution of clusters is illustrated in the first graph of Fig. 10, where $x$ axis is the date of the price curves, and $y$ axis is the corresponding clusters. The two graphs show that the first cluster represents the price curves from the summer, which are featured with unimodal shape, and the second cluster represents the ones from the winter, which are characterized with bimodal shape. The price curves in the third cluster reveal the transition from unimodal shape to bimodal shape. The average price curves in the 3 clusters closely resemble the typical load shapes observed in summer, winter, and rest-of-year, respectively.

The second graph of Fig. 10 shows the distribution of clusters by applying K-means clustering with correlation distance to the high-dimensional price curves. The two graphs in Fig. 10 have the similar patterns, which gives a good illustration that

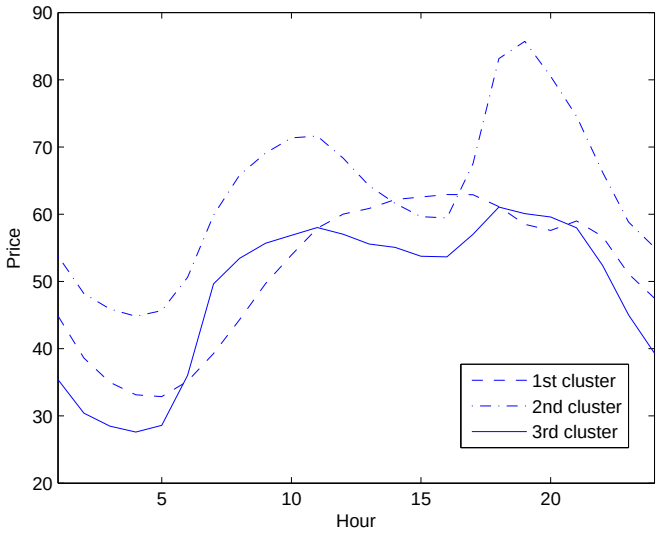



**Figure 9:** The coordinate-wise average of the actual price curves in each cluster, where clustering is based on low-dimensional feature vectors.

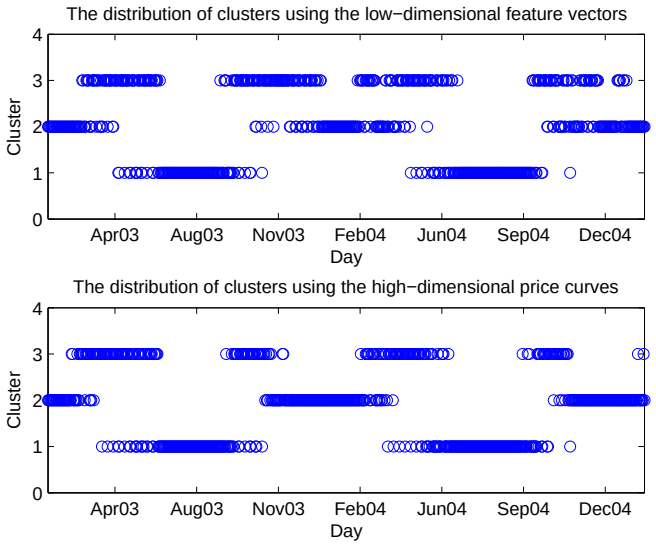



**Figure 10:** Distribution of clusters.

low-dimensional feature vectors capture the major factors of the price curve dynamics.

### 4.3.4 Parameter Setting and Sensitivity Analysis

The selections of several parameters, including the number of intrinsic dimensions, the number of the nearest neighbors and the length of the calibration data, are discussed in this subsection.

#### *4.3.4.1 Intrinsic Dimension*

Intrinsic dimension $d$ is an important parameter of manifold learning. Papers [75] and [110] provide several approaches of estimating the intrinsic dimension. In [75], the maximum likelihood estimator of the intrinsic dimension is established. In [110], the intrinsic dimension is estimated based on a nearest neighbor algorithm. Without LLP smoothing, the two methods show that the intrinsic dimension is some value between four and five. Thus, it is reasonable to set the dimension of the linear space as four in LLP smoothing. After LLP smoothing, the intrinsic dimension is reduced to a value between three and four. The numerical experiments indicate that LLP smoothing can not only denoise, but also improve the efficiency of estimating the intrinsic dimension.

Another empirical way of estimating the intrinsic dimension is to analyze the sensitivity of the TRE to the different values of the intrinsic dimension. Fig. 11 shows that the TRE is a decreasing function of the intrinsic dimension with a increasing slope. The slope of the curve in the figure has a dramatic change when the intrinsic dimension is around four. Therefore, we choose the intrinsic dimension as four in this chapter.

#### *4.3.4.2 The Number of the Nearest Neighbors*

The plot of the TRE against the number of the nearest neighbors is used to select the appropriate number of the nearest neighbors. Fig. 12 indicates the TRE first falls

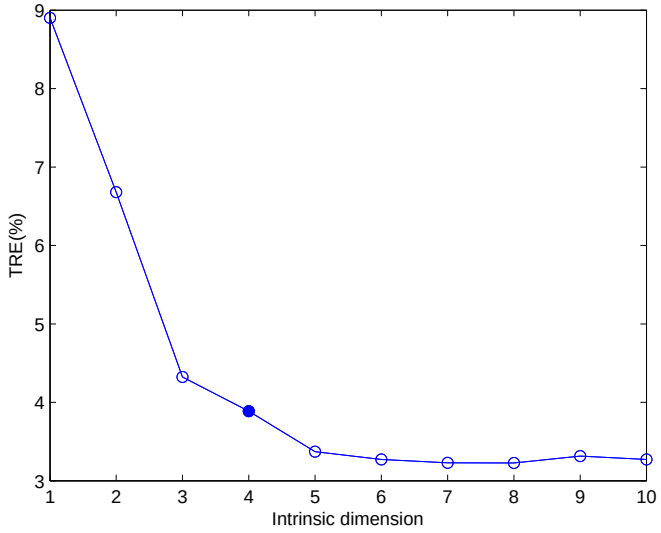



**Figure 11:** The sensitivity of TRE to the intrinsic dimension (data length = 731 days, number of the nearest neighbors = 23).

steeply when the number of the nearest neighbors is small, and then remains steady when the number of the nearest neighbors is greater than 22. We set the number of the nearest neighbors to be 23 for all the numerical studies. This is only one of the many choices as the reconstruction error is not sensitive to the number of the nearest neighbors within a range.

#### *4.3.4.3 The Length of the Calibration Data*

The plot of the TRE against the length of the calibration data in Fig. 13 illustrates that the TRE is not very sensitive to the data length. Two years of data are applied to the manifold learning, and it helps to study whether there is yearly seasonality.

## *4.4 Prediction of Electricity Price Curves*

The prediction of future electricity price curves is an important issue in the electricity price market, because accurate predictions enable market participants to increase their profit by trading energy and hedge the potential risk successfully. However, it is difficult to make accurate predictions for the electricity prices due to their multiple

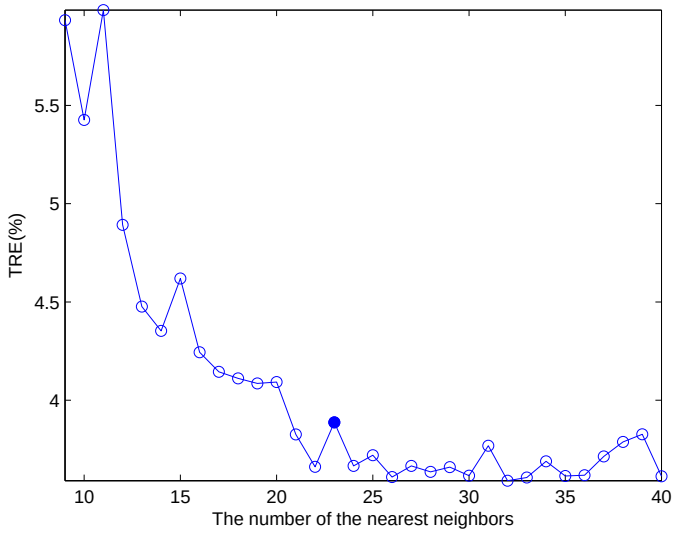



**Figure 12:** The sensitivity of TRE to the number of the nearest neighbors (data length = 731 days, intrinsic dimension = 4).

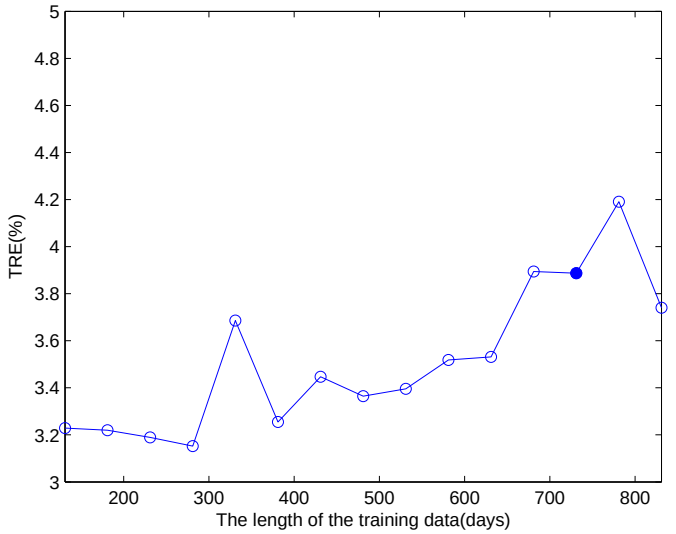



**Figure 13:** The sensitivity of TRE to the length of the calibration data (intrinsic dimension = 4, number of the nearest neighbors = 23).

seasonalities—daily and weekly seasonality. The speciality of the electricity price data often results in complicated models to forecast future electricity prices, which are often overfitting and fail to make accurate predictions in a longer horizon. Our method converts the hourly electricity price time series with multiple seasonalities into several time series with only weekly seasonality by manifold learning. After conversion, each data point in the new time series represents a day rather than an hour. The simplification of the new time series makes the longer horizon prediction easier and more accurate. Therefore, our method has an advantage in the longer horizon prediction over many other prediction methods.

A large amount of existing forecasting methods focus on one-day-ahead price predictions, i.e., the horizon of prediction is one day (24 hours). Two articles [82] and [19] give a good review on many prediction methods, and make a comparison on their performance. In this chapter, we compare our prediction methods with three models —ARIMA, ARX and the naive method. The ARIMA model [21] and the naive method are pure time series methods. The ARX model (also called dynamic regression model) includes the explanatory variable, load, and is suggested to be the best model in [19] and one of the best models in [82].

The longer horizon prediction has not drawn much attention so far. However, it also plays an important role in biding strategy and risk management. Our numerical results show that our prediction methods not only generate competent results in forecasting one-day-ahead price curves, but also produce more accurate predictions for one-week-ahead and one-month-ahead price curves, compared to ARX, ARIMA and the naive method. Moreover, as the new time series generated by manifold learning are simple, it is very easy to identify the time series models or utilize some nonparametric forecasting techniques. Our prediction methods also allow larger size of data for model calibration and incorporate more past information, but the size of the calibration data for ARIMA and ARX is often restricted to be several months.

### 4.4.1 Prediction Method

In our prediction method, we first make the prediction in the low-dimensional space, and then reconstruct the predicted price curves in the high-dimensional space from the low-dimensional prediction. There are three steps in detail:

1. Learn the low-dimensional manifold of electricity price curves with LLE. The sequence of each coordinates of the low-dimensional feature vectors comprises a time series.

2. Predict each time series in the low-dimensional space via univariate time series forecasting. Three prediction methods are applied: the Holt-Winters algorithm (HW) [7], the structural model (STR) [7] and the seasonal decomposition of time series by loess (STL) [16]. Each data point in the time series represents one day, so for the one-week-ahead (one-day-ahead or one-month-ahead) price curve predictions, seven (one or 28) data points are forecasted for each time series.

3. Reconstruct the predicted price curves in the high-dimensional space from the predictions in low-dimensional space with LLE reconstruction.

The first and third step have been described in the previous sections. In the second step, we make the univariate time series forecasting for each coordinates of the feature vectors rather than making the multivariate time series forecasting for all the time series in the low-dimensional space, because the coordinates are orthogonal to each other.

There are a variety of methods of univariate time series forecasting, among which Holt-Winters algorithm, structural model and STL are selected. Both the Holt-Winters algorithm and structural model are pure time series prediction methods (models), and do not require any model identification as in ARIMA. The STL method

can involve the explanatory variable in the prediction. All the prediction methods can be easily and fast implemented in statistical software R. The following is some brief description of the three prediction methods.

#### *4.4.1.1 Holt-Winters Algorithm(HW)*

In Holt-Winters filtering, seasonals and trends are computed by exponentially weighted moving averages. In our numerical experiments, Holt-Winters algorithm is executed with starting period equal to 7 days and 14 days respectively. This choice is due to the weekly effect of the electricity prices.

#### *4.4.1.2 Structural Models (STR)*

Structural time series model is a (linear Gaussian) state-space model for (univariate) time series based on a decomposition of the series into a number of components—trend, seasonal and noise.

#### *4.4.1.3 Seasonal Decomposition of Time Series by Loess (STL)*

The STL method can involve explanatory variables in the prediction. As the effect of temperature is usually embodied in electricity loads, only load is utilized as an exploratory variable. We first learn the manifold with the intrinsic dimension four for both prices and loads, and then decompose each time series in the low-dimensional space of price and load curves into seasonal, trend and irregular components using loess. Let $P_{i,t}$ and $Z_{i,t}$ denote the trend [2]of the $i$th coordinates of the feature vectors for prices and loads at time t. Then, we regress $P_{i,t}$ on $Z_{i,t}$ and the lagged $P_{i,t}$ with the lag three. As the relationship between prices and loads are dynamic, the history data we applied to train the model are 70 days. The model is written as:

$$P_{i,t} = \beta_0 + \beta_1 Z_{i,t} + \beta_2 P_{i,t-1} + \beta_3 P_{i,t-2} + \beta_4 P_{i,t-3} + \varepsilon_t$$

[2]trend window=5

### 4.4.2 The Definition of Weekly Average Prediction Error

To assess the predictive accuracy of our methodology, three weekly average prediction errors are defined for one-day-ahead, one-week-ahead and one-month-ahead price predictions, respectively.

#### *4.4.2.1 Weekly Average One-Day-Ahead Prediction Error*

For the $i$th day of a certain week, $i = 1, \cdots, 7$, the calibration data are set to be the two-year data right before this day, and then one-day-ahead predictions are made, i.e., the horizon of the prediction is one day. The predictions are denoted as $\hat{x}_{(i,1)}$, which is a 24-dimensional vector. The one-day-ahead prediction error for the $i$th day is defined as

$$\mathrm{WPE}_d^{(i)} = \frac{1}{24} \frac{\|x_{(i,1)} - \hat{x}_{(i,1)}\|_1}{\bar{x}_{(i)}^d}$$

where $\bar{x}_{(i)}^d$ is the average of the actual electricity prices on the $i$th day. $\|\cdot\|_1$ is the $L_1$ norm of a vector, which is the sum of the absolute values of all the components in the vector.

The weekly average one-day-ahead prediction error is defined as

$$\mathrm{WPE}_d = \frac{1}{7} \sum_{i=1}^{7} \mathrm{WPE}_d^{(i)}$$

#### *4.4.2.2 Weekly Average One-Week-Ahead Prediction Error*

For the $i$th day of a certain week, $i = 1, \cdots, 7$, the calibration data are set to be the two-year data right before this day, and then one-week-ahead predictions are made, i.e., the horizon of the prediction is one week. The $j$th-day-ahead predictions are denoted as $\hat{x}_{(i,j)}$, $j = 1, \cdots, 7$. The one-week-ahead prediction error for the $i$th day is defined as

$$\mathrm{WPE}_w^{(i)} = \frac{1}{7 \times 24} \sum_{j=1}^{7} \frac{\|x_{(i,j)} - \hat{x}_{(i,j)}\|_1}{\bar{x}_{(i)}^w}$$

where $\bar{x}_{(i)}^w$ is the average of the actual electricity prices of the one-week-ahead predictions.

The weekly average one-week-ahead prediction error is defined as

$$\mathrm{WPE}_w = \frac{1}{7}\sum_{i=1}^{7} \mathrm{WPE}_w^{(i)}$$

#### *4.4.2.3 Weekly Average One-Month-Ahead Prediction Error*

For the $i$th day of a certain week, $i = 1, \cdots, 7$, the calibration data are set to be the two-year data right before this day, and then one-month-ahead (28-days-ahead) predictions are made, i.e., the horizon of the prediction is one month. The $j$th-day-ahead predictions are denoted as $\hat{x}_{(i,j)}$, $j = 1, \cdots, 28$. The one-month-ahead prediction error for the $i$th day is defined as

$$\mathrm{WPE}_m^{(i)} = \frac{1}{28 \times 24}\sum_{j=1}^{28} \frac{\|x_{(i,j)} - \hat{x}_{(i,j)}\|_1}{\bar{x}_{(i)}^m}$$

where $\bar{x}_{(i)}^m$ is the average of the actual electricity prices of the one-month-ahead predictions.

The weekly average one-month-ahead prediction error is defined as

$$\mathrm{WPE}_m = \frac{1}{7}\sum_{i=1}^{7} \mathrm{WPE}_m^{(i)}$$

We define $\sigma_d$, $\sigma_w$ and $\sigma_m$ as the standard deviations of $\mathrm{WPE}_d^{(i)}$, $\mathrm{WPE}_w^{(i)}$ and $\mathrm{WPE}_m^{(i)}$, respectively.

