The Report Committee for Jisong Yang
Certifies that this is the approved version of the following report:

Using Greedy Algorithm to Learn Graphical Model for Digit Recognition

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Pradeep Ravikumar

Tom Sager

Using Greedy Algorithm to Learn Graphical Model for Digit Recognition

by

Jisong Yang, B.E.

Report

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN STATISTICS

The University of Texas at Austin

December 2014

Acknowledgements

Special thanks to Dr. Pradeep Ravikumar, for his expert guidance and encouragement throughout the course of this report. His theoretical research is the foundation of this report. I would also like to thank Dr. Tom Sager, the reader of this report, for his patience and valuable comments.

Abstract

Using Greedy Algorithm to Learn Graphical Model for Digit Recognition

Jisong Yang, M.S.Stat

The University of Texas at Austin, 2014

Supervisor: Pradeep Ravikumar

Graphical model, the marriage between graph theory and probability theory, has been drawing increasing attention because of its many attractive features. In this paper, we consider the problem of learning the structure of graphical model based on observed data through a greedy forward-backward algorithm and with the use of learned model to classify the data into different categories. We establish the graphical model associated with a binary Ising Markov random field. And model selection is implemented by adding and deleting edges between nodes. Our experiments show that: compared with previous methods, the proposed algorithm has better performance in terms of correctness rate and model selection.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Graphical model is the marriage between graph theory and probability theory. Graphical model has many attractive features such as: inference and learning are allowed to be treated together, supervised and unsupervised learning are merged seamlessly, tolerance to the missing data, explicitly displaying conditional independence and clear interpretation of the graph. There are two kinds of graphical model: those based on directed graph, and those based on undirected graph.

One of the most important research of graphical model is learning the structure of the model based on observed data through the use of computer algorithms and with the use of learned model to take actions such as classifying the data into different categories. Taking the example of recognizing handwritten digits, illustrated in Figure 1.1. Each digit corresponds to a 28×28 pixel image and each pixel is gray range from [0, 255], so can be represented by a vector x comprising 784 real numbers. The goal is to build a machine that will take such a vector x as input and that will produce the identity of the digit 0, . . ., 9 as the output. This is a difficulty problem due to the wide variability of handwriting. It could be tackled using handcrafted rules or heuristics for distinguishing the digits based on the shapes of the strokes, but in practice such an approach leads to a proliferation of rules and of exceptions to the rules and so on, and invariably gives poor results[1].

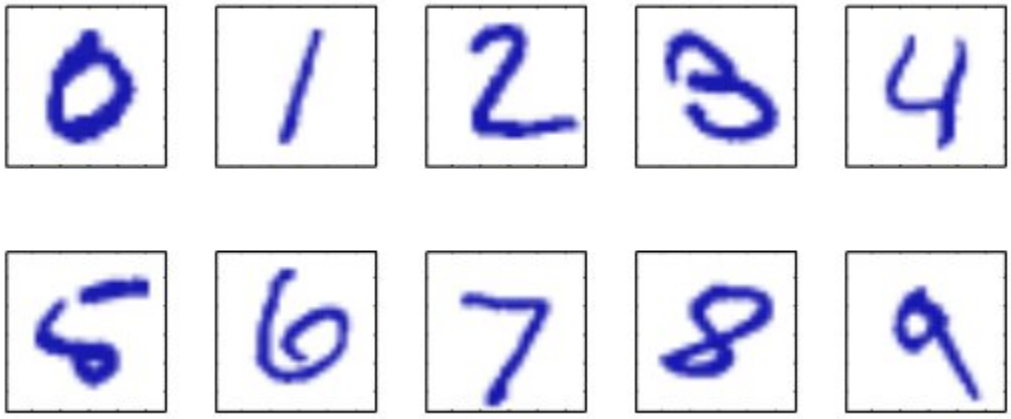


Figure 1.1 Handwritten digits

In order to get better results, the original input variables are typically preprocessed to transform them into some new space of variables. There are many methods to do this, such as principal component analysis (PCA) ([2]). The images of the digits are typically translated and scaled so that each digit is contained within a box of a fixed size. This greatly reduces the variability within each digit class, because the location and scale of all the digits are now the same, which makes it much easier for a subsequent pattern recognition algorithm to distinguish between the different classes. This preprocessing stage is called feature extraction. However, if the data is sparse, PCA does not appear to be applicable directly.

Based on Markov Random Field (MRF) modeling theory, Geman and Geman [3] made an analogy between images and statistical physical system and proposed to process pixel gray levels and image properties like molecules

or atoms in physical systems applying ideas and techniques used for the study of equilibrium states of chemicals processes at different temperatures. By the analogy, the posterior distribution defines another physical system which yielded the maximum a posterior (MAP) estimate of the image given the observation. However, Geman did not present a good way to extract the feature and the potential functions are very limited and specified forms, wheres in practice it is often desirable that the forms of the distributions should be determined or learner from the observed images.

Zhu and Mumford[4] proposed a new theory for building statistical models for images in a variety of applications. In their work, filtering theory, information feature functions and MRF come together in a general purpose learning approach. They suggested to obtain model balancing between model generality and model simplicity by two seemingly contrary criteria: (1) the maximum entropy principle, among all models, choosing the simplest model which maximizing the entropy over all distribution that reproduces the particular feature statistics. (2) the minimum entropy principle, among all sets of feature statistics, selecting the set has the minimum Kullback-Leibler divergence between feature sets and models given the image. The proposed feature selection scheme does not appear to be globe optimal by only using forward greed algorithm.

The main contribution of this paper may be described as following: we propose a forward and backward greed algorithm to learn the structure of graphical model and applied it to digits recognition, especially in binary scenario. Our experiments show that: compared with previous methods, the proposed algorithm has better performance in terms of correctness rate and model selection.

The remainder of this paper is organized as follow. We begin in chapter 2 by introducing previous methods and models, and then stating our model and algorithm. Simulation results and comparisons are shown in chapter 3. Finally, conclusion and discussion are made in chapter 4.

