Constrained Least Squares Regression for Semi-Supervised Learning

Bo Liu[1,2], Liping Jing[1,⋆], Jian Yu[1], and Jia Li[1]

[1] Beijing Key Lab of Traffic Data Analysis and Mining,
Beijing Jiaotong University, Beijing, 100044, China
[2] College of Information Science and Technology,
Agricultural University of Hebei, Hebei, 071000, China
liubohbu@126.com, {lpjing,jianyu}@bjtu.edu.cn, jiali.gm@gmail.com

Abstract. The core tasks of graph based semi-supervised learning (GSSL) are constructing a proper graph and selecting suitable supervisory information. The ideal graph is able to outline the intrinsic data structure, and the ideal supervisory information could represent the whole data. In this paper, we propose a new graph learning method, called constrained least squares regression (CLSR), which integrates the supervisory information into graph learning process. To learn a more adaptive graph, regression coefficients and neighbor relations are combined in CLSR to capture the global and local data structures respectively. Moreover, as byproduct of CLSR, a new strategy is presented to select the high-quality data points as labeled samples, which is practical in real applications. Experimental results on different real world datasets demonstrate the effectiveness of CLSR and the sample selection strategy.

Keywords: graph based semi-supervised learning, graph construction, constrained least squares regression, labeled sample selection.

1 Introduction

Lack of sufficiently labeled data is a big problem when building supervised learner in real applications. Semi-supervised learning (SSL) can bridge the gap between labeled and unlabeled data, as it combines limited labeled samples with rich unlabeled samples to enhance the learner's ability [20]. As an important branch of SSL, graph based semi-supervised learning (GSSL) propagates the supervisory information (class labels) on a pre-defined graph and aims to make the similar samples share the common labels [12]. Under the cluster assumption [4] or the manifold assumption [8], there are many GSSL methods have been proposed, including Gaussian fields and harmonic functions (GFHF) [21], local and global consistency (LGC) [19], manifold regularization [2], and etc. For GSSL, there are several key issues to be solved including graph construction, labeled sample selection, learning model formulation, parameter adjustment and etc. In this paper, we will limit to highlight the former two issues.

⋆ Corresponding Author.

V.S. Tseng et al. (Eds.): PAKDD 2014, Part II, LNAI 8444, pp. 110–121, 2014.

An adaptive graph construction is a main challenge of GSSL. Neighborhood-driven methods (e.g., k-nearest-neighbors (k-NN) [16], ϵ-ball neighborhood [1] and b-matching graph [7]) are unable to reflect the overall views of data and sensitive to noise. Recently, some researchers formulate the graph building process into a subspace learning problem. Under the subspace assumption, each sample can be represented as a linear combination of other samples, and intuitively, the representation coefficients could be accepted as a proper surrogate of similarity metric. In the literature, this measurement is referred to as self-expressive similarity [6]. There are several methods such as sparse representation (SR) [6], low-rank representation (LRR) [10], least squares regression (LSR) [13] to obtain the representation coefficients.

Although these approaches have gained great effects in some domains, there are still some drawbacks. First, labeled samples only work at propagating stage, so the supervisory information cannot directly influence the affinity learning process. Second, regardless of noise and outliers, data points may not strictly lie in a union of subspaces, which indicates that the graph's adaptability is restricted owing to the utilization of a single metric. Third, in the context of the subspace assumption, when we have to select some samples as a labeled set, however, the existing method, random sampling, does not leverage the structural characteristic of the original dataset.

Inspired by the work [13], we propose an effective graph construction framework, called constrained least squares regression (CLSR), and try to improve GSSL from three perspectives:

- The labeled samples are effectively integrated into the graph learning process of GSSL by representing them as additional pairwise constraints.
- Both local and global data structures are considered to build a more flexible graph via self-expressive similarity metric and k-NN.
- A greedy-like strategy is designed to pick out more representative samples as the labeled set.

2 Preliminaries and Related Works

Given a data set $X = [x_1, x_2, \ldots, x_l, x_{l+1}, \ldots, x_n] \in \mathbb{R}^{m \times n}$, the subset $X_l = \{x_i\}_{i=1}^{l}$ with cardinality $|X_l| = l$ contains labeled points and $X_u = \{x_i\}_{i=l+1}^{n}$ with cardinality $|X_u| = n - l$ contains unlabeled points. The target of graph learning is to generate a proper graph or weight matrix $W \in \mathbb{R}^{n \times n}$ and its element W_{ij} denotes the similarity between the ith point and the jth point under some measurement. SR [6] and LRR [10] are two popular affinity representation techniques. SR aims to construct a sparse graph or $\ell 1$-graph [17], where each point could be reconstructed by a combination of other limited points, and thus the sparse coefficients correspond to a kind of similarity. Basic SR is formulated as the following optimization problem:

$$\min_{Z} \|Z\|_1 \quad \text{s.t. } X = XZ, diag(Z) = 0 \tag{1}$$

where $Z = [z_1, z_2, \ldots, z_n] \in \mathbb{R}^{n \times n}$ denotes the coefficient matrix, $\|Z\|_1$ is the $\ell 1$-norm of Z which can promote sparse solution, $\|Z\|_1 = \sum_{i=1}^{n} \sum_{j=1}^{n} |z_{ij}|$. Then, the graph weight matrix W could be easily obtained by $W = (|Z| + |Z|^T)/2$