### 4.4.3 Prediction of Electricity Price Curves

Our numerical experiments are based on 12 weeks from February 2005 to January 2006, which consist of the second week of each month. Three weekly average prediction errors as defined above are calculated for each week, respectively. For each data set, the same parameter values taken from the previous section are used. The number of the nearest neighbors and the intrinsic dimension are set to be 23 and 4, respectively. Only one day, Jan 24, 2005, is identified with outliers. As we only have the forecasts of loads for six future days from the NYISO website, the weekly average

one-week-ahead prediction error for STL and ARX is actually the weekly average six-days-ahead prediction error.

Table 3 and 4 provides the weekly average one-day-ahead prediction errors for the 12 weeks and their standard deviations. Our prediction methods—Holt-Winters, structural model and STL—are compared with ARX, ARIMA and the naive method. The details of the ARIMA and ARX model are in Appendix A and B. The naive predictions of a certain week are given by the actual prices of the previous week. Holt-Winters and structural model outperform all the other methods. It seems that involving the exploratory variable does not necessarily improve the prediction accuracy. STL performs slightly worse than Holt-Winters and structural model, and ARX also has less accuracy than ARIMA. This is not consistent with the results in [19] and [82], where ARX has better performance than ARIMA. A potential cause is that the predictions of loads are not precise, or the correlation between loads and prices is not high enough in NYPP.

In Table 5 and 6, the weekly average one-week-ahead prediction errors for the 12 weeks and their standard deviations are presented. All of our prediction methods outperform ARX, ARIMA, and the naive method. The ARIMA model acts even worse than the naive method for one-week-ahead predictions. Since the ARIMA model is a very complicated model with multiple seasonalities, it is often overfitting and makes the longer horizon predictions less accurate. The ARX model is a little simpler and given more information by the load forecasts, so it performs better than ARIMA. However, both ARX and ARIMA need to predict 168 data points for one-week-ahead predictions, while our prediction methods only need to predict seven data points for each time series. Therefore, our prediction methods have a great advantage in the longer horizon predictions. Among Holt-Winters, structural model and STL, STL has slightly worse performance than other two, and structural model is the most accurate.

**Table 3:** Comparison of $\mathrm{WPE}_d(\%)$ of one-day-ahead predictions for 12 weeks.

| date | HW7* | HW14 | STR | STL | ARIMA | ARX | Naive |
|---|---|---|---|---|---|---|---|
| 02/06/05 ∼ 02/12/05 | 7.14 | 6.97 | 7.34 | 7.07 | 7.94 | 6.58 | 15.99 |
| 03/06/05 ∼ 03/12/05 | 6.29 | 5.82 | 5.48 | 6.03 | 5.66 | 8.02 | 9.81 |
| 04/03/05 ∼ 04/09/05 | 6.54 | 7.11 | 6.59 | 7.73 | 6.10 | 6.25 | 12.38 |
| 05/08/05 ∼ 05/14/05 | 6.45 | 6.00 | 6.22 | 7.39 | 7.47 | 6.17 | 5.89 |
| 06/05/05 ∼ 06/11/05 | 9.87 | 9.38 | 9.86 | 9.01 | 9.85 | 11.95 | 31.46 |
| 07/03/05 ∼ 07/09/05 | 7.79 | 8.46 | 7.55 | 7.07 | 7.38 | 6.06 | 17.90 |
| 08/07/05 ∼ 08/13/05 | 5.16 | 5.17 | 5.42 | 8.56 | 6.05 | 7.03 | 13.09 |
| 09/04/05 ∼ 09/10/05 | 6.98 | 8.14 | 7.55 | 7.75 | 7.20 | 5.58 | 14.55 |
| 10/02/05 ∼ 10/08/05 | 6.15 | 6.08 | 6.45 | 6.63 | 6.37 | 6.36 | 9.64 |
| 11/06/05 ∼ 11/12/05 | 6.71 | 6.65 | 6.11 | 6.30 | 5.91 | 5.41 | 18.78 |
| 12/04/05 ∼ 12/10/05 | 8.66 | 8.84 | 8.96 | 7.95 | 8.47 | 12.75 | 26.95 |
| 01/08/06 ∼ 01/14/06 | 8.63 | 8.72 | 8.26 | 9.28 | 10.49 | 8.17 | 15.61 |
| mean | 7.20 | 7.28 | 7.15 | 7.56 | 7.41 | 7.53 | 16.00 |

*HW7 and HW14 stand for Holt-Winter algorithm with starting period equal to 7 days and 14 days respectively.

**Table 4:** Comparison of $\sigma_d(\%)$ of one-day-ahead predictions for 12 weeks.

| date | HW7 | HW14 | STR | STL | ARIMA | ARX | Naive |
|---|---|---|---|---|---|---|---|
| 02/06/05 ∼ 02/12/05 | 3.03 | 3.34 | 3.10 | 2.21 | 4.58 | 2.28 | 4.66 |
| 03/06/05 ∼ 03/12/05 | 1.89 | 1.96 | 1.81 | 2.21 | 2.16 | 3.15 | 5.67 |
| 04/03/05 ∼ 04/09/05 | 2.24 | 2.88 | 2.34 | 2.76 | 1.83 | 3.13 | 3.33 |
| 05/08/05 ∼ 05/14/05 | 3.86 | 3.36 | 3.47 | 3.34 | 4.07 | 2.54 | 1.40 |
| 06/05/05 ∼ 06/11/05 | 3.31 | 3.36 | 4.03 | 2.53 | 4.31 | 6.13 | 5.62 |
| 07/03/05 ∼ 07/09/05 | 3.62 | 4.68 | 3.73 | 4.01 | 3.37 | 2.15 | 8.79 |
| 08/07/05 ∼ 08/13/05 | 1.65 | 1.53 | 2.28 | 3.59 | 3.01 | 2.63 | 4.37 |
| 09/04/05 ∼ 09/10/05 | 3.68 | 3.59 | 3.09 | 3.83 | 4.48 | 2.16 | 8.38 |
| 10/02/05 ∼ 10/08/05 | 2.27 | 2.02 | 2.26 | 3.36 | 2.18 | 3.06 | 4.45 |
| 11/06/05 ∼ 11/12/05 | 2.57 | 2.41 | 2.60 | 2.57 | 2.36 | 3.08 | 7.06 |
| 12/04/05 ∼ 12/10/05 | 4.14 | 4.27 | 3.69 | 2.36 | 2.80 | 6.62 | 10.55 |
| 01/08/06 ∼ 01/14/06 | 4.61 | 4.42 | 3.73 | 3.93 | 5.79 | 4.49 | 4.71 |
| mean | 3.07 | 3.15 | 3.01 | 3.06 | 3.41 | 3.45 | 5.75 |

**Table 5:** Comparison of $\text{WPE}_w(\%)$ of one-week-ahead predictions for 12 weeks

| date | HW7 | HW14 | STR | STL | ARIMA | ARX | Naive |
|---|---|---|---|---|---|---|---|
| 02/06/05 ∼ 02/12/05 | 8.31 | 8.33 | 7.65 | 8.28 | 14.57 | 9.00 | 12.19 |
| 03/06/05 ∼ 03/12/05 | 10.85 | 10.59 | 10.19 | 12.57 | 13.28 | 15.30 | 12.33 |
| 04/03/05 ∼ 04/09/05 | 11.03 | 10.88 | 10.53 | 11.74 | 10.68 | 11.46 | 14.59 |
| 05/08/05 ∼ 05/14/05 | 7.55 | 7.15 | 7.36 | 6.89 | 14.21 | 6.16 | 6.71 |
| 06/05/05 ∼ 06/11/05 | 15.58 | 15.23 | 13.88 | 11.39 | 21.64 | 19.49 | 26.68 |
| 07/03/05 ∼ 07/09/05 | 10.85 | 10.04 | 10.41 | 9.60 | 14.63 | 7.03 | 14.44 |
| 08/07/05 ∼ 08/13/05 | 6.82 | 7.09 | 6.42 | 12.87 | 9.49 | 9.14 | 10.28 |
| 09/04/05 ∼ 09/10/05 | 7.46 | 7.95 | 7.31 | 8.52 | 10.36 | 6.86 | 13.17 |
| 10/02/05 ∼ 10/08/05 | 9.78 | 9.60 | 10.11 | 8.97 | 11.84 | 8.30 | 11.57 |
| 11/06/05 ∼ 11/12/05 | 9.45 | 9.44 | 8.99 | 9.42 | 11.24 | 9.00 | 15.18 |
| 12/04/05 ∼ 12/10/05 | 12.94 | 13.09 | 13.30 | 11.55 | 21.78 | 22.92 | 23.94 |
| 01/08/06 ∼ 01/14/06 | 13.95 | 14.08 | 13.28 | 15.00 | 26.01 | 16.37 | 11.52 |
| mean | 10.38 | 10.29 | 9.95 | 10.57 | 14.98 | 11.75 | 14.38 |

The proposed method can be applied to forecast prices in a longer horizon than one week, e.g., two weeks or even one month. As there are only a few methods associated with one-month-ahead price predictions, we apply three naive methods to compare with. The first naive method takes the last month prices in the calibration data set as the predictions. The second method repeats the last week prices four times, and the third one replicates the prices of last two weeks twice, respectively, as the predictions. Table 7 and 8 provide the weekly average prediction errors of the one-month-ahead price predictions for the 12 weeks and their standard deviations. The notations—naive1, naive2 and naive3—stand for the three naive methods. From the comparison, the proposed methods outperform all the naive methods. We notice that the total stand deviation of the structural model is larger than that of the naive methods, and it is mainly due to an inaccurate prediction for one day in week five. Thus, Holt-Winters algorithm has the best performance among all the methods for one-month-ahead price predictions.

**Table 6:** Comparison of $\sigma_w(\%)$ of one-week-ahead predictions for 12 weeks

| date | HW7 | HW14 | STR | STL | ARIMA | ARX | Naive |
|---|---|---|---|---|---|---|---|
| 02/06/05 ∼ 02/12/05 | 1.32 | 1.66 | 1.14 | 1.85 | 7.06 | 2.62 | 2.35 |
| 03/06/05 ∼ 03/12/05 | 3.80 | 3.73 | 3.86 | 3.88 | 7.06 | 2.22 | 1.18 |
| 04/03/05 ∼ 04/09/05 | 3.12 | 3.48 | 3.41 | 2.90 | 3.50 | 4.18 | 2.47 |
| 05/08/05 ∼ 05/14/05 | 3.45 | 3.13 | 3.50 | 1.99 | 6.28 | 1.82 | 0.88 |
| 06/05/05 ∼ 06/11/05 | 5.91 | 6.10 | 6.46 | 4.64 | 10.30 | 8.18 | 4.75 |
| 07/03/05 ∼ 07/09/05 | 3.10 | 3.44 | 3.43 | 1.94 | 5.51 | 0.55 | 1.50 |
| 08/07/05 ∼ 08/13/05 | 1.02 | 0.95 | 1.40 | 3.10 | 3.77 | 2.60 | 1.48 |
| 09/04/05 ∼ 09/10/05 | 3.19 | 2.96 | 2.33 | 2.75 | 4.88 | 1.17 | 2.28 |
| 10/02/05 ∼ 10/08/05 | 2.06 | 1.73 | 2.01 | 4.32 | 5.50 | 3.11 | 1.34 |
| 11/06/05 ∼ 11/12/05 | 2.17 | 2.42 | 2.30 | 2.40 | 3.92 | 2.14 | 3.05 |
| 12/04/05 ∼ 12/10/05 | 4.88 | 4.41 | 5.17 | 4.53 | 13.77 | 4.47 | 4.65 |
| 01/08/06 ∼ 01/14/06 | 3.25 | 3.37 | 2.15 | 6.15 | 11.60 | 3.70 | 2.66 |
| mean | 3.11 | 3.12 | 3.10 | 3.37 | 6.93 | 3.06 | 2.38 |

**Table 7:** Comparison of $\mathrm{WPE}_m(\%)$ of one-month-ahead predictions for 12 weeks

| date | HW7 | HW14 | STR | Naive1 | Naive2 | Naive3 |
|---|---|---|---|---|---|---|
| 02/06/05 ∼ 02/12/05 | 8.87 | 9.32 | 9.42 | 29.63 | 12.07 | 25.72 |
| 03/06/05 ∼ 03/12/05 | 12.52 | 12.07 | 12.16 | 17.17 | 13.71 | 14.39 |
| 04/03/05 ∼ 04/09/05 | 17.62 | 17.28 | 16.51 | 14.96 | 15.96 | 14.20 |
| 05/08/05 ∼ 05/14/05 | 12.00 | 11.81 | 11.93 | 13.62 | 11.52 | 12.04 |
| 06/05/05 ∼ 06/11/05 | 16.67 | 16.71 | 25.48 | 24.37 | 23.93 | 27.55 |
| 07/03/05 ∼ 07/09/05 | 15.88 | 15.58 | 16.72 | 16.99 | 13.63 | 13.91 |
| 08/07/05 ∼ 08/13/05 | 10.70 | 10.69 | 11.36 | 12.96 | 12.84 | 15.21 |
| 09/04/05 ∼ 09/10/05 | 14.53 | 14.21 | 13.74 | 19.48 | 15.91 | 18.95 |
| 10/02/05 ∼ 10/08/05 | 18.22 | 18.40 | 19.96 | 19.70 | 17.10 | 19.67 |
| 11/06/05 ∼ 11/12/05 | 13.98 | 13.79 | 14.58 | 31.93 | 18.35 | 28.49 |
| 12/04/05 ∼ 12/10/05 | 18.99 | 18.87 | 19.48 | 30.00 | 26.17 | 27.64 |
| 01/08/06 ∼ 01/14/06 | 13.08 | 13.11 | 12.04 | 31.80 | 14.27 | 14.99 |
| mean | 14.42 | 14.32 | 15.28 | 21.88 | 16.29 | 19.40 |

**Table 8:** Comparison of $\sigma_m(\%)$ of one-month-ahead predictions for 12 weeks

| date | HW7 | HW14 | STR | Naive1 | Naive2 | Naive3 |
|---|---|---|---|---|---|---|
| 02/06/05 ∼ 02/12/05 | 1.13 | 1.55 | 2.63 | 0.18 | 2.79 | 6.82 |
| 03/06/05 ∼ 03/12/05 | 3.64 | 3.33 | 4.11 | 0.47 | 0.46 | 1.01 |
| 04/03/05 ∼ 04/09/05 | 3.02 | 2.06 | 3.49 | 0.46 | 2.25 | 1.08 |
| 05/08/05 ∼ 05/14/05 | 1.22 | 1.18 | 1.22 | 0.40 | 1.38 | 0.65 |
| 06/05/05 ∼ 06/11/05 | 4.59 | 4.28 | 25.41 | 0.36 | 5.96 | 0.62 |
| 07/03/05 ∼ 07/09/05 | 3.95 | 4.50 | 3.58 | 1.15 | 2.90 | 0.89 |
| 08/07/05 ∼ 08/13/05 | 0.35 | 0.36 | 1.03 | 0.62 | 1.24 | 0.58 |
| 09/04/05 ∼ 09/10/05 | 3.22 | 3.82 | 2.82 | 0.92 | 0.50 | 0.46 |
| 10/02/05 ∼ 10/08/05 | 4.20 | 4.01 | 6.80 | 0.80 | 2.77 | 1.95 |
| 11/06/05 ∼ 11/12/05 | 1.09 | 1.21 | 1.62 | 1.51 | 3.47 | 3.77 |
| 12/04/05 ∼ 12/10/05 | 2.40 | 2.21 | 3.29 | 0.54 | 4.33 | 1.82 |
| 01/08/06 ∼ 01/14/06 | 6.04 | 6.25 | 3.44 | 4.16 | 1.83 | 1.76 |
| mean | 2.90 | 2.90 | 4.95 | 0.96 | 2.49 | 1.78 |

In summary, our prediction methods without a exploratory variable—Holt-Winters and structural model—outperform all of ARX, ARIMA and the naive method in both one-day-ahead and one-week-ahead predictions. STL is competent with ARX and ARIMA in one-day-ahead predictions, and performs better in one-week-ahead predictions. Our prediction methods have a great advantage in the longer horizon predictions spanning days to weeks.

## *4.5 Some Discussions about Modeling and Prediction*

In this section, we discuss about some extensions of our modeling and prediction of electricity price curves. The restriction of our method is also discussed.

### 4.5.1 Modeling and Prediction with New Historical Price Curves

It is not necessary to build a new model whenever new historical price curves are coming. Denote the new historical price curve as $x_0$. The procedure of computing the low-dimensional feature vector of $x_0$ is as follows.

1. Identify the $k$ nearest neighbors of the new data point $x_0$ among $x_1, \cdots, x_N$. Let $N_0$ denote the set of the indices of the $k$ nearest neighbors of $x_0$.

2. Compute the linear weights $w_j$ which best reconstruct $x_0$ from its neighbors, i.e., minimize the following objective function,

$$E(w) = \|x_0 - \sum_{j \in N_0} w_j x_j\|^2, \tag{5.37}$$

   subject to the sum-to-one constraint, $\sum_{j \in N_0} w_j = 1$.