Chapter 2: Methodology

In a variety of disciplines such as computational vision, pattern recognition, image coding, a main goal is to establish a probability model characterizing a set of images. This is often posed as a statistical inference problem: suppose there is a joint distribution f(I) over the image space I, f(I) should concentrate on a subspace which corresponds to ensemble of images in the application and the objective is to estimate f(I) given a set of samples.

However, making inference about f(I) is more difficult than many of the learning problems in graphical model for the following reasons.

Firstly, the dimension of the image space is extremely large compared with the number of available training data. For instance, suppose the size of images is about 200*200 pixels, and each pixels is gray color for simplicity, the value is between [0, 255], thus the probability distribution is a function of 40,000 variables and each variable has 256 possible values, while the number of training data set is usually not sufficient to allow us to make direct inference.

Secondly, for the sparse graphical model, how to search nodes and find the edges between them is a big challenge in recent research. Figure 2.1 displays the picture of number “1”, the size is 28* 28 pixels. The background is black for the most of part, only the central of the picture is bright. Figure 2.2 shows the number “1” under matrix form that explains the concept more explicitly. Many elements of the matrix are zero corresponding to the dark background.

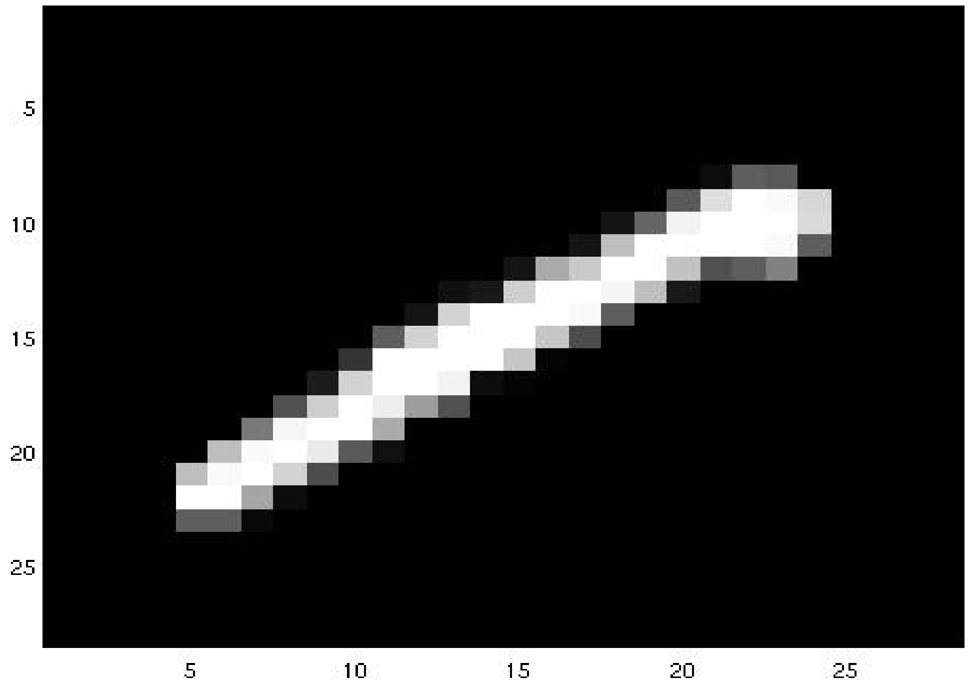


Figure 2.1 Image of number “1”

-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	20
0	0	0	0	0	0	0	0	0	20	168	203
0	0	0	0	0	0	0	16	27	206	253	253
0	0	0	0	0	0	23	209	253	254	253	248
0	0	0	0	0	93	210	253	253	254	196	76
0	0	0	0	54	254	254	254	254	198	7	0
0	0	0	29	209	253	253	240	13	7	0	0
0	0	80	207	253	238	159	81	0	0	0	0
0	123	247	253	253	170	0	0	0	0	0	0
191	248	253	235	88	17	0	0	0	0	0	0
250	253	208	77	0	0	0	0	0	0	0	0
253	167	13	0	0	0	0	0	0	0	0	0
93	10	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

Figure 2.2 Matrix form of number “1”

2.1 Previous Methods

Principle component analysis (PCA) is commonly used for identifying the images. First, eigenvectors with larger eigenvalues are extracted from the sample training data to retain the principal components of the images. Second, projecting both test data and train data onto the eigenspace, which is reduced dimension. Finally, using the features captured by projection to identify test data by comparing the distance. Figure2.3 shows the position of a ball attached to an oscillating spring is recorded using three cameras A, B and C. The position of the ball tracked by each camera is depicted in each panel below[2].

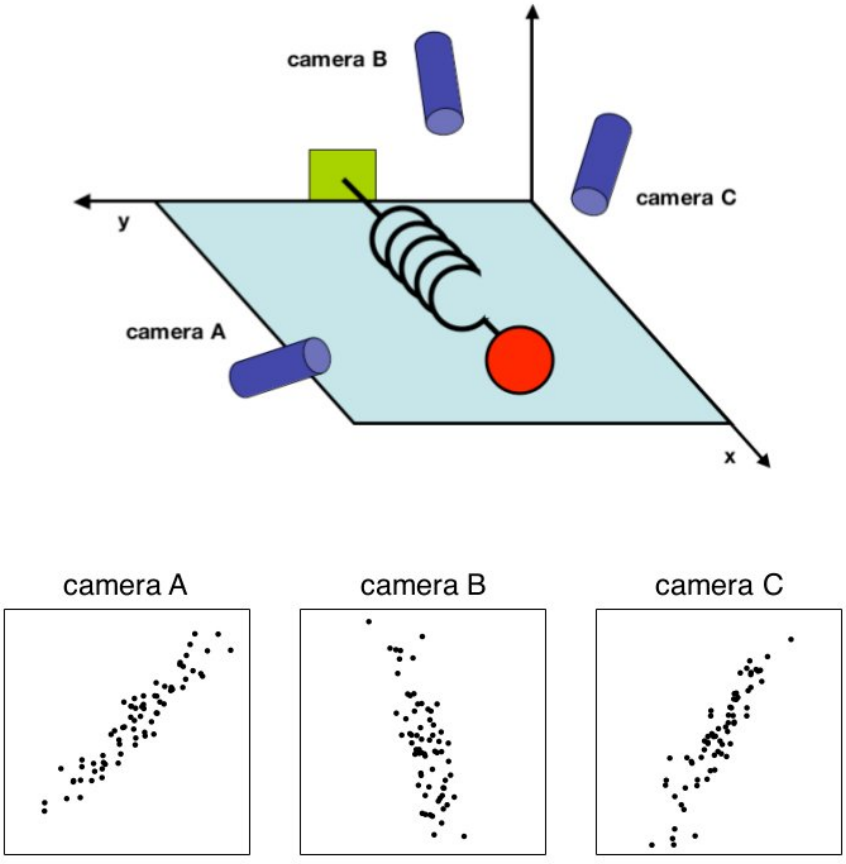


Figure 2.3 A toy example

In order to capture the principal components of the images, the eigenvectors derived from the samples of train data are sorted from the largest to the smallest according to their eigenvalues. A set of eigenvectors which corresponding to the larger eigenvalue are selected because they represent in which directions the data span dramatically. With those principal eigenvectors, less information is needed to reconstruct the test data and identification. Figure 2.4 show the largest 64 eigenvectors derived from training data.

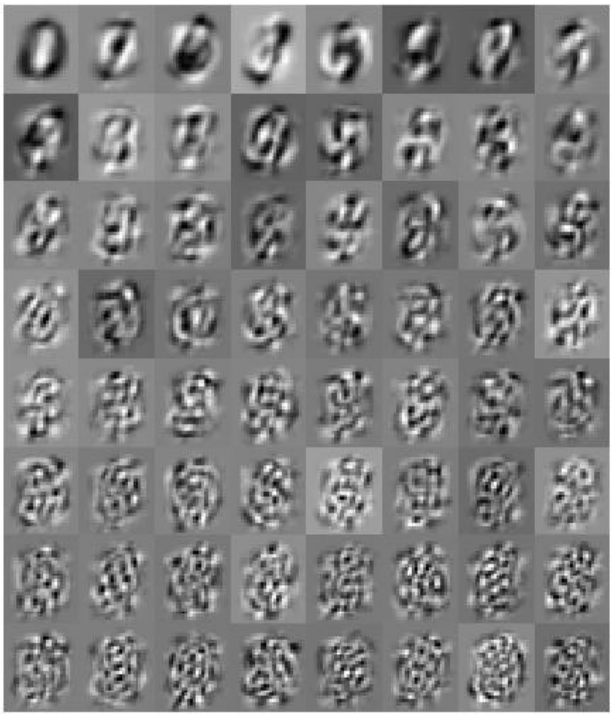


Figure 2.4 Top 64 Eigenvectors derived from training data

Technically speaking, PCA applies eigenvectors as features for pattern recognition and selects the eigenvectors by the descending order of eigenvalue. However, PCA is not suitable for processing the sparse images, for instance, binary images. First of all, since the matrix form of the sparse images has many zeros, the eigenvectors have little information to be good features, which is illustrated in table 2.1. Second, since the eigenvectors are sparse, it degrade the quality of the recovered pictures, the first row of figure 2.5 displays the original images from the raw images, the second row shows the reconstructed images. The recovered images have much noise since the image is binary.

1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
-6.7937e...	5.6305e-...	1.1889e-...	1.5184e-...	-3.8036e...	1.3873e-...	4.6917e-...	7.4574e-...	1.1290e-...	-9.0718e
-1.8569e...	1.5390e-...	3.2497e-...	4.1504e-...	-1.0397e...	3.7920e-...	1.2824e-...	2.0384e-...	3.0859e-...	-2.4796e
-8.8318e...	7.3197e-...	1.5456e-...	1.9740e-...	-4.9447e...	1.8035e-...	6.0992e-...	9.6947e-...	1.4677e-...	-1.1793e
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
-1.1664e...	1.6637e-...	1.2544e-...	2.3289e-...	-9.0441e...	6.1007e-...	8.5107e-...	-1.4343e...	-1.8674e...	-7.5077e
-4.0220e...	5.7369e-...	4.3254e-...	8.0308e-...	-3.1187e...	2.1037e-...	2.9347e-...	-4.9457e...	-6.4394e...	-2.5889e
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
-2.7198e...	1.2429e-...	4.0059e-...	4.8645e-...	2.6063e-...	-1.4535e...	1.5929e-...	-2.1716e...	-9.8100e...	3.3173e-
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Table 2.1 Part of eigenvector calculated from handwritten digits