3. The low-dimensional feature vector $y_0$ is computed by $y_0 = \sum_{j \in N_0} w_j y_j$.

For the prediction of each dimension in the low-dimensional space, the original prediction models can still be employed. Therefore, our modeling and prediction of electricity price curves can be utilized online for real forecasting.

### 4.5.2 Weekday and Weekend Effect

Electricity price has different daily profiles, in particular, weekdays verses weekend. To detect the significance of the weekday and weekend effect, we can add the dummy variables, e.g., Saturday and Sunday, into our prediction method in the same fashion as electricity loads. We did the numerical experiments, but the prediction results are almost the same as those without the dummy variables. The reason may be that the effect of weekdays and weekend is mostly captured by the weekly seasonal and the effect of electricity loads.

### 4.5.3 Effects of Other Factors, e.g., Katrina and Rita Hit and Higher Prices for Natural Gas

Our prediction method can be extended to incorporate other factors which affect the price curve dynamics. For the irregularly occurring event, e.g., Katrina and Rita hits, the invention analysis can be considered in the prediction in low-dimensional space. For the effect of the high prices of natural gas, our prediction method can include the

natural gas price in the same manner as it does with the electricity load. Exploration along these directions is left for future research.

### 4.5.4 Restriction of Our Method

Our model captures the price spike aspect of the electricity price curves but does not focus on the prediction of the extreme spikes that are likely caused by one-of-a-kind events. For instance, the historical data set used for calibrating the forward price curve model in the New York area from Feb 2003 to Jan 2006 includes all price spikes but one outlier. This implies that the calibrated forward price curve model is capable of predicting price spikes that are of certain stationarity nature. As for the extreme spikes resulting from one-of-a-kind events, they shall not be viewed as being sampled from an embedded low-dimensional intrinsic manifold structure, thus they can be removed from the calibration data set. However, if such extreme price spikes were caused by changes to the fundamental structure of aggregate supply and demand, then the intrinsic dimension of the low-dimensional manifold would change accordingly, and yield a different set of major factors of the price dynamics in a low-dimensional space.

## *4.6 Conclusion*

We apply manifold-based dimension reduction to electricity price curve modeling. LLE is demonstrated to be an efficient method for extracting the intrinsic low-dimensional structure of electricity price curves. Using price data taken from the NYISO, we find that there exists a low-dimensional manifold representation of the day-ahead price curve in NYPP, and specifically, the dimension of the manifold is around four. The interpretation of each dimension and the cluster analysis in the low-dimensional space are given to analyze the main factors of the price curve dynamics. Numerical experiments show that our prediction preforms well for the short-term prediction, and our method also facilitates medium-term prediction, which is difficult,

even infeasible for other methods.

## 4.7 Appendix

Appendix A

The procedure of identifying ARIMA model follows paper [21]. The history data we applied to train the model are 90 days. The model is as follows [3],

$$
\begin{aligned}
&(1-\phi_1 B^1-\phi_2 B^2)\\
&\times(1-\phi_{23}B^{23}-\phi_{24}B^{24}-\phi_{47}B^{47}-\phi_{48}B^{48}\\
&-\phi_{72}B^{72}-\phi_{96}B^{96}-\phi_{120}B^{120}-\phi_{144}B^{144}\\
&\times(1-\phi_{168}B^{168}-\phi_{336}B^{336}-\phi_{504}B^{504})\\
&\times(1-B)(1-\phi_{24}B^{24})(1-\phi_{168}B^{168})log(price_t)\\
&=c+(1-\theta_1 B^1-\theta_2 B^2-\theta_3 B^3-\theta_4 B^4-\theta_5 B^5)\\
&\times(1-\theta_{24}B^{24}-\theta_{48}B^{48})\\
&\times(1-\theta_{168}B^{168}-\theta_{336}B^{336}-\theta_{504}B^{504})\varepsilon_t.
\end{aligned}
$$

The model estimation and prediction is implemented through the SCA system.

Appendix B

The ARX model with explanatory variable load follows paper [19]. The history data we applied to train the model are 45 days, as the relationship between prices and

[3] Occasionally, we slightly change the model when it does not converge.

loads is dynamic. The model is as follows,

$$
\begin{aligned}
&log(price_t) = c + (u_1B^1 + u_2B^2 + u_3B^3 + u_{24}B^{24} \\
&+u_{25}B^{25} + u_{48}B^{48} + u_{49}B^{49} + u_{72}B^{72} + u_{73}B^{73} \\
&+u_{96}B^{96} + u_{97}B^{97} + u_{120}B^{120} + u_{121}B^{121} + u_{144}B^{144} \\
&+u_{145}B^{145} + u_{168}B^{168} + u_{169}B^{169} + u_{192}B^{192} \\
&+u_{193}B^{193})log(price_t) + (v_1B^1 + v_2B^2 + v_3B^3 \\
&+v_{24}B^{24} + v_{25}B^{25} + v_{48}B^{48} + v_{49}B^{49} + v_{72}B^{72} \\
&+v_{73}B^{73} + v_{96}B^{96} + v_{97}B^{97} + v_{120}B^{120} + v_{121}B^{121} \\
&+v_{144}B^{144} + v_{145}B^{145} + v_{168}B^{168} + v_{169}B^{169} \\
&+v_{192}B^{192} + v_{193}B^{193})log(load_t) + \varepsilon_t.
\end{aligned}
$$

The model estimation and prediction are implemented in MATLAB.

# CHAPTER V

# A HESSIAN REGULARIZED NONLINEAR TIME-SERIES MODEL (HRM)

## *5.1 Introduction*

Consider a univariate time series $X_1, X_2, \cdots, X_n$, $n \geq 1$. A nonlinear data generation mechanism can be written as

$$X_t = f(X_{t-1}, \cdots, X_{t-p}) + \varepsilon_t, \tag{1.38}$$

for $p+1 \leq t \leq n$, and i.i.d. $\varepsilon_t$'s. We assume that $E(\varepsilon_t | X_{t-1}, \cdots, X_{t-p}) = 0$. Function $f$ has $p$ variables. We assume that $f$ has square integrable second partial derivatives. To simplify the notation, denote $\mathbf{Z}_{t-1} = (X_{t-1}, \cdots, X_{t-p})^T \in \mathbb{R}^p$; vector $\mathbf{z} = (z_1, \cdots, z_p)^T \in \mathbb{R}^p$ is a generic $p$-dimensional vector. Let $f_{ij}(z) = \frac{\partial^2 f(z)}{\partial z_i \partial z_j}$. Recall the integrated Hessian of function $f$ is defined as

$$\mathcal{H}f = \int_\Omega \sum_{i,j} |f_{ij}(z)|^2 dz,$$

where subset $\Omega \subset \mathbb{R}$ is the support of function $f$. Because $f$ has square integrable second derivatives, we have $\mathcal{H}f < \infty$.

Model (1.38) encompasses many nonlinear time-series models. Readers can compare it with models, e.g., functional coefficient autoregressive model [14], functional coefficient autoregressive model and its adaptive version [8, 42], threshold autoregressive model, multivariate local polynomial regression model [37, 17, 18] and many more. Book [41] provides an excellent overview. [Note some of these models may not satisfy the condition of $\mathcal{H}f < \infty$. For example, the threshold autoregressive model does not have second derivative on the boundary. This analytical shortcoming does not prevent us from developing a numerical approximation.]

We assume that $\Omega \subset \mathbb{R}^p$ is compact. Let $\mathcal{F} = \{f : \mathcal{H}f < \infty, \text{and support } f \subset \Omega\}$. The estimated $\hat{f}$ is the minimizer of the following objective function:

$$\min_{f \in \mathcal{F}} \sum_{t=p+1}^{N} [X_t - f(\mathbf{Z}_{t-1})]^2 + \lambda \mathcal{H}f. \tag{1.39}$$

Adding a penalty term $\mathcal{H}f$ is a standard approach in regularization. Function $\mathcal{H}f$ is called "bending energy" in deriving the thin-plate spline [111]. Note that if $p = 1$, then the minimizer in (1.39) is the natural cubic spline, whose knots are at $\mathbf{Z}_t$'s, $p \leq t \leq n-1$. In this sense, our approach can be considered as an extension of the natural cubic spline to high dimension. Also note that if $p = 2$ or 3, then the solution to (1.39) is the thin plate spline [111]. However, the reproducible kernel Hilbert space approach in [111] does not provide analytical solutions for $p \geq 4$ [52, Section 7.9].

Rather than finding the analytical solution for problem (1.39), which may not uniquely exist for $p \geq 4$, we propose a hessian regularized nonlinear time-series model (HRM) in this chapter, for which a numerical approximation to integrated hessian functional is given and a solution through local method is suggested. Theoretical results with respect to the convergence of our solution and the choice of the penalty function $\lambda$ are introduced. We also discuss a fast computing approach to our model. Our simulations not only verify the theoretical results, but also show the powerful predictability of our model by the comparison with many other models.

An alternative penalty is the Laplacian:

$$\mathcal{L}f = \int_{\Omega} \sum_{i} |f_{ii}(z)|^2 dz.$$

The difference between the Hessian ($\mathcal{H}f$) and Laplacian ($\mathcal{L}f$) is that the latter does not consider the cross terms. It is known that $\mathcal{H}f$ is translation and rotation invariant, while $\mathcal{L}f$ is not. Hence, we prefer the Hessian. Another interesting alternative is to consider a modified function:

$$\int_{\Omega} \sum_{i,j} |f_{ij}(z)| dz. \tag{1.40}$$

Note the above penalty function uses sum of absolute values, instead of sum of squares. Penalty function (1.40) may have some nice property. However, it is known that minimizing the sum of absolute values (i.e., the $\ell_1$ norm) is much more computational demanding than minimizing the sum of squares (i.e., the $\ell_2$ norm). A penalty function like (1.40) has been explored in triograms [68].

We discuss more about the solutions to problem (1.39). If we are willing to impose boundary conditions to $f$, using the integration-by-part argument that has been utilized for natural cubic spline, we can argue that $f$ is the "unique" minimizer, subject to a biharmonic function addition, if $f$ interpolates at every point.

The rest of this chapter is organized as follows. Section 5.2 derives a numerical approximation to the functional, given an analytical solution is not available. Section 5.3 derives a theorem that reveals some conditions under which our proposed method should work. Section 5.4 discusses issues related to how to choose an optimal value of $\lambda$. Section 5.5 introduces a fast computing approach. Section 5.6 presents numerical results for some simulations and real data analysis. Section 5.7 comes to the conclusion.

## *5.2 Numerical Approximation to Hessian*

Problem (1.39) does not have an analytical solution, unless in special cases (i.e., when $p = 1$, natural cubic spline; when $p = 2, 3$, thin plate spline). We propose a numerical approach that mimics problem (1.39). The key idea is to introduce a least squares estimator of the Hessian matrix $\mathcal{H}f(\mathbf{Z}_{t-1})$ at locations $\mathbf{Z}_{t-1}, t = p+1, p+2, \cdots, n$.

Because $\mathbf{Z}_t$'s are random variables, we slightly modify the original hessian functional $\mathcal{H}f$, so that the density function of $\mathbf{Z}_t$'s can be involved. The new hessian functional is:

$$\int_{\Omega} \sum_{i,j} |f_{ij}(z)|^2 g(z) dz, \tag{2.41}$$

where $g(z)$ is the density function of $z$. Actually, $\mathcal{H}f$ is a special case of (2.41), if $z$

is uniformly distributed. Thus, objective function (1.39) becomes

$$\min_{f\in\mathcal{F}} \sum_{t=p+1}^{N} [X_t - f(\mathbf{Z}_{t-1})]^2 + \lambda \int_{\Omega} \sum_{i,j} |f_{ij}(z)|^2 g(z) dz \tag{2.42}$$

An unbiased estimator of functional (2.41) is $\frac{1}{n-p}\sum_{t=p+1}^{n} \|\mathcal{H}f(\mathbf{Z}_{t-1})\|_F^2$, where $\|\cdot\|_F$ denotes the Frobenius norm (or Euclidean norm) of a matrix. Therefore, a numerical approximation of problem (2.42) is

$$\min_{f\in\mathcal{F}} \sum_{t=p+1}^{N} [X_t - f(\mathbf{Z}_{t-1})]^2 + \lambda \sum_{t=p+1}^{n} \|\mathcal{H}f(\mathbf{Z}_{t-1})\|_F^2. \tag{2.43}$$

Recall we have $\mathbf{Z}_{t-1} = (X_{t-1}, \cdots, X_{t-p})^T \in \mathbb{R}^p$. Consider a set $\mathcal{V} = \{\mathbf{Z}_{t-1}, p+1 \leq t \leq n\}$ is a collection of $(n-p)$ $p$-dimensional vectors. Assume $\mathbf{V}_0 = \mathbf{Z}_{t-1}$, for $p+1 \leq t \leq n$. Let $\mathbf{V}_i, i = 1, 2, \cdots, k$, denote the $k$ ($k \geq 1$) nearest neighbors of $\mathbf{V}_0$, while $\mathbf{V}_i \in \mathcal{V}$. Let $\bar{\mathbf{V}} = \frac{1}{k+1}\sum_{i=0}^{k} \mathbf{V}_i$, i.e., $\bar{\mathbf{V}}$ is the average of the $k+1$ vectors. A Taylor expansion at point $\bar{\mathbf{V}}$ generates the following approximation

$$f(\mathbf{V}_i) \approx f(\bar{\mathbf{V}}) + (\mathbf{V}_i - \bar{\mathbf{V}})^T \mathcal{J}f(\bar{\mathbf{V}}) + \frac{1}{2}(\mathbf{V}_i - \bar{\mathbf{V}})^T \mathcal{H}f(\bar{\mathbf{V}})(\mathbf{V}_i - \bar{\mathbf{V}}),$$
$$i = 0, 1, \cdots, k,$$

where $f(\bar{\mathbf{V}})$ is the value of function $f$ at location $\bar{\mathbf{V}}$, $\mathcal{J}f(\bar{\mathbf{V}})$ is the Jacobian at $\bar{\mathbf{V}}$, and $\mathcal{H}f(\bar{\mathbf{V}})$ is the Hessian matrix at $\bar{\mathbf{V}}$. Note we have $\mathcal{J}f(\bar{\mathbf{V}}) \in \mathbb{R}^p$ and $\mathcal{H}f(\bar{\mathbf{V}}) \in \mathbb{R}^{p\times p}$.

If $f$ is analytical, then the above approximation is close. A matrix version of the above approximation is

$$\mathbf{f}^* \approx \mathbf{1}_{k+1} \cdot c + \mathbf{V} \cdot \mathbf{J} + \frac{1}{2}\mathbf{C} \cdot \mathbf{H}, \tag{2.44}$$

where

$$\mathbf{f}^* = (f(\mathbf{V}_0), f(\mathbf{V}_1), \cdots, f(\mathbf{V}_k))^T \in \mathbb{R}^{k+1},$$

$$\mathbf{1}_{k+1} = (1, \cdots, 1)^T \in \mathbb{R}^{k+1},$$

$c$ is a constant; the $i$th ($1 \leq i \leq k+1$) row of matrix $\mathbf{V}$, $\mathbf{V} \in \mathbb{R}^{(k+1)\times p}$, is $(\mathbf{V}_{i-1} - \bar{\mathbf{V}})^T$; vector $\mathbf{J} \in \mathbb{R}^p$ is the Jacobian ($J_i = f_i(\bar{\mathbf{V}})$) at $\bar{\mathbf{V}}$. The $i$th row of matrix $\mathbf{C}$, $\mathbf{C} \in$

$\mathbb{R}^{(k+1)\times\frac{p^2+p}{2}}$, is $\mathcal{V}_1[(\mathbf{V}_i-\bar{\mathbf{V}})(\mathbf{V}_i-\bar{\mathbf{V}})^T]$, where for an arbitrary column vector $\mathbf{x}=(x_1,x_2,\ldots,x_p)^T$, we define $\mathcal{V}_1[\mathbf{x}\cdot\mathbf{x}^T]=(x_1^2,x_2^2,\ldots,x_p^2,\sqrt{2}x_1x_2,\ldots,\sqrt{2}x_1x_p,\sqrt{2}x_2x_3,\ldots,\sqrt{2}x_2x_p,\ldots,\sqrt{2}x_{p-1}x_p)\in\mathbb{R}^{\frac{p^2+p}{2}}$. Vector $\mathbf{H}$ is a vectorization with respect to the Hessian matrix at $\bar{\mathbf{V}}$, after eliminating identical entries: $\mathbf{H}=\mathcal{V}_2[\mathcal{H}f(\bar{\mathbf{V}})]$, where for a symmetric matrix $\mathbf{S}=(S_{ij})\in\mathbb{R}^{p\times p}$, we have $\mathcal{V}_2[\mathbf{S}]=(S_{11},S_{22},\ldots,S_{pp},\sqrt{2}S_{12},\sqrt{2}S_{13},\ldots,\sqrt{2}S_{1p},\sqrt{2}S_{23},\ldots,\sqrt{2}S_{2p},\ldots,\sqrt{2}S_{p-1,p})^T$. It is a standard exercise to verify that $\mathbf{1}^T\mathbf{V}=\mathbf{0}$.