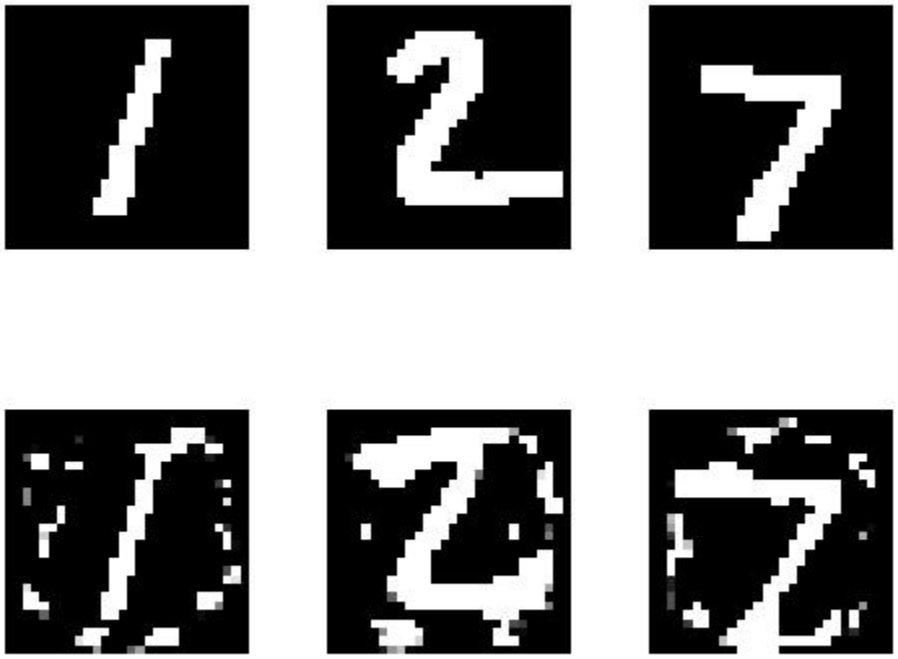


Figure 2.5 Original images and reconstructed images by PCA

A better solution to image process is thinking image as undirected graphical model also unknown as Markov random field (MRF). MRF is specified by an undirected graph G = (V, E) with vertex set V = {1, 2, . . , p} and edge set E belongs to V × V. The structure of this graph encodes certain conditional independence assumptions among subsets of the p-dimensional discrete random variable X = $(X_1, X_2 , \ldots , X_p)$ where variable $X_i$is associated with vertex i V . One important problem for such models is to estimate the underlying graph from n independent and identically distributed samples $\{x^{(1)}, x^{(2)}, \ldots , x^{(n)}\}$ drawn from the distribution specified by the Undirected graphical model. X is a MRF if P(X_s=x_s|X_r=x_r, r s) = P (X_s=x_s|X_r=x_r, $r \in N(s)$). Roughly speaking, in MRF is possible to evaluate the probability to

have a specific value in a state of the system having only knowledge of the relative neighbors set.

Based on Markov random field theory, Geman suggested to view pixel levels and presence and orientation of edges as states of atoms or molecules in grid physical system. The model can be described as following:

Given a set of sites S and a neighborhood system Geman modeling the image as Gibbs distribution such that:

(2.1) $$P(\omega) = \frac{1}{Z} exp(\frac{-U(\omega)}{T})$$

where Z is the normalizing or partition constant defined as

(2.2) $$Z = \sum_{\omega \in \Omega} exp(\frac{-U(\omega)}{T})$$

T is the constant relative to the temperature and U is the energy function of the firm

(2.3) $$U(\omega) = \sum_C V_c(\omega)$$

where the $V_c$functions are called potentials or potential functions and referred to specific cliques C. The potential functions are clique dependent, which means their value depends only on the values assumed by the states presents in the given clique C. The globe parameter T, is used to smoothly simulate an annealing process that converges to an equilibrium stages of the system. The general computational problem are:

I . sample from the distribution P(ω);

II. minimize U over Ω;

III. compute the expectation

One of the biggest problems working with the Gibbs distribution is the constant Z; it is often difficulty to calculate its value directly because it needs a great number of possible configurations. To obtain the quantitative information, Monte Carlo Markov Chain (MCMC) methods are good candidates to extract samples from complicated distribution.

Zhu and Mumford derived similar model based on entropy theory. The method is called minimax entropy principle which is composed by two key components:

I. The maximum entropy principle

The problem is formulated as the following constrained optimization problem,

(2.4) $P(I) = \arg\max\{-\int P(I)\log p(I)dI\}$,

Subject to $E_p\left(\emptyset^{(\partial)}(I)\right) = \int \emptyset^{(\partial)}P(I)\log p(I)dI \qquad \partial = 1, \dots K,$

and $\int P(I)dI = 1.$

where P(I) is the probability distribution with respect to image I. p(I) has the following Gibbs distribution form:

(2.5) $P(I) = \frac{1}{Z}\exp\{-\sum_{\partial=1}^{K} < \omega^{(\alpha)} \cdot \emptyset^{(\alpha)}(I) >\}$

where $\omega^{(\alpha)}$is a parameter vector has the same dimension as $\emptyset^{(\alpha)}(I)$, <·> denotes inner product and Z is the partition function for normalization.

II. The minimum entropy principle

Let B be the set of all possible features, and S ⊂B an arbitrary set of K features, entropy minimization provides a criterion for choosing the optimal set of features,

(2.6) $S^* = \arg\min entropy(P_S(I;\omega))$

where $P_S(I;\omega)$ denotes the fitted model using features in S.

Zhu and Mumford derived a forward greedy algorithm to find the set of features, which is not guaranteed to learn the graphical model with high probability.

Geman and zhu's work share the same model, Gibbs distribution, more generally, it belongs exponential family. Here is the question: can we do better with a more general model to learn the structure of model with less samples and still has the performance guarantee?

2.2 Proposed method

We derived a new algorithm based on the work [5] [6], we focus on the pairwise Markov random fields and Ising model. First, let's begin by introduce some background about them.

2.2.1 Pairwise Markov random fields

Let X = (X_1, X_2 , . . . , X_p) denote a random vector with each variable (X_s taking values in a corresponding set Xs. Suppose G = (V, E) is an undierectd graph consist of a set of vertices V = {1, . . . , p} and a set of unordered pairs E representing edges between the vertices, so that each random variable Xs is associated with a vertex s ∈ V . The pairwise Markov random field associated with the graph G over the random vector X is the family of distributions of X which factorize as P(x) ∝ exp{$\sum_{\partial(s,t)\in E=1} \emptyset_{st}(X_s, X_t)$} where for each edge (s, t) ∈ E, $\emptyset_{st}$is a mapping from pairs (xs, xt) ∈ Xs × Xt to the real line. The pairwise assumption providers no loss of generality for discrete random variables. With purely pairwise interactions, any Markov random field with higher order interactions can be converted (by introducing additional variables) to an equivalent pairwise Markov random field.

2.2.2 Ising model

The Ising model is a simple system originally created to study magnetism but is now used to examine a number of natural phenomena outside of physics. Figure 2.6 shows the atoms explained by Ising model. In this paper we assume random variable X_s∈ {-1, 1} for each vertex s ∈V, and $\emptyset_{st}(X_s, X_t)$= $\theta^* * (X_s X_t)$ for some parameter θ^*∈ R. And the distribution can be written as:

$$P(x) = \frac{1}{Z} \exp\{\sum_{(s,t)\in E} \theta^*(X_s X_t) + \sum_{(r)\in V} \theta^*(X_r)\} \quad (2.7)$$

where Z is the partition function as mentioned before.

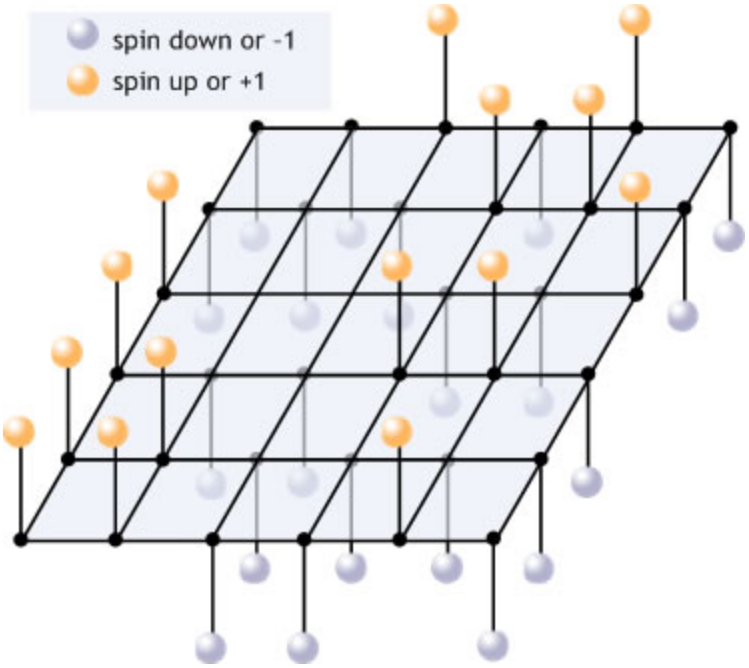


Figure 2.6 Lattice of each atom's magnetic moment

$$P_\theta^*(X_r|\ X_{v\setminus r}) = \frac{\mathrm{P}(X_r,\ X_{v\setminus r})}{\mathrm{P}(X_{v\setminus r})}$$

$$= \frac{exp(\sum \theta_{rt}\varphi(X_r,X_t)}{\sum_{X_r} exp(\sum \theta_{rt}\varphi(X_r,X_t)}$$

Since the $X_s \in \{-1, 1\}$, $\varphi(X_r, X_t) = X_r * X_t$

This conditional distribution can be written as:

(2.8) $$\mathbb{P}\left(X_r = x_r \middle| X_{V\setminus r} = x_{V\setminus r}\right) = \frac{\exp(\theta_r x_r + \sum_{t\in V\setminus r} \theta_{rt} x_r x_t)}{1 + \exp(\theta_r + \sum_{r\in V\setminus r} \theta_{rt} x_r)}.$$

Hence the variable $X_r$can be viewed as the response variable in a logistic regression in which all of the other variables$X_{v\backslash r}$play the role of the covariates.

With formula (2.8), based on computing a logistic regression of $X_r$on its neighbors$X_{v\backslash r}$, we may estimate the parameter vector $\theta^*$and the neighborhood structure. Suppose that we are given a collection of n samples $(X_1, X_2\ ,\ \ldots\ , X_n)$ where each one is p-dimensional vector and every element of the vector is $x^i \in \{-1, +1\}^p$that is i.i.d. drawn from a distribution of the form (2.7) for parameter vector $\theta^*$and graph G = (V , E) over the p variables. The parameter vector θ^*may be viewed as a $\binom{P}{2}$dimensional vector, indexed by pairs of distinct vertices but nonzero if and only if the vertex pair (s, t) belongs to the unknown edge set E of the underlying graph G. The value of each element of $\theta^*$is fixed to a certain valuev_θ. The task of graphical model selection is to infer the edge set E. more specifically, according the logistical regression of formula (2.8), θ^*can be estimated by minimization of loss function:

(2.9) $$\mathcal{L}(\Theta_r; D) = \frac{1}{n}\sum_{i=1}^{n}\left\{\log\left(1+\exp\left(\theta_r x^{(i)} + \sum_{t\in V\backslash r}\theta_{rt}x_r^{(i)}x_t^{(i)}\right)\right) - \theta_r x_r^{(i)} - \sum_{t\in V\backslash r}\theta_{rt}x_r^{(i)}x_t^{(i)}\right\}.$$

There are many classes of graphical model. Figure 2.7 shows four that are commonly used.