A partial implementation of QR-decomposition (via, e.g., a modified Gram-Schmidt algorithm) can produce

$$\left[\begin{array}{ccc}\mathbf{1}_{k+1} & \mathbf{V} & \frac{1}{2}\mathbf{C}\end{array}\right]=\left[\begin{array}{cc}\mathbf{Q}_1 & \mathbf{Q}_2\end{array}\right]\left[\begin{array}{cc}\mathbf{R}_{11} & \mathbf{R}_{12}\\ \mathbf{0} & \mathbf{I}_{(p^2+p)/2}\end{array}\right], \tag{2.45}$$

where columns of $\mathbf{Q}_1\in\mathbb{R}^{(k+1)\times(p+1)}$ are orthonormal ($\mathbf{Q}_1^T\mathbf{Q}_1=\mathbf{I}_{p+1}$), and columns of $\mathbf{Q}_2\in\mathbb{R}^{(k+1)\times\frac{p^2+p}{2}}$ are orthogonal to the columns of $\mathbf{Q}_1$ (i.e., $\mathbf{Q}_2^T\mathbf{Q}_1=\mathbf{0}$).

From (2.44), we have

$$\begin{aligned}\mathbf{Q}_2^T\mathbf{f}^* &= \left(\begin{array}{cc}\mathbf{0} & \mathbf{Q}_2^T\mathbf{Q}_2\end{array}\right)\left[\begin{array}{cc}\mathbf{R}_{11} & \mathbf{R}_{12}\\ \mathbf{0} & \mathbf{I}_{(p^2+p)/2}\end{array}\right]\left[\begin{array}{c}c\\ \mathbf{J}\\ \mathbf{H}\end{array}\right]\\ &= \mathbf{Q}_2^T\mathbf{Q}_2\mathbf{H}.\end{aligned}$$

Hence, a least-squares estimator of $\mathbf{H}$ is

$$\hat{\mathbf{H}}=(\mathbf{Q}_2^T\mathbf{Q}_2)^+\mathbf{Q}_2^T\mathbf{f}^*,$$

where $(\cdot)^+$ denotes a pseudo-inverse of a matrix.

For the local Hessian matrix, we have

$$\begin{aligned}\|\hat{\mathcal{H}}f(\mathbf{Z}_{t-1})\|_F^2 &= \|\hat{\mathbf{H}}\|_2^2=\hat{\mathbf{H}}^T\hat{\mathbf{H}}\\ &= (\mathbf{f}^*)^T\mathbf{Q}_2(\mathbf{Q}_2^T\mathbf{Q}_2)^+(\mathbf{Q}_2^T\mathbf{Q}_2)^+\mathbf{Q}_2^T\mathbf{f}^*.\end{aligned}$$

To construct the matrix form of (2.43), we introduce the following notation:

$$\mathbf{K}_{t-1} = \mathbf{Q}_2(\mathbf{Q}_2^T\mathbf{Q}_2)^+(\mathbf{Q}_2^T\mathbf{Q}_2)^+\mathbf{Q}_2^T.$$

We also bring in a selection matrix $\mathbf{S}_{t-1}, p+1 \leq t \leq n$. Matrix $\mathbf{S}_{t-1}, \mathbf{S}_{t-1} \in \mathbb{R}^{(k+1)\times(n-p)}$, is made by two possible components: 0 and 1. For $\mathbf{V}_0, \mathbf{V}_1, \cdots, \mathbf{V}_k$ and $\mathbf{Z}_p, \cdots, \mathbf{Z}_{n-1}$ that are defined before, $\mathbf{S}_{t-1}$ satisfies

$$(\mathbf{V}_0, \mathbf{V}_1, \cdots, \mathbf{V}_k) = (\mathbf{Z}_p, \mathbf{Z}_{p+1}, \cdots, \mathbf{Z}_{n-1})\mathbf{S}_{t-1}^T, \forall t.$$

Apparently we have $\mathbf{f}^* = \mathbf{S}_{t-1}\mathbf{f}$, where $\mathbf{f} = (f(\mathbf{Z}_p), f(\mathbf{Z}_{p+1}), \cdots, f(\mathbf{Z}_{n-1}))^T$. We have

$$\sum_{t=p+1}^{n} \|\hat{\mathcal{H}} f(\mathbf{Z}_{t-1})\|_F^2 = \sum_{t=p}^{n-1} (\mathbf{f}^T\mathbf{S}_t^T\mathbf{K}_t\mathbf{S}_t\mathbf{f}).$$

Let $\mathbf{M} = (\mathbf{S}_p^T, \cdots, \mathbf{S}_{n-1}^T)\text{diag}\{\mathbf{K}_p, \mathbf{K}_{p+1}, \cdots, \mathbf{K}_{n-1}\}\begin{pmatrix} \mathbf{S}_p \\ \vdots \\ \mathbf{S}_{n-1} \end{pmatrix}$, we have

$$\sum_{t=p+1}^{n} \|\hat{\mathcal{H}} f(\mathbf{Z}_{t-1})\|_F^2 = \mathbf{f}^T\mathbf{M}\mathbf{f},$$

which is a quadratic function of $\mathbf{f}$.

Problem (2.43) becomes

$$\min_{\mathbf{f}} \|\mathbf{Y} - \mathbf{f}\|_2^2 + \lambda \mathbf{f}^T\mathbf{M}\mathbf{f},$$

where variable $\mathbf{f} \in \mathbb{R}^{n-p}$, vector $\mathbf{Y} = (X_{p+1}, \cdots, X_n)^T \in \mathbb{R}^{n-p}$, and $\mathbf{M}$ is derived before. The least squares estimator of $\mathbf{f}$ becomes

$$\hat{\mathbf{f}} = (\mathbf{I}_{n-p} + \lambda \cdot \mathbf{M})^{-1} \cdot \mathbf{Y}. \tag{2.46}$$

### 5.2.1 Null Space of Matrix M

The estimator in (2.46) requires inverting an $(n-p) \times (n-p)$ matrix, which can be challenging. It is easy to verify that matrix $\mathbf{M}$ is positive-semidefinite. Moreover,

matrix $\mathbf{M}$ has eigenvalue 0, whose multiplicity is at least $p+1$, conditioning that the following matrix is of full column rank:

$$\begin{pmatrix} 1 & \mathbf{Z}_p^T \\ \vdots & \vdots \\ 1 & \mathbf{Z}_{n-1}^T \end{pmatrix}.$$

As a matter of fact, every column of the above matrix solves the equation $\mathbf{M} \cdot \mathbf{x} = 0$.

### 5.2.2 Prediction

We would like to estimate $f(\mathbf{Z})$ at a new point $\mathbf{Z} \in \mathbb{R}^p$. First we identify the $k+1$ ($k \geq 1$) nearest neighbors of $\mathbf{Z}$ for the vectors in the set $\mathcal{V}$. The reason for choosing the $k+1$ instead of $k$ nearest neighbors is that we want a similar expression in the prediction step as in the estimation step. Recall $\mathcal{V}$ contains all the $p$-dimensional vectors generated by a scanning window going through the time series. Without loss of generality, let $\mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_{k+1}$ denote the $k+1$ nearest neighbors. Let $\bar{\mathbf{V}}$ denote the average: $\bar{\mathbf{V}} = \frac{1}{k+1}\sum_{i=1}^{k+1} \mathbf{V}_i$. Recall $\mathcal{J}f(\bar{\mathbf{V}})$ denotes the Jacobian at $\bar{\mathbf{V}}$, and $\mathcal{H}f(\bar{\mathbf{V}})$ denotes the Hessian matrix at $\bar{\mathbf{V}}$. A second order approximation via Taylor expansion at point $\bar{\mathbf{V}}$ yields

$$f(\mathbf{V}_i) - f(\bar{\mathbf{V}}) = (\mathbf{V}_i - \bar{\mathbf{V}})^T \mathcal{J}f(\bar{\mathbf{V}}) + \frac{1}{2}(\mathbf{V}_i - \bar{\mathbf{V}})^T \mathcal{H}f(\bar{\mathbf{V}})(\mathbf{V}_i - \bar{\mathbf{V}}).$$

Recall $\hat{f}(\mathbf{V}_1), \cdots, \hat{f}(\mathbf{V}_{k+1})$ are the fitted values at $\mathbf{V}_1, \cdots, \mathbf{V}_{k+1}$. Similar to the analysis in establishing the least squares estimators for Hessian, we have the following equation:

$$\begin{aligned} \begin{pmatrix} \hat{f}(\mathbf{V}_1) \\ \vdots \\ \hat{f}(\mathbf{V}_{k+1}) \end{pmatrix} &= \mathbf{1}_{k+1} f(\bar{\mathbf{V}}) + \begin{pmatrix} (\mathbf{V}_1 - \bar{\mathbf{V}})^T \\ \vdots \\ (\mathbf{V}_{k+1} - \bar{\mathbf{V}})^T \end{pmatrix} \mathcal{J}f(\bar{\mathbf{V}}) \\ &\quad + \frac{1}{2} \begin{pmatrix} \mathcal{V}_1[(\mathbf{V}_1 - \bar{\mathbf{V}})(\mathbf{V}_1 - \bar{\mathbf{V}})^T] \\ \vdots \\ \mathcal{V}_1[(\mathbf{V}_{k+1} - \bar{\mathbf{V}})(\mathbf{V}_{k+1} - \bar{\mathbf{V}})^T] \end{pmatrix} \mathcal{V}_2[\mathcal{H}f(\bar{\mathbf{V}})], \end{aligned} \tag{2.47}$$

where vectorization operators $\mathcal{V}_1(\cdot)$ and $\mathcal{V}_2(\cdot)$ have been defined earlier. Note $f(\bar{\mathbf{V}})$ is a scalar, vector $\mathcal{J}f(\bar{\mathbf{V}})$ is $p$-dimensional, and matrix $\mathcal{H}f(\bar{\mathbf{V}})$ contains $p(p+1)/2$ unknown variables. If we have

$$k \geq p + \frac{p(p+1)}{2} = \frac{1}{2}(p+1)(p+2),$$

then least squares estimators can be established for $f(\bar{\mathbf{V}}), \mathcal{J}f(\bar{\mathbf{V}})$ and $\mathcal{H}f(\bar{\mathbf{V}})$ on the right hand side of (2.47). Hence, an estimated value of $f(\cdot)$ at $\mathbf{Z}$ is

$$\hat{f}(\mathbf{Z}) = \hat{f}(\bar{\mathbf{V}}) + (\mathbf{Z} - \bar{\mathbf{V}})^T \hat{\mathcal{J}} f(\bar{\mathbf{V}}) + \frac{1}{2}(\mathbf{Z} - \bar{\mathbf{V}})^T \hat{\mathcal{H}} f(\bar{\mathbf{V}})(\mathbf{Z} - \bar{\mathbf{V}}). \tag{2.48}$$

A variation of the above is to ignore the quadratic terms in the right hand sides of (2.47) and (2.48). Hence instead of a quadratic prediction, we adopt a linear prediction, where the first order approximation via Taylor expansion is applied, and the least squares estimators are established for $f(\bar{\mathbf{V}})$ and $\mathcal{J}f(\bar{\mathbf{V}})$. Thus, the prediction of $f(\cdot)$ at $\mathbf{Z}$ is

$$\hat{f}(\mathbf{Z}) = \hat{f}(\bar{\mathbf{V}}) + (\mathbf{Z} - \bar{\mathbf{V}})^T \hat{\mathcal{J}} f(\bar{\mathbf{V}}).$$

The simulation results show that the linear prediction is more robust than the quadratic prediction. Thus, in our numerical experiments, only the linear prediction is utilized.

## *5.3 A Convergence Theorem*

We have introduced a numerical approximation to (2.42). A question is that when there is an underlying function $f$, which is analytical, whether our estimator $\hat{\mathbf{f}}$ will indeed converge to this underlying function. In this section, we study quantity $\|\hat{\mathbf{f}}_n - \mathbf{f}\|_2^2/n$, where $\hat{\mathbf{f}}_n$ is the same as $\hat{\mathbf{f}}$ but with a subscript to integrate the length of the time series $n$, and vector $\mathbf{f}$ is the true value of the function at $\mathbf{Z}_t$'s. We will show that $\|\hat{\mathbf{f}}_n - \mathbf{f}\|_2^2/n \to 0$ is true under certain conditions that only depend on matrix $\mathbf{M}$ and underlying function $f(\cdot)$. These conditions can be verified numerically, hence

can be checked in simulations. Our conditions are analogous to the conditions for Sobolev space, which has played an important role in determining the optimal rate of estimation within a certain functional class [66].

Recall

$$\hat{\mathbf{f}}_n = (\mathbf{I}_{n-p} + \lambda_n \mathbf{M})^{-1}\mathbf{Y},$$

Consequently, we have

$$\hat{\mathbf{f}}_n - \mathbf{f} = (\mathbf{I} + \lambda_n \mathbf{M})^{-1}\mathbf{f} - \mathbf{f} + (\mathbf{I} + \lambda_n \mathbf{M})^{-1}\varepsilon.$$

where $\varepsilon = (\varepsilon_{p+1}, \cdots, \varepsilon_n)^T$.

Let $\mathbf{M} = \mathbf{U}^T\mathbf{D}\mathbf{U}$, the eigenvalue decomposition of matrix $\mathbf{M}$. Denote $\mathbf{D} = \text{diag}\{d_1, \cdots, d_{n-p}\}$. Let $f_i' = (\mathbf{U}\mathbf{f})_i$ and $\varepsilon_i' = (\mathbf{U}\varepsilon)_i$.

**Lemma 5.3.1** *There exists a constant $c_n$ (e.g. $c_n = 2\log n$), such that for $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$,*

$$Pr\{|\varepsilon_i'|^2 < c_n\sigma^2, \forall i\} \to 1, \text{ as } n \to \infty.$$

*The above is a well-known property of normally distributed random variables.*

We have the following inequality:

$$\begin{aligned}\frac{1}{2}\|\hat{\mathbf{f}}_n - \mathbf{f}\|_2^2 &\leq \sum_{i=1}^{n-p} \frac{(\lambda_n d_i)^2 (f_i')^2 + (\varepsilon_i')^2}{(1+\lambda_n d_i)^2} \\ &\leq \sum_{i=1}^{n-p} \frac{(\lambda_n d_i)^2 (f_i')^2 + c_n\sigma^2}{(1+\lambda_n d_i)^2}.\end{aligned}$$

The last inequality utilizes the preceding lemma. We consider a function: for $\alpha \geq 0$.

$$g(\alpha) = \frac{\alpha^2 (f_i')^2 + c_n\sigma^2}{(1+\alpha)^2}.$$

The following can be verified through elementary calculation.

1. $g(0) = c_n\sigma^2, g(\infty) = (f_i')^2$.

2. When $0 < \alpha < \frac{c_n\sigma^2}{(f_i')^2}$, we have $g'(\alpha) < 0$; when $\alpha > \frac{c_n\sigma^2}{(f_i')^2}$, we have $g'(\alpha) > 0$.

3. The minimum point is $g(\frac{c_n\sigma}{(f_i')^2}) = \frac{(c_n\sigma^2)(f_i')^2}{(f_i')^2+c_n\sigma^2}$.

4. If $(f_i')^2 < c_n\sigma^2$ and $\alpha > \frac{1}{\gamma}\frac{c_n\sigma^2}{(f_i')^2}$, where $\gamma \geq 1$, we have $g(\alpha) < \gamma(f_i')^2$.

5. If $c_n\sigma^2 < (f_i')^2$ and $\alpha < \gamma\frac{c_n\sigma^2}{(f_i')^2}$, where $\gamma \geq 1$, we have $g(\alpha) < \gamma \cdot c_n\sigma^2$.

Consider two quantities:

$$\begin{aligned} a_n &= \max_i \left\{ \frac{c_n\sigma^2}{(f_i')^2} \cdot \frac{1}{d_i} : \text{ for } i \text{ such that } (f_i')^2 < c_n\sigma^2 \right\}, \\ b_n &= \min_i \left\{ \frac{c_n\sigma^2}{(f_i')^2} \cdot \frac{1}{d_i} : \text{ for } i \text{ such that } (f_i')^2 > c_n\sigma^2 \right\}. \end{aligned}$$

It won't be interesting if $a_n \leq b_n$. We suppose that

$$\gamma_n = \sqrt{\frac{a_n}{b_n}} \geq 1.$$

We pick $\lambda_n = \frac{a_n}{\gamma_n} = b_n \cdot \gamma_n = \sqrt{a_n b_n}$. We have the following main result.

**Theorem 5.3.2** *For the aforementioned $\lambda_n$, we have*

$$\frac{1}{2}\|\hat{\mathbf{f}}_n - \mathbf{f}\|_2^2 \leq \gamma_n \cdot \sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2].$$

The proof is an application of the preceding analysis.

*Remark.* It is known that $\sum_{i=1}^{n-p}(f_i')^2/n = O(1)$, i.e., such a quantity tends to be a constant. If sequence $(f_i')^2$ decay, (for example, analogous to the behavior of $\ell_p$ norm), then $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]$ could have lower order than $O(n)$. It is possible that $n^{-1}\gamma_n \sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2] \to 0$. Conditions under which the above holds can be a delicate problem. In simulated examples, in almost all the cases, we observe that the sequence $|f_i'|$ (after being sorted at a decreasing order) decays like an inverse polynomial (i.e., $x^{-\beta}$ for $\beta > 0$). From the above theorem, our algorithm will work in those situations. Section 5.6.1 provides some examples where $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]/n$ does decay with the increasing $n$.

## 5.4 Choice of Penalty Parameter $\lambda$

In Section 5.3, a theoretically appropriation $\lambda_n$ is provided. However, in practice, the underlying function $f(\cdot)$ is not available; hence one can not utilize the theoretical formula. Generalized cross validation can be adopted. Consider the generalized cross validation function GCV($\lambda$):

$$\text{GCV}(\lambda) = \frac{1}{n-p} \sum_{k=p+1}^{n} \left( \frac{X_k - \hat{f}_\lambda(\mathbf{Z}_{k-1})}{1 - \frac{1}{n-p}\text{Tr}(\mathbf{A}(\lambda))} \right)^2, \tag{4.49}$$

where $\mathbf{A}(\lambda) = (\mathbf{I}_{n-p} + \lambda \mathbf{M})^{-1}$. The optimal value of the penalty parameter $\lambda$ can be estimated by minimizing the above GCV function. The justification is relatively straightforward and is relegated to Appendix 5.8.

The derivation of the GCV function uses an approximation, because the numerical approximation of Hessian functional is applied. The following provides more justification on applying generalized cross validation here. To facilitate the following analysis, let us recall some notations. Recall that the eigenvalue decomposition of matrix $\mathbf{M}$ is $\mathbf{M} = \mathbf{U}^T\mathbf{D}\mathbf{U}$, where $\mathbf{D} = \text{diag}\{d_1, \cdots, d_{n-p}\}$, and $0 \le d_1 \le d_2 \le \cdots \le d_{n-p}$. Recall that $\varepsilon_i' = (\mathbf{U}\varepsilon)_i$. We define $y_i' = (\mathbf{U}\mathbf{Y})_i$. Note vectors $\mathbf{f}, \mathbf{Y}, \varepsilon$ have been used in Section 5.3. If we know the true value of the function at every point (i.e., $\mathbf{f}$ is known), then the mean square error as a function of $\lambda$ is

$$\begin{aligned} \text{MSE}(\lambda) &= \frac{1}{n-p}\|\mathbf{f} - \mathbf{A}(\lambda)\mathbf{Y}\|_2^2 \\ &= \frac{1}{n-p}\|[\mathbf{I} - \mathbf{A}(\lambda)]\mathbf{Y} - \varepsilon\|_2^2 \\ &= \frac{1}{n-p}\{\|[\mathbf{I} - \mathbf{A}(\lambda)]\mathbf{Y}\|_2^2 + \|\varepsilon\|_2^2 - 2\varepsilon^T[\mathbf{I} - \mathbf{A}(\lambda)]\mathbf{Y}\} \\ &= \frac{1}{n-p}[\lambda^2 h_1(\lambda) + \|\varepsilon\|_2^2 - 2\lambda h_2(\lambda)], \end{aligned} \tag{4.50}$$

where

$$h_1(\lambda) = \sum_i \frac{d_i^2 (y_i')^2}{(1 + \lambda d_i)^2},$$

and

$$h_2(\lambda) = \sum_i \frac{d_i \cdot \varepsilon_i' \cdot y_i'}{1 + \lambda d_i}.$$

On the other hand, for GCV, we have

$$\mathrm{GCV}(\lambda) = (n - p)\frac{h_1(\lambda)}{[h_3(\lambda)]^2}, \tag{4.51}$$

where

$$h_3(\lambda) = \sum_i \frac{d_i}{1 + \lambda d_i}.$$

Hopefully, one can establish a quantitative connection between the minimizer of (4.50) and the minimizer of (4.51). This chapter has not pursued further. Note that if factor $\varepsilon_i' \cdot y_i'$ can be treated as a constant, there is a strong similarity between $h_2(\lambda)$ and $h_3(\lambda)$. In simulations, we plot $\mathrm{GCV}(\lambda)$ and $\mathrm{MSE}(\lambda)$ for multiple examples. In almost all the cases, we observe that the minimum of the two function are close. This in some sense validates the use of GCV to choose $\lambda$. More details are given in Section 5.6.2.

## *5.5 Fast Computing*

Recall that our estimate has the form $\hat{\mathbf{f}} = (\mathbf{I} + \lambda\mathbf{M})^{-1}\mathbf{Y}$. Such a solution bears strong similarity with the solution to the smoothing spline [52, Section 2.3]. It is well known that for smoothing spline, by taking advantage of a band matrix, fast computing is feasible ([52, Section 2.3.3] and [94]). A similar analysis can be developed for our method. It is not hard to observe that

$$(\mathbf{I} + \lambda\mathbf{M})\hat{\mathbf{f}} = \mathbf{Y}.$$

Denoting

$$\begin{aligned} \mathbf{B} &= \mathrm{diag}\{\mathbf{K}_p, \cdots, \mathbf{K}_{n-1}\}, \\ \mathbf{S} &= \begin{pmatrix} \mathbf{S}_p \\ \vdots \\ \mathbf{S}_{n-1} \end{pmatrix}, \end{aligned}$$

and $\mathbf{r} = \mathbf{BS}\hat{\mathbf{f}}$, we have

$$\begin{aligned} \hat{\mathbf{f}} &= \mathbf{Y} - \lambda \mathbf{M}\hat{\mathbf{f}} \\ &= \mathbf{Y} - \lambda \mathbf{S}^T\mathbf{BS}\hat{\mathbf{f}} \\ &= \mathbf{Y} - \lambda \mathbf{S}^T\mathbf{r}. \end{aligned} \tag{5.52}$$

The above leads to

$$\mathbf{B}^{-1}\mathbf{r} = \mathbf{S}\hat{\mathbf{f}} = \mathbf{SY} - \lambda \mathbf{SS}^T\mathbf{r},$$

which is equivalent to

$$(\mathbf{B}^{-1} + \lambda \mathbf{SS}^T)\mathbf{r} = \mathbf{SY}. \tag{5.53}$$

Matrix $\mathbf{B}^{-1}$ is blocky diagonal with $(k+1) \times (k+1)$ diagonal submatrices. If matrix $\mathbf{SS}^T$ is a blocky band matrix, then the standard technique in manipulating band matrices [52, Section 2.6] can be adopted. Hence an $O(nk^2)$ algorithm is available to find $\mathbf{r}$ via (5.53). Our estimate $\hat{\mathbf{f}}$ then can be obtained via (5.52). Overall complexity is still $O(nk^2)$.

Unfortunately, in the previous algorithm, matrix $\mathbf{SS}^T$ is not guaranteed to be a blocky band matrix. One can propose a modified version of our algorithm to facilitate fast computation. We discuss two possible approaches. Recall we are considering modeling $X_t = f(\mathbf{Z}_{t-1}) + \varepsilon_t$, $p+1 \leq t \leq n$. Let $\{P(p+1), \cdots, P(n)\}$ be a permutation of $p+1, \cdots, n-1, n$. It is equivalent to consider modeling with dataset $X_{P(t)} = f(\mathbf{Z}_{P(t)-1}) + \varepsilon_{P(t)}$, $p+1 \leq t \leq n$. Now we consider the distance matrix $\{D_{ij}\}_{(n-p)\times(n-p)}$, where $D_{ij} = \|\mathbf{Z}_{i+p-1} - \mathbf{Z}_{j+p-1}\|_2$, i.e., the $\ell_2$-distance (Euclidean distance). It is evident that $D_{ii} = 0$. For a fixed permutation $P$, define $\widetilde{D}_{ij}(P) = \|\mathbf{Z}_{P(i+p)-1} - \mathbf{Z}_{P(j+p)-1}\|_2$.

**Approach One:** Find permutation $P$ such that the quantity

$$\max\{\widetilde{D}_{ij}(P) : |i-j| \leq k\} \tag{5.54}$$

is minimized. Then "the $k$ nearest neighbors" of $\mathbf{Z}_t$ are specified to be the $k$ closest

elements of $P(t)$ in the sequence $\{P(p+1), \cdots, P(n)\}$. One can verify that the matrix $\mathbf{SS}^T$ is a blocky band matrix.

**Approach Two:** A drawback of Approach One is that the permutation $P$ may not be found easily. A compromise is to use some heuristic iterative algorithm to minimize the quantity in (5.54), and then still use the $k$ nearest neighbors in the Euclidean distance. Hopefully, the off-diagonal submatrix of $\mathbf{SS}^T$ becomes a zero matrix when it is far away from the diagonal. An example of such an algorithm is the Jacobi's method to find eigenvalues and eigenvectors [72]. The simulation study and further research is left as future work.

## *5.6 Numerical Experiments*

### 5.6.1 Simulations Regarding the Convergence Theorem

The following model is utilized to generate four times series:

$$X_t = f(X_{t-1}, X_{t-2}, X_{t-3}, X_{t-4}) + \varepsilon_t, \quad \text{for } t \geq 5,$$

where $\varepsilon_t \overset{\text{i.i.d.}}{\sim} N(0, \sigma^2)$, where $\sigma = 1$ for the first two time series, $\sigma = 0.2$ for the third time series, and $\sigma = 0.8$ for the last time series. Under the previous formulation, these are the cases when $p = 4$. The function $f(X_{t-1}, X_{t-2}, X_{t-3}, X_{t-4})$ is defined as:

- in the first time series, we have

$$f(x_1, x_2, x_3, x_4) = a_1 + a_2 + a_3 + a_4,$$

where

$$\begin{aligned} a_1 &= -x_2 e^{-x_2^2/2}, \\ a_2 &= \frac{x_1}{1 + x_2^2} \cos(1.5 x_2), \\ a_3 &= \frac{4 x_3}{1 + 0.8 x_3^2}, \\ a_4 &= \frac{e^{3(x_4 - 2)}}{1 + e^{3(x_4 - 2)}}; \end{aligned}$$

- in the second time series, we have

$$f(x_1, x_2, x_3, x_4) = \frac{2x_1}{1+0.8x_1^2} - \frac{2x_2}{1+0.8x_2^2} + \frac{2x_3}{1+0.8x_3^2} - \frac{2x_4}{1+0.8x_4^2};$$

- in the third time series, we have

$$f(x_1, x_2, x_3, x_4) = a_1 x_1 + a_2 x_2 + a_1 x_3 + a_2 x_4, \tag{6.55}$$

where

$$\begin{aligned} a_1 &= 0.2 + (0.3 + x_1)e^{-4x_1^2}, \\ a_2 &= -0.4 - (0.7 + 1.3x_1)e^{-4x_1^2}; \end{aligned}$$

- in the last time series, we have

$$f(x_1, x_2, x_3, x_4) = \frac{0.25x_4}{1+1.2x_1^2} - \frac{0.4x_1}{1+0.6x_2^2} + \frac{0.5x_2}{1+0.8x_3^2} - \frac{0.75x_3}{1+x_4^2} + \frac{e^{1.5(x_4-2)}}{1+e^{3(x_4-2)}}.$$

The simulated time series can be seen in Fig. 14.

For each time series model, time series with the different length ranging from 200 to 3000 are generated. The increment of the length of time series is 200 data points each time. Two quantities $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]/n$ and $\sum_{i=1}^{n-p}(f_i')^2/n$ are calculated, and illustrated for each time series in Fig.15[1]. Quantity $\sum_{i=1}^{n-p}(f_i')^2/n$ fluctuates around a constant with the different length of time series, while quantity $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]/n$ appears to decay as the length of the time series is increasing. The above observation verifies our conjecture to some degree: although $\sum_{i=1}^{n-p}(f_i')^2/n$ tends to be a constant, $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]/n$ could have lower order, i.e., it is possible that $n^{-1}\gamma_n \sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2] \to 0$.

### 5.6.2 Adoption of the Generalized Cross Validation Principle

Using simulations of the four time series in the last section, we study the relation between the minimizers of GCV$(\cdot)$ and MSE$(\cdot)$. Since the data generation mechanism

[1]Without normalization, $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]/n$ is always smaller than $\sum_{i=1}^{n-p}(f_i')^2/n$.

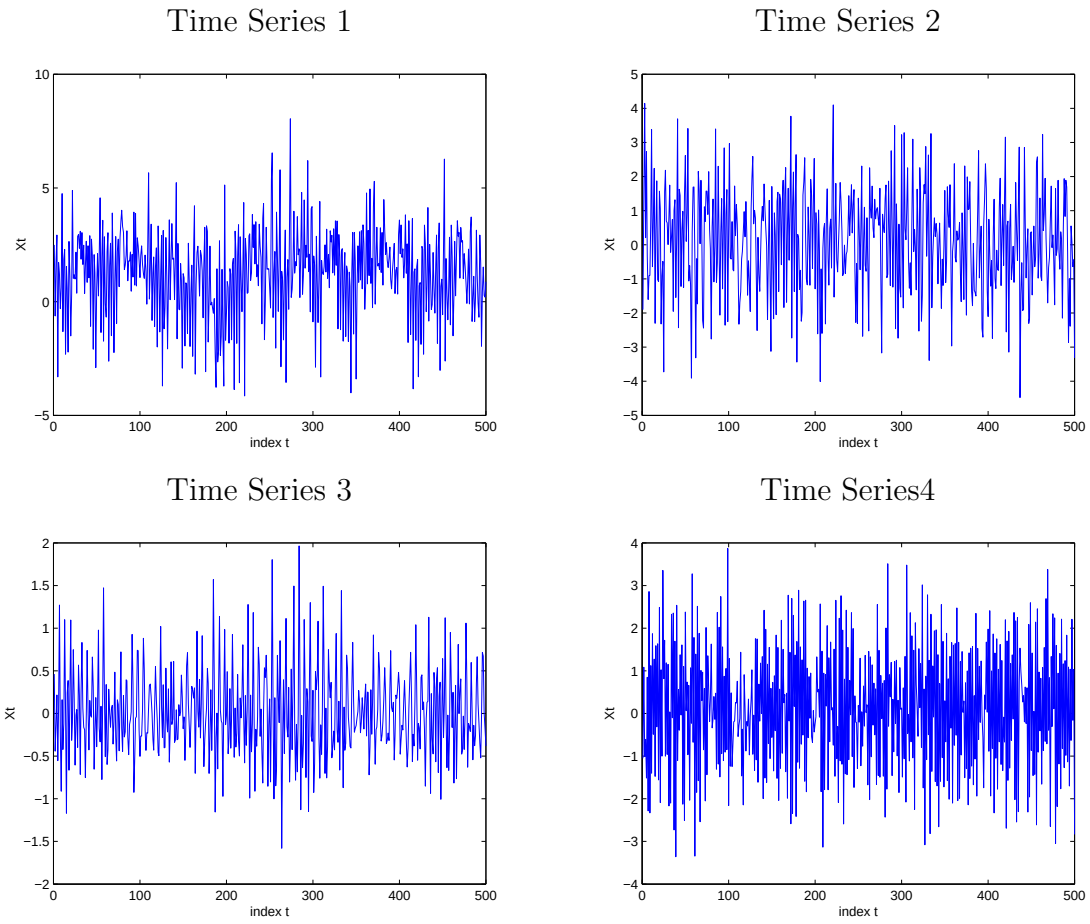



**Figure 14:** The simulated four time series with 500 data points. The data generation mechanism is described in Section 5.6.1.

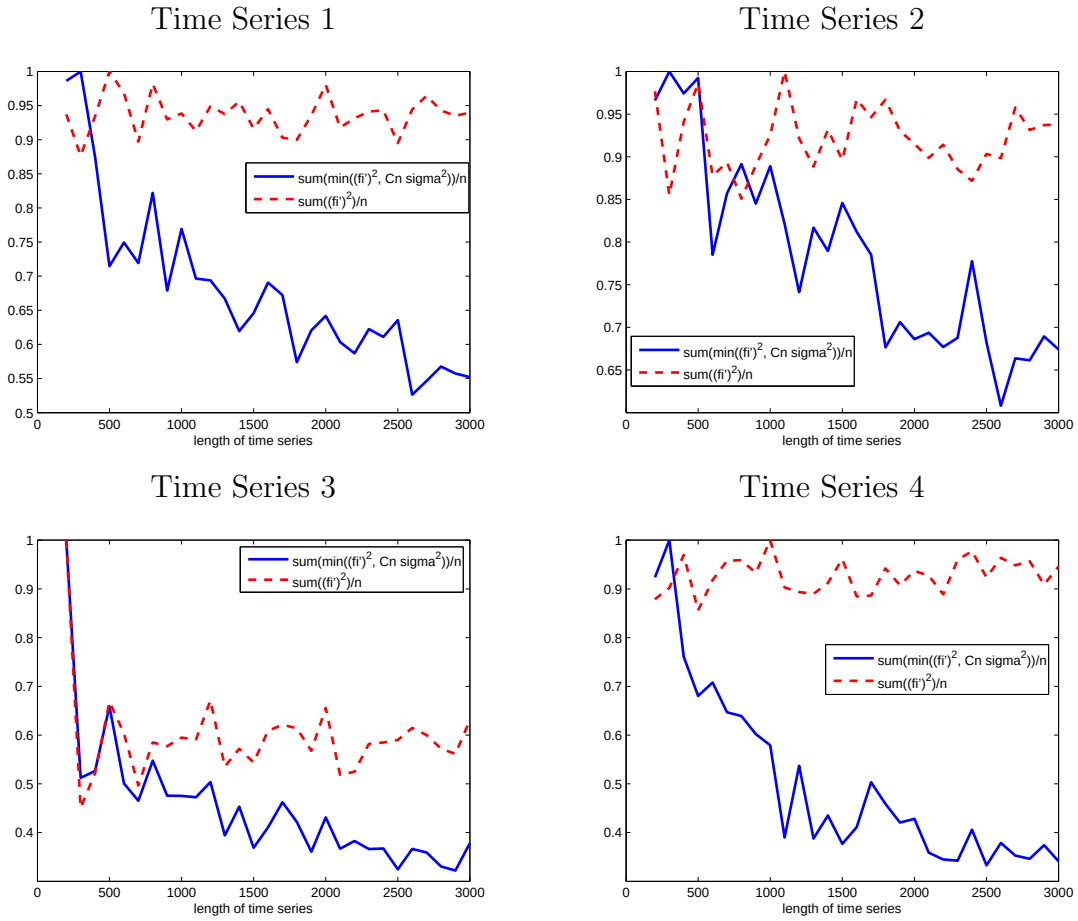



**Figure 15:** The trend of $\sum_{i=1}^{n-p} \min[(f_i')^2, c_n\sigma^2]/n$ and $\sum_{i=1}^{n-p}(f_i')^2/n$ as $n$ is increasing. The length of time series $n$ ranges from 200 to 3000. In order to compare two quantities more clearly, we normalize the two quantity sequences by deviding their maximal values, respectively .

is known, we can plot MSE($\lambda$) for a range of values for $\lambda$. Function MSE($\cdot$) is plotted against GCV($\cdot$) in Fig. 16. It is evident that the minimizer of GCV also renders a small value of MSE, which can be considered as a validation of using GCV function to choose optimal $\lambda$.