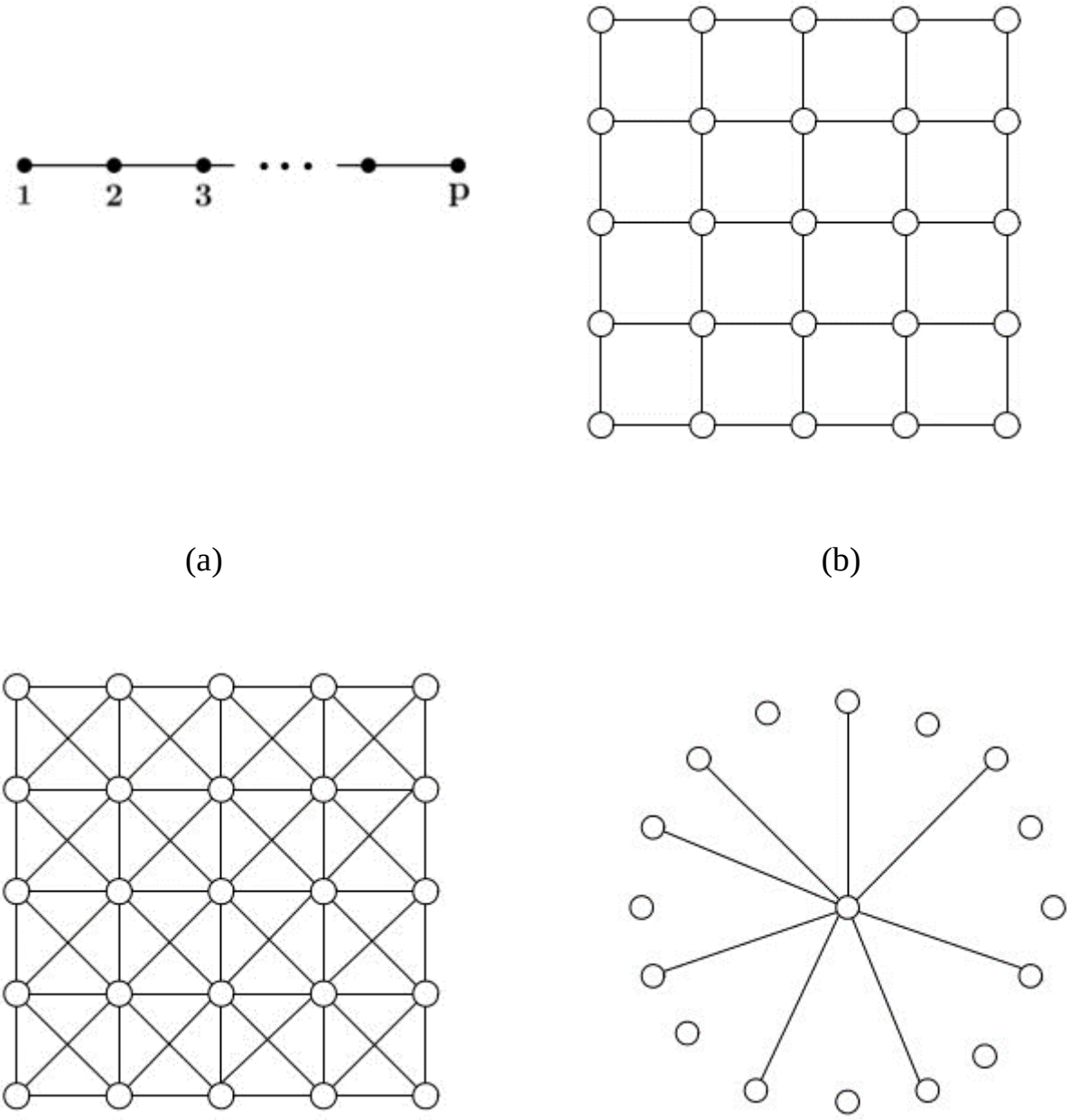


Figure 2.7 Four kinds of graph classes. (a) Chain graph. (b) Four-nearest neighbor grid. (c) Eight-nearest neighbor grid. (d) Star shaped graph.

In this paper, we choose four-nearest neighbor grid for graphical model learning. Based on the general loss function (2.9), we proposed the forward-backward greedy algorithm that rewrites the algorithm in [6]. First forward subroutine is executed to find the preliminary structure of the model, then backward subroutine is executed to delete some edges in the preliminary graph. Ultimately, the algorithm is end up with the true structure of the model.

Forward-backward greedy algorithm for finding a sparse graph model:

Input: Data D = $\{X_1, X_2, \ldots, X_n\}$ n samples, forward stopping threshold E_F, backforward stopping threshold E_B.

Output: estimated parameter θ

Initialization: $\theta^0 = 0,\ K = 1$

for each node in V

while true do {Forward step: adding edges}

$$(j_*, \alpha_*) \longleftarrow \arg \min_{j \in \left(\widehat{S}^{(k-1)}\right)^c ; \alpha} \mathcal{L}\left(\widehat{\theta}^{(k-1)} + \alpha e_j ; D\right)$$

$$\widehat{S}^{(k)} \longleftarrow \widehat{S}^{(k-1)} \cup \{j_*\}$$

$$\delta_f^{(k)} \longleftarrow \mathcal{L}\left(\widehat{\theta}^{(k-1)}; D\right) - \mathcal{L}\left(\widehat{\theta}^{(k-1)} + \alpha_* e_{j_*}; D\right)$$

if $\delta_f^{(k)} \leq \epsilon_S$ **then**

break

end if

$$\widehat{\theta}^{(k)} \longleftarrow \arg\min_{\theta} \mathcal{L}\left(\theta_{\widehat{S}^{(k)}}; D\right)$$

$$k \longleftarrow k + 1$$

while true do {backward step: deleting edges}

$$j^* \longleftarrow \arg \min_{j \in \widehat{S}^{(k-1)}} \mathcal{L}(\widehat{\theta}^{(k-1)} - \widehat{\theta}_j^{(k-1)} e_j; D)$$

if $\mathcal{L}\big(\widehat{\theta}^{(k-1)} - \widehat{\theta}_{j^*}^{(k-1)} e_{j^*}; D\big) - \mathcal{L}\big(\widehat{\theta}^{(k-1)}; D\big) > \nu \delta_f^{(k)}$ **then**

break

end if

$$\widehat{S}^{(k-1)} \longleftarrow \widehat{S}^{(k)} - \{j^*\}$$

$$\widehat{\theta}^{(k-1)} \longleftarrow \arg\min_{\theta} \mathcal{L}\big(\theta_{\widehat{S}^{(k-1)}}; D\big)$$

$$k \longleftarrow k - 1$$

end while

end while

end for

Here K is the cardinality of edges, will be discussed in the experiment. Since the loss function satisfy the restricted strong convexity and restricted strong smoothness (Negahban et al [7]), such that sparsistency is guaranteed. The optimal structure of graph will be found with high probability.