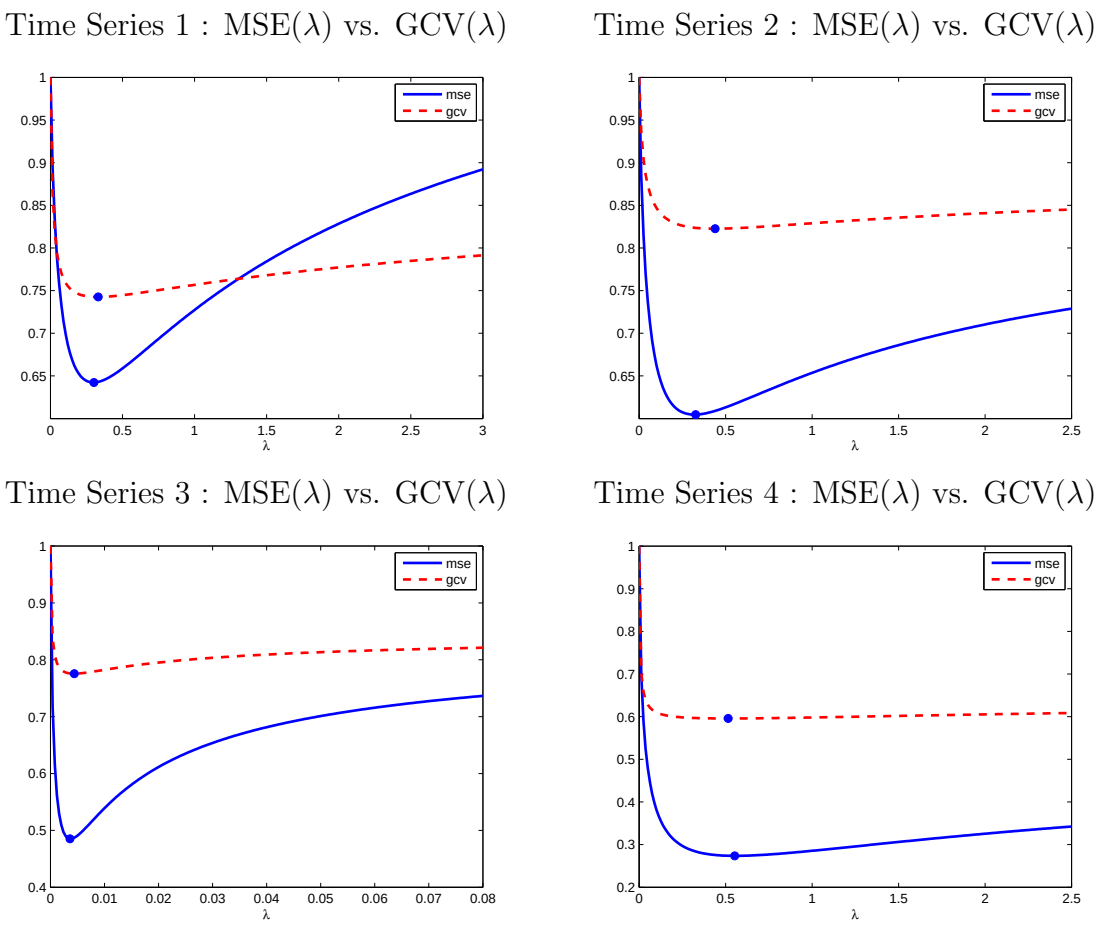



**Figure 16:** The functions GCV($\cdot$)and MSE($\cdot$) of the four time series. The GCV and MSE achieves minima at $(0.3317, 0.3015), (0.4397, 0.3266), (0.0044, 0.0036)$ and $(0.5151, 0.5528)$ respectively in the above four cases. The minima are marked with circles. For comparison, the maximal values of functions GCV and MSE are normalized to 1.

### 5.6.3 Synthetic Examples

This section contains three parts. The first two parts consist of simulation models, which are chosen from functional coefficient autoregressive model (FAR) and threshold autoregressive model (TAR), respectively. The last part consists of two

simulation models, which are nonlinear models and belong to none of FAR, TAR or additive autoregressive model (AAR). For each of the model, three types of prediction errors—one-step prediction errors, iterative two-step prediction errors, and direct two-step prediction errors—are computed for AAR, FAR, TAR, AR, Loess, Locpoly and our method. The difference between *iterative* two-step prediction and *direct* two-step prediction can be found in [41, Section 8.3.6]. The following is a brief introduction of the models we applied for comparison.

- AAR($p$): additive nonlinear autoregressive model with the embedding dimension $p$. The formula is

$$X_t = f_1(X_{t-1}) + ... + f_p(X_{t-p}) + \varepsilon_t.$$

- FAR($p, d$): functional coefficient autoregressive model [14] with $p$ lags and $X_{t-d}$ being the model dependent variable (see [41, page 318] for additional details). The formula is

$$X_t = f_1(X_{t-d})X_{t-1} + \cdots + f_p(X_{t-d})X_{t-p} + \varepsilon_t.$$

- AR($p$): autoregressive model with $p$ lags.

- TAR($p_1, p_1; d$): threshold autoregressive model [102, Section 3.3], where $p_1$ and $p_2$ are autoregressive orders for low and high regime respectively, and $d$ is the time delay or time lag for the threshold variable. The formula is

$$X_t = \begin{cases} b_{10} + b_{11}X_{t-1} + \cdots + b_{1p_1}X_{t-p_1} + \varepsilon_t, \text{ if } X_{t-d} \leq c, \\ b_{20} + b_{21}X_{t-1} + \cdots + b_{2p_2}X_{t-p_2} + \varepsilon_t, \text{ if } X_{t-d} > c. \end{cases}$$

- Loess($p$)(or Lowess($p$)): locally weighted scatterplot smoothing with $p$ covariates [17, 18]. It is a local polynomial regression with tricubic weighting.

- Locpoly($p$): multivariate local polynomial regression with Epanechnikov kernel. $p$ is the number of covariates [37, 113, 89].

All the models listed above, other than AR model, are nonlinear models. For Loess, Locpoly and our model, the first order (linear) prediction is utilized for all the numerical experiments.

In each experiment for each model, we produce 300 simulations. In each simulation, a time series with length 602 is generated from the simulation model. We make the one-step and two-step predictions based on the first 600 data points, and then calculate the three types of prediction errors by comparing with the observed values, i.e., the last two generated data points. The mean, median, and standard deviation of the absolute prediction errors are computed over the 300 simulations for the each type of prediction errors. [The statistics are denoted by mean, median, std respectively in the tables of this section.] The mean square prediction error (MSPE) for each type of prediction errors is also calculated. In all the simulations, we fix the number of the nearest neighbors in our method as $k = 20$.

The software for FAR was downloaded from

> http://orfe.princeton.edu/∼jqfan/fan/nls.html. (A supplement of [41].)

Implementation of TAR and AAR is based on an online software package that is downloadable at

> http://cran.r-project.org/src/contrib/Descriptions/tsDyn.html. (Maintainer: Antonio, Fabio Di Narzo.)

Implementation of Locpoly is based on an online software package that is downloadable at

> http://cran.r-project.org/src/contrib/Descriptions/JLLprod.html. (Maintainer: David Tom*ás*, Jacho-Ch*á*vez.)

Implementation of AR can be found in the Matlab system identification toolbox , and Loess is implemented based on the function “loess” in standard R package “stats”.

#### 5.6.3.1 A Functional Coefficient Autoregressive Model

The first model is model (6.55), an FAR model. The model is initialized with $X_0 = X_1 = X_2 = X_3 = 2$, and the first 100 data points are warm-up period.

**Table 9:** Prediction error under an FAR model.

| | HRM | | | AAR(4) | | FAR(4,1) | | |
|---|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | 1-s | Ite | Dir |
| mean | 0.16 | 0.18 | 0.18 | 0.19 | 0.20 | 0.16 | 0.17 | 0.20 |
| median | 0.13 | 0.15 | 0.15 | 0.16 | 0.17 | 0.14 | 0.14 | 0.17 |
| std | 0.12 | 0.13 | 0.14 | 0.15 | 0.15 | 0.12 | 0.13 | 0.15 |
| MSPE | 0.04 | 0.05 | 0.05 | 0.06 | 0.06 | 0.04 | 0.05 | 0.06 |

| | AR(4) | | Loess(4) | | Locpoly(4) | |
|---|---|---|---|---|---|---|
| | 1-s | Dir | 1-s | Ite | 1-s | Ite |
| mean | 0.60 | 0.57 | 0.19 | 0.20 | 0.18 | 0.19 |
| median | 0.53 | 0.45 | 0.16 | 0.17 | 0.14 | 0.15 |
| std | 0.43 | 0.43 | 0.15 | 0.15 | 0.15 | 0.14 |
| MSPE | 0.54 | 0.51 | 0.06 | 0.06 | 0.05 | 0.06 |

Table 9 shows that there is no significant difference between our method (HRM) and the method specific for FAR model. "1-s" stands for one-step prediction; "Ite" stands for iterative two-step prediction; "Dir" stands for direct two-step prediction. Among nonlinear ones, FAR and our model gives the most accurate performance for this example. Linear AR model does not fit the nonlinear situation very well.

#### 5.6.3.2 A Threshold Autoregressive Model

The second model is TAR model,

$$X_t = \begin{cases} 0.62 + 1.25X_{t-1} - 0.43X_{t-2} + 0.3X_{t-3} - 0.2X_{t-4} + \varepsilon_t, \text{ if } X_{t-2} \leq 2.25, \\ 2.25 + 1.52X_{t-1} - 1.24X_{t-2} - 1.25X_{t-3} + 0.4X_{t-4} + \varepsilon_t, \text{ if } X_{t-2} > 2.25, \end{cases}$$

where $\{\varepsilon_t\}$ are i.i.d. from $N(0, 1.5^2)$. The model is initialized with $X_0 = X_1 = X_2 = X_3 = 0$, and the first 100 data points are warm-up period.

From Table 10, we can see that TAR outperforms all other methods. This is not surprising given the data generation mechanism. Because TAR may be considered as a special case of FAR, FAR(4,2) is applied for the example. Our method performs

**Table 10:** Prediction error for a TAR model.

| | HRM | | | AAR(4) | | TAR(4,4;2) | |
|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | 1-s | Ite |
| mean | 1.42 | 2.30 | 2.31 | 1.39 | 2.30 | 1.25 | 2.14 |
| median | 1.10 | 1.87 | 1.94 | 1.10 | 1.85 | 1.03 | 1.72 |
| std | 1.32 | 1.90 | 1.95 | 1.27 | 1.84 | 1.00 | 1.55 |
| MSPE | 3.75 | 8.89 | 9.15 | 3.55 | 8.63 | 2.59 | 6.98 |

| | FAR(4,2) | | | AR(4) | | Loess(4) | | Locpoly(4) | |
|---|---|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | 1-s | Ite | 1-s | Ite |
| mean | 1.45 | 2.43 | 2.48 | 3.76 | 6.08 | 2.16 | 4.45 | 1.69 | 2.49 |
| median | 1.15 | 1.81 | 1.82 | 2.62 | 4.14 | 1.79 | 3.57 | 1.30 | 2.00 |
| std | 1.23 | 2.01 | 2.38 | 4.07 | 6.79 | 1.72 | 3.36 | 1.53 | 2.07 |
| MSPE | 3.60 | 9.95 | 11.84 | 30.66 | 82.94 | 7.61 | 31.10 | 5.20 | 10.46 |

similarly to AAR and FAR. Loess and Locpoly have the worst performance among the nonlinear methods. Once again linear AR model does not fit the nonlinear situation well.

One reason for the underperformance of our method possibly is the discontinuity of the underlying model on boundaries. Recall our method assumes that the underlying function $f$ is differentiable, and we penalize on its Hessian. It will be interesting to further study the dependence of our method on the regularity of the underlying model.

#### *5.6.3.3 Other More Generic Models*

The above two examples show that even when the data are generated from a perfect model, e.g. AAR, TAR, or FAR, our method still produce comparable prediction results with the original generating model.

The third model is

$$\begin{aligned} X_t &= -X_{t-4}e^{-2X_{t-3}^2} + \frac{1}{1+4X_{t-2}^2}\cos(1.5X_{t-1})X_{t-1} \\ &\quad + \frac{X_{t-3}}{1+4X_{t-1}^2} + \frac{e^{1.5(X_{t-4}-1)}}{1+e^{1.5(X_{t-4}-1)}} + \varepsilon_t, \end{aligned} \tag{6.56}$$

where $\{\varepsilon_t\}$ are i.i.d. from $N(0, 0.5^2)$. The model is initialized with $X_j \sim N(0, 0.2^2)$, where $j = 0, \cdots, 3$, and the first 100 data points are warm-up period. The fourth

model is

$$\begin{aligned} X_t &= [0.2 + (0.3 + X_{t-3})e^{-4X_{t-4}^2}]X_{t-1} \\ &\quad +[-0.4 - (0.7 + 1.3X_{t-3})e^{-4X_{t-4}^2}]X_{t-2} + \varepsilon_t, \end{aligned} \tag{6.57}$$

where $\{\varepsilon_t\}$ are i.i.d. from $N(0, 0.5^2)$. The model is initialized with $X_j \sim N(0, 1)$, where $j = 0, \cdots, 3$, and the first 100 data points are warm-up period.

**Table 11:** Prediction error under two nonlinear models.

(a) Under model (6.56)

| | HRM | | | AAR(4) | | AR(4) | | Loess(4) | | Locpoly(4) | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | 1-s | Dir | 1-s | Ite | 1-s | Ite |
| mean | 0.47 | 0.56 | 0.56 | 0.52 | 0.64 | 0.95 | 0.89 | 0.58 | 0.62 | 0.46 | 0.59 |
| median | 0.39 | 0.41 | 0.43 | 0.44 | 0.53 | 0.82 | 0.76 | 0.47 | 0.50 | 0.38 | 0.44 |
| std | 0.35 | 0.55 | 0.52 | 0.40 | 0.58 | 0.73 | 0.69 | 0.45 | 0.56 | 0.35 | 0.56 |
| MSPE | 0.34 | 0.62 | 0.58 | 0.43 | 0.74 | 1.43 | 1.26 | 0.54 | 0.70 | 0.33 | 0.66 |

| | FAR(4,1) | | | FAR(4,2) | | | FAR(4,3) | | | FAR(4,4) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | Dir | 1-s | Ite | Dir | 1-s | Ite | Dir |
| mean | 0.51 | 1.00 | 0.62 | 0.59 | 0.97 | 0.63 | 0.56 | 1.03 | 0.58 | 0.57 | 1.00 | 0.59 |
| median | 0.41 | 0.82 | 0.48 | 0.49 | 0.81 | 0.51 | 0.44 | 0.92 | 0.46 | 0.45 | 0.81 | 0.48 |
| std | 0.39 | 0.78 | 0.63 | 0.47 | 0.77 | 0.53 | 0.55 | 0.77 | 0.50 | 0.45 | 0.75 | 0.55 |
| MSPE | 0.42 | 1.61 | 0.79 | 0.57 | 1.54 | 0.67 | 0.62 | 1.66 | 0.58 | 0.53 | 1.56 | 0.65 |

(b) Under model (6.57)

| | HRM | | | AAR(4) | | AR(4) | | Loess(4) | | Locpoly(4) | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | 1-s | Dir | 1-s | Ite | 1-s | Ite |
| mean | 0.52 | 0.49 | 0.52 | 0.65 | 0.53 | 0.82 | 0.71 | 1.53 | 1.12 | 0.52 | 0.48 |
| median | 0.40 | 0.37 | 0.36 | 0.48 | 0.39 | 0.57 | 0.52 | 0.61 | 0.54 | 0.40 | 0.38 |
| std | 0.81 | 0.50 | 0.74 | 1.08 | 0.52 | 1.08 | 0.69 | 5.71 | 2.71 | 0.90 | 0.45 |
| MSPE | 0.92 | 0.48 | 0.82 | 1.59 | 0.54 | 1.85 | 0.97 | 34.85 | 8.54 | 1.09 | 0.43 |

| | FAR(4,1) | | | FAR(4,2) | | | FAR(4,3) | | | FAR(4,4) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1-s | Ite | Dir | 1-s | Ite | Dir | 1-s | Ite | Dir | 1-s | Ite | Dir |
| mean | 0.64 | 0.61 | 0.62 | 0.80 | 0.56 | 0.52 | 0.61 | 0.54 | 0.60 | 0.67 | 0.58 | 0.56 |
| median | 0.47 | 0.43 | 0.42 | 0.46 | 0.39 | 0.40 | 0.44 | 0.41 | 0.42 | 0.48 | 0.42 | 0.38 |
| std | 1.00 | 0.94 | 0.98 | 3.33 | 0.80 | 0.55 | 0.89 | 0.60 | 0.65 | 1.10 | 0.60 | 0.63 |
| MSPE | 1.42 | 1.25 | 1.35 | 11.69 | 0.94 | 0.57 | 1.16 | 0.65 | 0.77 | 1.65 | 0.70 | 0.71 |

The above two examples are nonlinear and can not be included by AAR, FAR, or TAR. Table 11 demonstrates that our method outperforms all these three methods

in both one-step and iterative two-step predictions.

All of Loess, Locpoly and our method are nonlinear models, and do not require any specific model structure. Although all these three methods are local approaches, the great advantage of our method is that the penalty term of hessian functional can also take into account of the global properties of the data. For the above four examples, Loess always has the worst performance among the three. Occasionally, our method and Locpoly have the comparable performance, e.g., for the one-step prediction in the third example, but our method often performs better than Locpoly, e.g., in the first two examples.

The third and fourth example illustrate the flexibility of our method, as our method does not enforce any specific structure on the model. For some cases when the data are not generated from a perfect model, e.g. AAR, TAR, or FAR, our method can generate more accurate prediction results than the other models.

### 5.6.4 Real Datasets

We apply our method to some well-studied datasets. Comparison with other reported works is carried out. In many cases, we outperform models that are used by other researchers.

#### *5.6.4.1 Sunspot Data*

Sunspot data is well studied in the literature ([41, 14] and many more). Table 8.5 in [41] summarizes results for several previous models, including:

- FAR-1, a functional coefficient autoregressive model fitted via local polynomial methods, as specified by equation (8.19) together with Figure 8.5 in [41];
- FAR-2, a functional coefficient autoregressive model fitted by Chen and Tsay (1993) [14], with exact formula given in (8.18) in [41];
- TAR, a threshold autoregressive model that is specified in (8.20) in [41].

In all the above models, the number of lags is $p = 8$. In our model, we choose $p = 6$. (We choose $p = 6$ because the GCV and prediction errors seem to be stabilized at this point.) The number of the nearest neighbors $k = 29$ is chosen by minimizing the aforementioned generalized cross validation function GCV$(\cdot)$ as a function of both $\lambda$ and $k$.

Table 12 presents prediction errors of our method (column HRM) together with errors of the above three methods (copied from [41, Table 8.5]). Our method generates the smallest one-step average absolute prediction error. Our average absolute prediction error of two-step prediction is slightly worse than FAR-1. However, we still outperform FAR-2 and TAR. Note the above is achieved by using six (instead of eight) predictors in our method.

**Table 12:** Prediction Errors for Sunspot Data.

| | HRM | | FAR-1 | | FAR-2 | | TAR | |
|---|---|---|---|---|---|---|---|---|
| Year | 1-s | Ite | 1-s | Ite | 1-s | Ite | 1-s | Ite |
| 1980 | 1.5 | 1.5 | 1.4 | 1.4 | 13.8 | 13.8 | 5.5 | 5.5 |
| 1981 | 7.8 | 9.1 | 11.4 | 10.4 | 0.0 | 3.8 | 1.3 | 0.0 |
| 1982 | 9.6 | 14.1 | 15.7 | 20.7 | 10.0 | 16.4 | 19.5 | 22.1 |
| 1983 | 8.1 | 4.0 | 10.3 | 0.7 | 3.3 | 0.8 | 4.8 | 6.5 |
| 1984 | 3.3 | 1.0 | 1.0 | 1.5 | 3.8 | 5.6 | 14.8 | 15.9 |
| 1985 | 10.3 | 8.1 | 2.6 | 3.4 | 4.6 | 1.7 | 0.2 | 2.7 |
| 1986 | 0.4 | 7.3 | 3.1 | 0.7 | 1.3 | 2.5 | 5.5 | 5.4 |
| 1987 | 9.5 | 9.1 | 12.3 | 13.1 | 21.7 | 23.6 | 0.7 | 17.5 |
| AAPE | 6.3 | 6.8 | 7.2 | 6.5 | 7.3 | 8.3 | 6.6 | 9.5 |

#### *5.6.4.2 Blowfly Data*

We apply our method to the blowfly data. It is known that the first 206 data points are nonlinear, and the remaining data points are almost linear [108]. Thus we use the first 195 data points as training data, then make postsample prediction for data point 196 to 210. The results are reported in Table 13.

Other four models are compared. A threshold autoregressive model TAR$(1, 3; 8)$

is suggested in [102, page 337], We apply the model with the same order but refit the model (column TAR), because the original model is applied to a different segment of the time series. To verify the order of the TAR model for our training data, We automatically select the best order of the TAR model with respect to the pooled AIC criteria, using the "selectSETAR" procedure in R package "tsDyn". By fixing the threshold variable as the 8th lag, TAR$(2, 3; 8)$ is selected. As the second variable in the lower regime is not significant, it means that TAR$(1, 3; 8)$ is almost same as TAR$(2, 3; 8)$.

The second and third models are functional coefficient autoregressive (FAR) models with different number of dependent variables:

- FAR–1: $X_t = f_0(X_{t-8}) + f_1(X_{t-8})X_{t-1} + f_2(X_{t-8})X_{t-2} + f_3(X_{t-8})X_{t-3} + \varepsilon_t$.
- FAR–2: $X_t = f_0(X_{t-8}) + f_1(X_{t-8})X_{t-1} + f_2(X_{t-8})X_{t-2} + f_3(X_{t-8})X_{t-3} + f_4(X_{t-8})X_{t-4} + \varepsilon_t$.

A similar model with two dependent variables is given in [112]. Note that the above FAR–1 model corresponds to the TAR model given in [102]. Moreover, we observe that more dependent variables can dramatically reduce the prediction errors of the FAR model for our training data. The fourth one is a standard autoregressive model with 8 lags.

For comparison, applying our algorithm, we fit the following generic model,

$$X_t = f(X_{t-1}, X_{t-2}, X_{t-3}, X_{t-4}, X_{t-8}) + \varepsilon.$$

The number of the nearest neighbors $k = 21$ is chosen by minimizing GCV function.

From Table 13, it is observed that when our model is applied, both the average one-step prediction error and the average two-step prediction error are better than those of four competing methods. This example once again demonstrates the powerfulness of our algorithm in time series prediction.

**Table 13:** Prediction Errors for Blowfly Data.

| | HRM | | TAR(1,3;8) | | FAR–1 | | FAR–2 | | AR(8) | |
|---|---|---|---|---|---|---|---|---|---|---|
| obs. | 1-s | Ite | 1-s | Ite | 1-s | Ite | 1-s | Ite | 1-s | Ite |
| 196 | 0.040 | 0.196 | 0.048 | 0.175 | 0.112 | 0.236 | 0.089 | 0.227 | 0.003 | 0.266 |
| 197 | 0.052 | 0.008 | 0.035 | 0.031 | 0.012 | 0.174 | 0.009 | 0.139 | 0.075 | 0.144 |
| 198 | 0.004 | 0.076 | 0.014 | 0.033 | 0.062 | 0.081 | 0.057 | 0.071 | 0.087 | 0.078 |
| 199 | 0.010 | 0.003 | 0.015 | 0.035 | 0.029 | 0.125 | 0.023 | 0.113 | 0.031 | 0.117 |
| 200 | 0.078 | 0.061 | 0.092 | 0.113 | 0.097 | 0.142 | 0.090 | 0.126 | 0.066 | 0.033 |
| 201 | 0.136 | 0.261 | 0.163 | 0.290 | 0.129 | 0.266 | 0.138 | 0.257 | 0.271 | 0.241 |
| 202 | 0.049 | 0.240 | 0.063 | 0.287 | 0.060 | 0.215 | 0.071 | 0.232 | 0.322 | 0.410 |
| 203 | 0.000 | 0.040 | 0.081 | 0.004 | 0.045 | 0.024 | 0.047 | 0.033 | 0.121 | 0.442 |
| 204 | 0.108 | 0.107 | 0.071 | 0.040 | 0.082 | 0.032 | 0.063 | 0.011 | 0.051 | 0.416 |
| 205 | 0.046 | 0.158 | 0.036 | 0.134 | 0.031 | 0.114 | 0.012 | 0.077 | 0.088 | 0.199 |
| 206 | 0.180 | 0.130 | 0.175 | 0.124 | 0.177 | 0.146 | 0.159 | 0.147 | 0.208 | 0.167 |
| 207 | 0.234 | 0.016 | 0.298 | 0.059 | 0.186 | 0.018 | 0.192 | 0.038 | 0.040 | 0.039 |
| 208 | 0.007 | 0.286 | 0.033 | 0.441 | 0.028 | 0.176 | 0.019 | 0.190 | 0.210 | 0.078 |
| 209 | 0.081 | 0.090 | 0.107 | 0.061 | 0.099 | 0.139 | 0.055 | 0.081 | 0.167 | 0.174 |
| 210 | 0.111 | 0.210 | 0.154 | 0.301 | 0.134 | 0.290 | 0.148 | 0.235 | 0.350 | 0.114 |
| Ave. | 0.076 | 0.126 | 0.092 | 0.142 | 0.086 | 0.145 | 0.078 | 0.132 | 0.139 | 0.195 |

## *5.7 Conclusion*

We introduce a hessian regularized nonlinear time-series model for prediction in time series. The approach is especially powerful when the number of dependent variables is greater than three, which can not be handled by natural cubic spline and thin plate spline. Moreover, our approach is nonlinear and nonparametric, and does not enforce any specific structure on the model. Both the theoretical and simulation results provide a strong verification and support of our model.

## *5.8 Appendix: Derivation for Generalized Cross Validation*

The derivation of the GCV function is as follows. We first establish the leave-one-out theorem. Then we demonstrate that (4.49) gives a reasonable approximation to the leave-one-out cross validation error.

**Theorem 5.8.1** *Suppose that*

$$\hat{f}_\lambda = \arg\min_f \sum_{t=p+1}^{n} (X_t - f(\mathbf{Z}_{t-1}))^2 + \lambda \mathcal{H} f, \tag{8.58}$$

*and*

$$h_\lambda(k, \tilde{X}_k) = \arg\min_f \sum_{t \neq k} (X_t - f(\mathbf{Z}_{t-1}))^2 + (\tilde{X}_k - f(\mathbf{Z}_{k-1}))^2 + \lambda \mathcal{H} f,$$

*Denote $\hat{f}_\lambda^{[k]}$ as the optimal solution of $f$ after leaving $k$th observation out, i.e. $\hat{f}_\lambda^{[k]} = \arg\min_f \sum_{t \neq k} (X_t - f(\mathbf{Z}_{t-1}))^2 + \lambda \mathcal{H} f$, then we have*

$$\hat{f}_\lambda^{[k]} = h_\lambda(k, \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1})).$$

Proof. The following has been used in deriving leave-one-out theorem in other situations, such as for smoothing splines and for regression. We have

$$\begin{aligned} L_k(f) &\overset{\text{def.}}{=} \sum_{t \neq k} (X_t - f(\mathbf{Z}_{t-1}))^2 + (\hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1}) - f(\mathbf{Z}_{k-1}))^2 + \lambda \mathcal{H} f \\ &\geq \sum_{t \neq k} (X_t - f(\mathbf{Z}_{t-1}))^2 + \lambda \mathcal{H} f \\ &\geq \sum_{t \neq k} (X_t - \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1}))^2 + \lambda \mathcal{H} f \\ &\overset{\text{def.}}{=} L_k(\hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1})). \end{aligned}$$

□

Suppose the solution of problem (8.58) is a linear estimator, i.e.,

$$(\hat{f}_\lambda(\mathbf{Z}_p), \cdots, \hat{f}_\lambda(\mathbf{Z}_{n-1}))^T = \mathbf{A}(\lambda)(X_{p+1}, \cdots, X_n)^T.$$

Meanwhile, according to theorem 5.8.1, we have

$$(\hat{f}_\lambda^{[k]}(\mathbf{Z}_p), \cdots, \hat{f}_\lambda^{[k]}(\mathbf{Z}_{n-1}))^T = \mathbf{A}(\lambda)(X_{p+1}, \cdots, X_{k-1}, \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1}), X_{k+1}, \cdots, X_n)^T.$$

Thus, we have

$$\begin{aligned} X_k - \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1}) &= \frac{X_k - \hat{f}_\lambda(\mathbf{Z}_{k-1})}{\frac{X_k - \hat{f}_\lambda(\mathbf{Z}_{k-1})}{X_k - \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1})}} \\ &= \frac{X_k - \hat{f}_\lambda(\mathbf{Z}_{k-1})}{1 - \frac{\hat{f}_\lambda(\mathbf{Z}_{k-1}) - \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1})}{X_k - \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1})}} \\ &= \frac{X_k - \hat{f}_\lambda(\mathbf{Z}_{k-1})}{1 - a_{kk}(\lambda)}, \end{aligned}$$

where $a_{kk}(\lambda)$ is $(k-p, k-p)$th element of matrix $\mathbf{A}(\lambda)$. Thus, GCV function satisfies

$$\begin{aligned} \mathrm{GCV}(\lambda) &= \frac{1}{n-p} \sum_{k=p+1}^{n} (X_k - \hat{f}_\lambda^{[k]}(\mathbf{Z}_{k-1}))^2 \\ &= \frac{1}{n-p} \sum_{k=p+1}^{n} \left( \frac{X_k - \hat{f}_\lambda(\mathbf{Z}_{k-1})}{1 - a_{kk}(\lambda)} \right)^2. \end{aligned}$$

As the above function is not stable when $a_{kk}(\lambda) = 1$, we replace $a_{kk}(\lambda)$ with $\frac{1}{n-p} \sum_{k=p+1}^{n} a_{kk}(\lambda)$, i.e., $\frac{1}{n-p} tr(\mathbf{A}(\lambda))$. Therefore, the final GCV function is (4.49).

# REFERENCES


[1] Antoniadis, A. and Fan, J., "Regularization of wavelets approximations (with discussion)," *J. Amer. Statist. Assoc.*, vol. 96, pp. 939–967, September 2001.

[2] Audet, N., Heiskanen, P., Keppo, J., and Vehvilainen, I., *Modelling Prices in Competitive Electricity Markets*, ch. Modeling Electricity Forward Curve Dynamics in the Nordic Market. London: John Wiley & Sons, 2004.

[3] Belkin, M. and Niyogi, P., "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, pp. 1373–1396, June 2003.

[4] Bloomfield, P. and Steiger, W. L., *Least Absolute Deviations: Theory, Applications, and Algorithms*. Boston: Birkha*ü*ser, 1983.

[5] Bookstein, F. L., "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 567–585, June 1989.

[6] Borg, I. and Groenen, P., *Modern Multidimensional Scaling: Theory and Applications*. New York: Springer-Verlag, 1997.

[7] Brockwell, P. J., *Introduction to Time Series and Forecasting*. Springer, 2 ed., 2003.

[8] Cai, Z., Fan, J., and Yao, Q., "Functional-coeffcient regression models for nonlinear time series models," *J. Am. Statist. Ass.*, vol. 95, pp. 941–956, 2000.

[9] Candes, E. J., Romberg, J., and Tao, T., "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, pp. 489–509, Feb 2006.

[10] Carreira-Perpinan, M. A., "A review of dimension reduction techniques," Tech. Rep. Technical Report CS-96-09, Dept. of Computer Science, University of Sheffield, 1997.

[11] Chen, J. and Huo, X., "Sparse representations for multiple measurement vectors (MMV) in an over-complete dictionary," in *Proceedings of ICASSP 2005*, (Philadelphia), March 2005.

[12] Chen, J. and Huo, X., "Theoretical results on sparse representations of multiple-measurement vectors," *IEEE Trans. Signal Processing*, vol. 54, no. 12, pp. 4634–4643, 2006.

[13] Chen, J., Deng, S.-J., and Huo, X., "Electricity price curve modeling by manifold learning," tech. rep., Georgia Institute of Technology, 2007. also available at http://www2.isye.gatech.edu/statistics/papers/07-02.pdf.

[14] Chen, R. and Tsay, R. S., "Functional-coeffcient autoregressive models," *J. Am. Statist. Ass.*, vol. 88, pp. 298–308, 1993.

[15] Chen, S. S., Donoho, D. L., and Saunders, M. A., "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001. Selected from *SIAM J. Sci. Comput.*, vol 20, no. 1, 33–61, 1998.

[16] Cleveland, R. B., Cleveland, W. S., McRae, J., and Terpenning, I., "Stl: A seasonal-trend decomposition procedure based on loess," *Journal of Official Statistics*, vol. 6, pp. 3–73, 1990.

[17] Cleveland, W. S., "Robust locally weighted regression and smoothing scatterplots," *J. Amer. Statist. Assoc.*, vol. 74, pp. 829–836, 1979.

[18] Cleveland, W. S., "Locally weighted regression: an approach to regression analysis by local fitting," *J. Amer. Statist. Assoc.*, vol. 83, pp. 596–610, 1988.

[19] Conejo, A. J., Contreras, J., Espínola, R., and Plazas, M., "Forecasting electricity prices for a day-ahead pool-based electric energy market," *International Journalof Forecasting*, vol. 21, pp. 435–462, 2005.