Chapter 3: Experiment

The data used in this paper is from Kaggle: *www.kaggle.com.* Totally, there are 70000 images. 42,000 training images and 28,000 images. Each image is 28 pixels in height and 28 pixels in width, for a total of 784 pixels in total. Each pixel has a single pixel-value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255, inclusively.

In order to apply Ising model, the raw data is convert to binary data first, then the training data is categorized into 10 group according to their labels which are “0” ,“1”, “2”, “3”.... “9”. For each group, structure of model is learned by the forward-backward greedy algorithm.

Another issue is about the cardinality of edges should be select in the model. To get the optimal number K of edge, a ten-fold cross validation is carried out. Figure 3.1 shows how error changes with number of K edges. The error goes down when number of edges increase, and then error goes up. The optimal K is between 30 and 40.

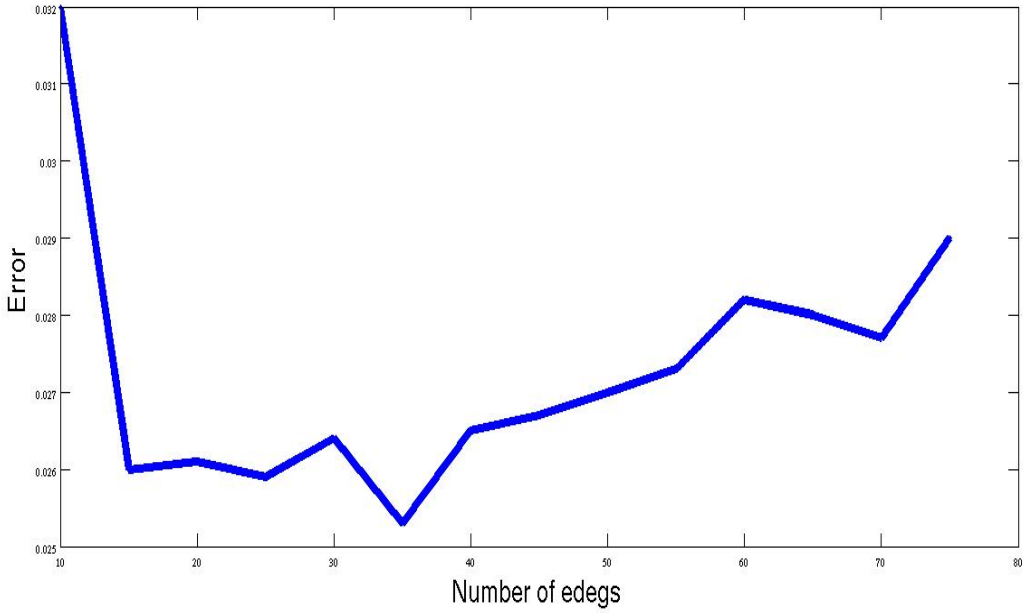


Figure 3.1 Error vs Number of edges K

In [5], it is proved that the consistent neighborhood selection can be obtained for sample size $n = \Omega(d^2 \log p)$, where d is the maximal degree of the node, p

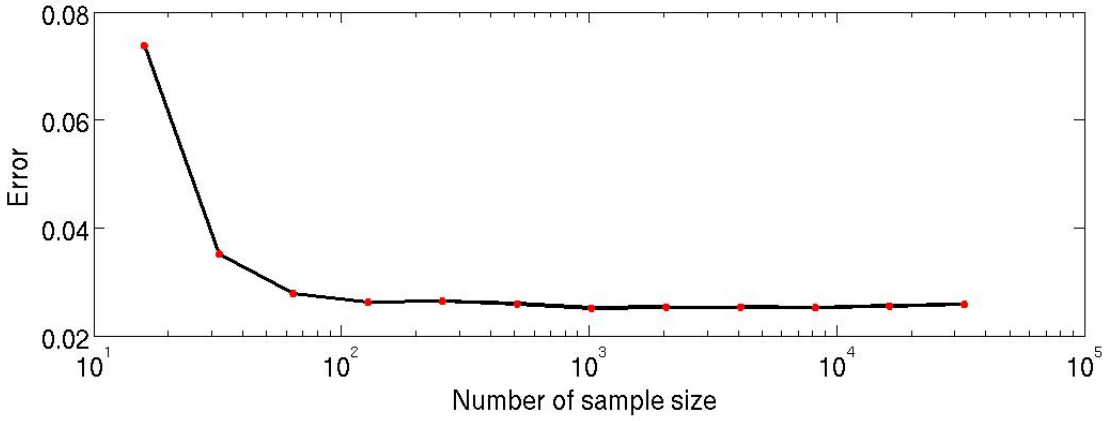


Figure 3.2 Error vs Number of sample N

is the number of node . In this paper, we use four-nearest neighbor grid, d is fixed to 4. Figure 3.2 shows that error is decreasing as number of size increase, and then the error stay parallel with the horizontal axis, which is consistent to the [5].