[20] Conejo, A. J., Plazas, M. A., Espínola, R., and Molina, A. B., "Day-ahead electricity price forecasting using the wavelet transform and ARIMA models," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 1035–1042, 2005.

[21] Contreras, J., Espínola, R., Nogales, F. J., and Conejo, A. J., "ARIMA models to predict next-day electricity prices," *IEEE Transactions on Power Systems*, vol. 18, no. 3, pp. 1014–1020, 2003.

[22] Cotter, S. F., Rao, B. D., Engan, K., and Kreutz-Delgado, K., "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

[23] Couvreur, C. and Bresler, Y., "On the optimality of the backward greedy algorithm for the subset selection problem," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 3, pp. 797–808, 2000.

[24] Davis, G., Mallat, S., and Zhang, Z., "Adaptive time-frequency decompositions," *Optical Engineering.*, vol. 33, pp. 2183–2191, 1994.

[25] Davison, M., Anderson, L., Marcus, B., and Anderson, K., "Development of a hybrid model for electricity spot prices," *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 257–264, 2002.

[26] Deng, S. J., "Stochastic models of energy commodity prices and their applications: Mean-reversion with jumps and spikes," *UCEI POWER Working Paper P-073*, 2000.

[27] Deng, S. J. and Jiang, W. J., "Levy process driven mean-reverting electricity price model: a marginal distribution analysis," *Decision Support Systems*, vol. 40, no. 3-4, pp. 483–494, 2005.

[28] Donoho, D. L., "For most large underdetermined systems of equations, the minimal $l_1$-norm near-solution approximates the sparsest near-solution," *Comm. Pure Appl. Math.*, vol. 59, no. 7, pp. 907–934, 2006.

[29] Donoho, D. L., "For most large underdetermined systems of linear equations the minimal $l_1$-norm solution is also the sparsest solution," *Comm. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, 2006.

[30] Donoho, D. L. and Elad, M., "Optimally sparse representation in general (non-orthogonal) dictionaries via $\ell^1$ minimization," *Proc. Nat. Aca. Sci.*, vol. 100, pp. 2197–2202, 2002.

[31] Donoho, D. L., Elad, M., and Temlyakov, V., "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inform. Theory*, vol. 52, no. 1, pp. 6–18, 2006.

[32] Donoho, D. L. and Grimes, C., "Hessian eigenmaps: new locally linear embedding techniques for high-dimensional data," *Proceedings of the National Academy of Sciences*, vol. 100, pp. 5591–5596, 2003.

[33] Donoho, D. L. and Huo, X., "Uncertainty principles and ideal atomic decomposition," *IEEE Transactions on Information Theory*, vol. 47, pp. 2845–2862, November 2001.

[34] Donoho, D. L. and Stark, P., "Uncertainty principles and signal recovery," *SIAM J. Appl. Math.*, vol. 49, no. 3, pp. 906–931, 1989.

[35] Donoho, D. and Johnstone, I., "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.

[36] Elad, M. and Bruckstein, A. M., "A generalized uncertainty principle and sparse representation in pairs of bases," *IEEE Trans. Inform. Theory*, vol. 48, no. 9, pp. 2558–2567, 2002.

[37] Fan, J. and Gijbels, I., *Local polynomial modelling and its applications*, vol. 66 of *Monographs on statistics & applied probability*. New York, NY: Chapman & Hall, 1996.

[38] Fan, J. and Li, R., "Variable selection via nonconvave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, pp. 1348–1360, December 2001.

[39] Fan, J. and Li, R., "Statistical challenges with high dimensionality: Feature selection in knowledge discovery," *Proceedings of the International Congress of Mathematicians*, vol. 3, pp. 595–622, 2006.

[40] Fan, J. and Peng, H., "Nonconcave penalized likelihood with a diverging number of parameters," *Ann. Statist.*, vol. 32, pp. 928–961, June 2004.

[41] Fan, J. and Yao, Q., *Nonlinear Time Series: Nonparametric and Parametric Methods*. Springer, 2003.

[42] Fan, J., Yao, Q., and Cai, Z., "Adaptive varying-coefficient linear models," *J. R. Statist. Soc. B*, vol. 65, pp. 57–80, 2003.

[43] Feuer, A. and Nemirovski, A., "On sparse representation in pairs of bases," *IEEE Trans. Inform. Theory*, vol. 49, no. 6, pp. 1579–1581, 2003.

[44] Fodor, I. K., "A survey of dimension reduction techniques," Tech. Rep. UCRL-ID-148494, LLNL technical report, 2002. http://www.llnl.gov/CASC/sapphire/pubs.html.

[45] Frank, I. E. and Friedman, J. H., "A statistical view of some chemometrics regression tools," *Technometrics*, vol. 35, no. 2, pp. 109–148, 1993.

[46] Fuchs, J. J., "On sparse representations in arbitrary redundant bases," *IEEE Trans. Inform. Theory*, vol. 50, pp. 1341–1344, June 2004.

[47] Garey, M. R. and Johnson, D. S., *Computers and intractability. A guide to the theory of NP-completeness*. San Francisco, CA, USA: W. H. Freeman and Co., 1979.

[48] Gentle, J. E., "Least absolute values estimation: an introduction," *Comm. Statist. B*, vol. 6, pp. 313–328, 1977.

[49] Golub, G. H. and Loan, C. F. V., *Matrix Computation*. Baltimore: the Johns Hopkins University Press, 1996.

[50] Gonzalez, A. M., Roque, A. M. S., and Gonzalez, J. G., "Modeling and forecasting electricity prices with input/output hidden markov models," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 13–24, 2005.

[51] Gorodnitsky, I. F., George, J. S., and Rao, B. D., "Neuromagnetic source imaging with FOCUSS: a recursive weighted minimum norm algorithm," *Electroencephalography and Clinical Neurophysiology*, vol. 95, pp. 231–251, October 1995.

[52] Green, P. J. and Silverman, B. W., *Nonparametric regression and generalized linear models: a roughness penalty approach*, vol. 58 of *Monographs on statistics and applied probability*. New York, NY: Chapman & Hall, 1994.

[53] Gribonval, R. and Nielsen, M., "Sparse representations in unions of bases," *IEEE Trans. Inform. Theory*, vol. 49, no. 12, pp. 3320–3325, 2003.

[54] Harikumar, G. and Bresler, Y., "A new algorithm for computing sparse solutions to linear inverse problems," in *Proc. ICASSP*, vol. 3, (Atlanta, GA), pp. 1331–1134, May 1996.

[55] Harikumar, G., Couvreur, C., and Bresler, Y., "Fast optimal and suboptimal algorithms for sparse solutions to linear inverse problems," in *Proc. of ICASSP*, (Seattle, Washington), May 1998.

[56] Hastie, T., Tibshirani, R., and Friedman, J., *The elements of statistical learning*. Springer, 2001.

[57] Horn, R. A. and Johnson, C. R., *Matrix analysis*. Cambridge University Press, 1985.

[58] Hunter, D. R. and Li, R., "Variable selection using MM algorithms," *Ann. Statist.*, vol. 33, pp. 1617–1642, August 2005.

[59] Huo, X., "A geodesic distance and local smoothing based clustering algorithm to utilize embedded geometric structures in high dimensional noisy data," in *SIAM International Conference on Data Mining, Workshop on Clustering High Dimensional Data and its Applications*, (San Francisco, CA), May 2003.

[60] Huo, X. and Chen, J., "Local linear projection (LLP)," in *First IEEE Workshop on Genomic Signal Processing and Statistics (GENSIPS)*, (Raleigh, NC), October 2002. http://www.gensips.gatech.edu/proceedings/.

[61] Huo, X., Ni, X., and Smith, A. K., *Mining of Enterprise Data*, new york Ch. A survey of manifold-based learning methods, invited book chapter. Springer, 2005. to appear, also available at http://www2.isye.gatech.edu/statistics/papers/06-10.pdf.

[62] Huo, X. and Ni, X. S., "When do stepwise algorithms meet subset selection criteria?," *Ann. Statist.*, vol. 35, no. 2, pp. 870–887, 2007.

[63] Jiang, T., "The asymptotic distributions of the largest entries of sample correlation matrices," *Ann. Appl. Probab.*, vol. 14, no. 2, pp. 865–880, 2004.

[64] Jiang, T., "Maxima of entries of Haar distributed matrices," *Probab. Theory Related Fields*, vol. 131, no. 1, pp. 121–144, 2005.

[65] Johnson, B. and Barz, G., *Energy Modelling and the Management of Uncertainty*, ch. Selecting Stochastic Processes for Modeling Electricity Prices. Risk Books, 1999. London.

[66] Johnstone, I. M., *Function Estimation and Gaussian Sequence Models*, un published monograph ed., 2004. Available at www-stat.stanford.edu/∼imj/baseb.pdf.

[67] Knittel, C. and Roberts, M., "Empirical examination of deregulated electricity prices," *Energy Economics*, vol. 27, no. 5, pp. 791–817, 2005.

[68] Koenker, R. and Mizera, I., "Penalized triograms: total variation regularization for bivariate smoothing," *J. R. Stat. Soc. Ser. B Stat. Methodol*, vol. 66, no. 1, pp. 145–163, 2004.

[69] Kreutz-Delgado, K. and Rao, B. D., "Measures and algorithms for best basis selection," in *Proc. of ICASSP*, vol. 3, (Seattle, Washington), pp. 1881–1884, May 1998.

[70] Kruskal, J. B., "Multidimensal scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, pp. 1–27, 1964.

[71] Kruskal, J. B., "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra and Its Applications*, vol. 18, no. 2, pp. 95–138, 1977.

[72] Lange, K., *Numerical analysis for statisticians*. New York, NY: Springer, 1999.

[73] Leviatan, D. and Temlyakov, V. N., "Simultaneous approximation by greedy algorithms," tech. rep., IMI Report 2003:02, Univ. of South Carolina at Columbia, 2003.

[74] Leviatan, D. and Temlyakov, V. N., "Simultaneous greedy approximation in Banach spaces," tech. rep., IMI Report 2003:26, Univ. of South Carolina at Columbia, 2003.

[75] Levina, E. and Bickel, P. J., "Maximum likelihood estimation of intrinsic dimension," in *Advances in Neural Information Processing Systems 17 (NIPS2004)*, MIT Press, 2005.

[76] Lora, A. T., Santos, J. M. R., Expósito, A. G., Ramos, J. L. M., and Santos, J. C. R., "Electricity market price forecasting based on weighted nearest neighbors techniques," *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1294–1301, 2007.

[77] Lucia, J. J. and Schwartz, E. S., "Electricity prices and power derivatives: Evidence from the nordic power exchange," *Review of Derivatives Research*, vol. 5, no. 1, pp. 5–50, 2002.

[78] Lutoborski, A. and Temlyakov, V. N., "Vector greedy algorithms," *J. Complexity*, vol. 19, pp. 458–473, 2004.

[79] Malioutov, D. M., Cetin, M., and Willsky, A. S., "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 3010–3022, 2005.

[80] Mallat, S., *A wavelet tour of signal processing*. San Diego, CA: Academic Press, Inc., 1998.

[81] Mallat, S. and Zhang, Z., "Matching pursuit in a time-frequency dictionary," *IEEE Trans. Signal Proc.*, vol. 41, pp. 3397–3415, 1993.

[82] Misiorek, A., Trueck, S., and Weron, R., "Point and interval forecasting of spot electricity prices: Linear vs. non-linear time series models," *Studies in Nonlinear Dynamics and Econometrics*, vol. 10, no. 3, 2006. Article 2.

[83] Mount, T. D., Ning, Y., and Cai, X., "Predicting price spikes in electricity markets using a regime-switching model with time-varying parameters," *Energy Economics*, vol. 28, no. 1, pp. 62–80, 2006.

[84] Nadler, B., Lafon, S., Coifman, R. R., and Kevrekidis, I. G., "Diffusion maps, spectral clustering and reaction coordinates of dynamical systems," *Applied and Computational Harmonic Analysis: Special issue on Diffusion Maps and Wavelets*, vol. 21, pp. 113–127, July 2006.

[85] Natarajan, B. K., "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.

[86] Ni, X. S., *New results in detection, estimation, and model selection*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, 2006. http://etd.gatech.edu/.

[87] Nikolova, M., "Local strong homogeneity of a regularized estimator," *SIAM J. Appl. Math.*, vol. 61, no. 2, pp. 633–658, 2000.

[88] Nogales, F. J., Contreras, J., Conejo, A. J., and Espínola, R., "Forecast next-day electricity prices by time series models," *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 342–348, 2002.

[89] Pagan, A. and Ullah, A., *Nonparametric Econometrics*. Cambridge Universtiy Press, 1999.

[90] Pati, Y. C., Rezaiifar, R., and Krishnaprasad, P. S., "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conference on Signals, Systems and Computers* (Singh, A., ed.), (Los Alamitos, CA), IEEE Comput. Soc. Press, 1993.

[91] Ramsay, B. and Wang, A. J., "A neural network based estimator for electricity spot-pricing with particular reference to weekend and public holidays," *Neurocomputing*, no. 47–57, 1998.

[92] Rao, B., Engan, K., and Cotter, S., "Diversity measure minimization based method for computing sparse solutions to linear inverse problems with multiple measurement vectors," in *Proceedings of ICASSP*, (Montreal), May 2004.

[93] Rao, B. D. and Kreutz-Delgado, K., "An affine scaling methodology for best basis selection," *IEEE Trans. on Signal Processing*, vol. 47, pp. 187–200, January 1999.

[94] Reinsch, C. H., "Smoothing by spline functions," *Numerische Mathematik*, vol. 10, no. 3, pp. 177–183, 1967.

[95] Saul, L. K. and Roweis, S. T., "Think globally, fit locally: unsupervised learning of low dimensional manifolds," *Journal of Machine Learning Research*, vol. 4, pp. 119–155, 2003.

[96] Saul, L. K. and T.Roweis, S., "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[97] Sugar, C. and James, G., "Finding the number of clusters in a data set: An information theoretic approach," *Journal of the American Statistical Association*, vol. 98, no. 750–763, 2003.

[98] Szkuta, B. R., Sanabria, L. A., and Dillon, T. S., "Electricity price short-term forecasting using artificial neural networks," *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 851–857, 1999.

[99] Temlyakov, V. N., "A remark on simultaneous greedy approximation," *East J. Approx.*, vol. 10, pp. 17–25, 2004.

[100] Tenenbaum, J. B., de Silva, V., and Langford, J. C., "A global geometric framework for nonlinear dimensionality reduction," *Science,*, vol. 290, pp. 2319–2323, 2000.

[101] Tibshirani, R., "Regression shrinkage and selection via the Lasso," *J. Roy. Statist. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[102] Tong, H., *Nonlinear Time Series: A Dynamical System Approach*. Oxford University Press, 1990.

[103] Tropp, J. A., "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, pp. 2231–2242, October 2004.

[104] Tropp, J. A., "Algorithms for simultaneous sparse approximation. part ii: Convex relaxation," *Signal Processing*, vol. 86, pp. 589–602, April 2006.

[105] Tropp, J. A., "Just relax: Convex programming methods for identifying sparse signals," *IEEE Trans. Inform. Theory*, vol. 51, pp. 1030–1051, March 2006.

[106] Tropp, J. A., Gilbert, A. C., and Strauss, M. J., "Simultaneous sparse approximation via greedy pursuit," in *Proceedings of ICASSP 2005*, (Philadelphia), March 2005.

[107] Tropp, J. A., Gilbert, A. C., and Strauss, M. J., "Algorithms for simultaneous sparse approximation. part i: Greedy pursuit," *Signal Processing*, vol. 86, pp. 572–588, April 2006.

[108] Tsay, R. S., "Non-linear time series analysis of blowfly population," *J. Time Ser. Anal.*, vol. 9, pp. 247–264, 1988.

[109] van der Maaten, L. J. P., Postma, E. O., and van den Herik, H. J., "Dimensionality reduction: A comparative review," tech. rep., University of Maastricht, 2007. http://www.cs.unimaas.nl/l.vandermaaten/dr/DR_draft.pdf.

[110] Verveer, P. and Duin, R., "An evaluation of intrinsic dimensionality estimators," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 81–86, 1995.

[111] Wahba, G., *Spline Models for Observational Data (CBMS-NSF Regional Conference Series in Applied Mathematics)*. SIAM: Society for Industrial and Applied Mathematics, 1990.

[112] Xia, Y. and Li, W. K., "On the estimation and testing of functional-coefficient linear models," *Statistica Sinica*, vol. 9, pp. 735–757, 1999.

[113] Yatchew, A., *Semiparametric Regression for the Applied Econometrician*. Cambridge University, 2003.

[114] Zhang, L., Luh, P. B., and Kasiviswanathan, K., "Energy clearing price prediction and confidence interval estimation with cascaded neural networks," *IEEE Transactions on Power Systems*, vol. 18, no. 1, pp. 99–105, 2003.

[115] Zhang, Z. and Zha, H., "Principal manifolds and nonlinear dimension reduction via tangent space alignment," *SIAM Journal of Scientific Computing*, vol. 26, no. 1, pp. 313–338, 2004.

[116] Zhu, J., Rosset, S., Hastie, T., and Tibshirani, R., "1-norm support vector machines," in *Neural Information Processing Systems*, vol. 16, 2003.

[117] Zou, H., "The adaptive lasso and its oracle properties," *J. Amer. Statist. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.