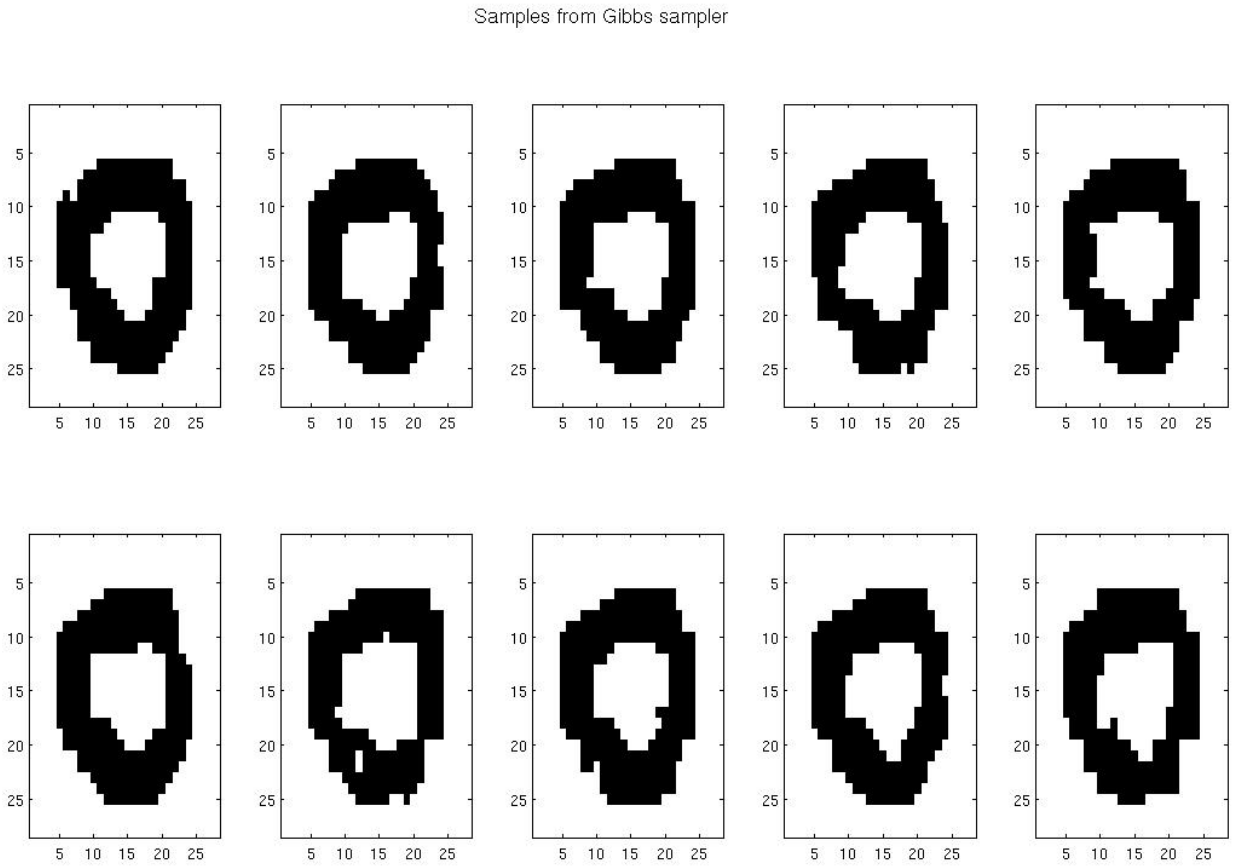
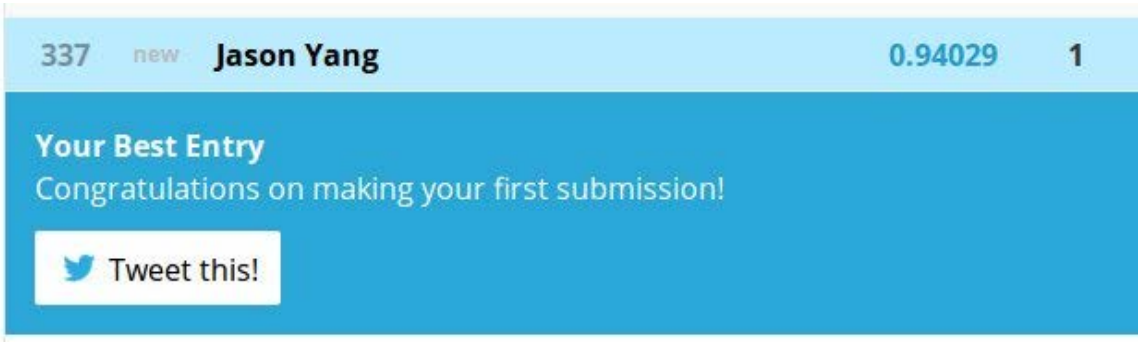


Figure3.3 Gibbs sampling from the learned model

Figure 3.3 shows ten samples of number “0” generated by the learned model. Samples are taken after 1000 iterations to make sure the distribution is convergent. The learned probability distribution fit the date quite well.

Finally, we compare our scheme with PCA by submitting the predicting result to Kaggle. Figure 3.4 shows PCA reached score 0.94029, which means around 94% of prediction is correct, while proposed scheme is 0.978, better than PCA.



(a)

(b)

Figure 3.4 Kaggle's score (a) PCA. (b) forward-backward greedy.

Chapter 4 Discussion

This report presents an algorithm to learn the structure of undirected model in sparse scenario. In order to perform consistent model selection in binary Ising graphical model, a forward- backward greedy selection scheme is applied to find the edges between the random variables. Our results show the learned probability distribution matches the data quiet well and can be used as pattern recognition with decent accuracy.

It would be interesting to extend current work from binary Markov random fields to general discrete graphical models with more than two number states, for instance, the RGB graph. Moreover, how to learn the underlying graph with mixed class grid and make selections adaptively should be also interesting.

References

[1] Christopher M. Bishop Pattern, (2006) Recognition and Machine Learning, ISBN:0387310738

[2] Jonathon Shlens, (2005) A Tutorial on Principal Component Analysis

[3] G EMAN , S. and G EMAN , D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans. PAMI 6 721–741.

[4] S.ch. Zhu, D. Mumford (1997), Minimax entropy principle and its application to texture modeling, Neural computation Vol.9, No.8

[5] P. Ravikumar, M. J. Wainwright, and J. Lafferty (2010). High-dimensional ising model selection using ℓ1- regularized logistic regression. Annals of Statistics, 38(3):1287–1319.

[6]A.Jalali, C.Johnson, P. Ravikumar (2011), On Learing Discrete Graphical models Using Greedy Methods Advances. In Neural Information Processing Systems (NIPS)

[7] S. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu (2009). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. In Neural Information Processing Systems (NIPS)