Compared with the k-NN graph, the $\ell 1$-graph avoids evaluating the hyper-parameter k and therefore it outputs more robust result. Nevertheless, both $\ell 1$-graph and k-NN graph are lack of the global views of data, so their performance would be degenerated when there is no "clean" data available [22]. In order to capture the global data structure, Liu et al. [10] proposed LRR method which enforces a rank minimization constraint on the coefficient matrix. The basic LRR problem can be formulated as:

$$\min_{Z} \|Z\|_* \quad \text{s.t. } X = XZ \tag{2}$$

where $\|Z\|_*$ denotes the nuclear norm of Z, which is a usual surrogate of rank function, i.e., the sum of the singular values. Since the sparseness and low-rankness are merits of a graph, Zhuang et al. [22] presented a non-negative low-rank and sparse graph (NNLRS) learning method. Recently, Lu et al. [13] pointed out that, besides $\ell 1$-norm and nuclear norm, Frobenius norm is also an appropriate constraint for the coefficient matrix Z, and presented the LSR model with noise as follows:

$$\min_{Z} \lambda \|Z\|_F^2 + \|X - XZ\|_F^2 \tag{3}$$

where $\lambda > 0$ is the regularization parameter. Note there are little differences in (1), (2) and (3), but (3) has a close-form solution

$$Z^* = (X^T X + \lambda I)^{-1} X^T X \tag{4}$$

In this case, LSR can be solved efficiently.

Even though all the above approaches could output suitable graphs for GSSL, the graph learning itself is still unsupervised. In recent work [15], Shang et al. presented an enhanced spectral kernel (ESK) model, which makes use of pairwise constraints to favor graph learning, and is solved as a low-rank matrix approximation problem [9]. The main difference between ESK and our approach is as follows. ESK uses the Gaussian kernel to initialize the weight matrix, and encodes the known labels as the pairwise constraints. While in CLSR, we adopt the regression coefficients to measure the correlations among data points, and consider additional local constraints to promote the model's flexibility.

Additionally, the quality of labeled points play an important role in GSSL, thus, it is necessary to select the samples with high representability and discriminability as labeled set. In [5] and [11], k-means algorithm has been verified as an effective method for sample selection. But for GSSL, an extra step is needed to estimate the labels of clustering centers. Recently, some researchers pointed out that collaborative representation is a promising method for sample selection [18], [14]. In this paper, we propose a simple and effective method which applies minimal reconstruction error criterion to labeled sample selection.

3 Constrained Least Squares Regression for Graph Learning

In this section, we first introduce the label consistent penalty for encoding known labels, then integrate it with original LSR, and finally design its optimization algorithm.

3.1 Label Consistent Penalty

Given two sets for labeled points, $ML = \{(x_i, x_j)\}$ includes must-link constraints, where x_i and x_j have the same label, and $CL = \{(x_i, x_j)\}$ covers cannot-link constraints, where x_i and x_j have different labels. Let Ω be a set of indices which correspond to all pairwise constraints. The label consistent penalty is defined as:

$$f(Z) = \|S \circ Z - L\|_F^2 \tag{5}$$

where $\circ$ denotes the element-wise product. The sampling matrix $S \in \mathbb{R}^{n \times n}$ is defined as:

$$S_{ij} = \begin{cases} 1 & (i,j) \in \Omega \\ 0 & otherwise \end{cases} \tag{6}$$

The constraint matrix $L \in \mathbb{R}^{n \times n}$ is defined as:

$$L_{ij} = \begin{cases} 1 & (i,j) \in ML \\ 0 & (i,j) \in CL \\ 0 & otherwise \end{cases} \tag{7}$$

Equation (5) is a squared lose function to measure the consistency between the predicted affinity matrix induced by Z and the given pairwise constraints. Here, the pairwise constraints are expected to reflect the data structure. However, the number of labeled samples is usually few so that it is hard to sufficiently capture the essential structure of data with them. Thus, it is necessary to bring in more local pairwise constraints which are encoded as $L' \in \mathbb{R}^{n \times n}$:

$$L'_{ij} = \begin{cases} 1 & i \in N_j \text{ and } j \in N_i \\ 0 & otherwise \end{cases} \tag{8}$$

where N_i stands for the set of k-nearest neighbor of x_i. Actually, L' employs a k-NN graph to roughly recover the local relations among data points by 0/1 assignments, and thus it will result in some wrong assignments. One way to fix these incorrect assignments is to utilize the original L with correct assignments from labeled samples, and the fixed $L^f \in \mathbb{R}^{n \times n}$ is defined as:

$$L^f_{ij} = \begin{cases} L_{ij} & (i,j) \in ML \text{ or } (i,j) \in CL \\ L'_{ij} & (i,j) \notin ML \text{ and } (i,j) \notin CL \end{cases} \tag{9}$$

From the perspective of matrix approximation, these wrong assignments in (5) can be taken as one kind of sparse noise. Therefore, the $\ell 1$-norm is used here instead of the Frobenius norm and we have

$$f(Z) = \|S \circ Z - L\|_1 \tag{10}$$

3.2 Objection Function

After adding the label consistent penalty to the LSR model, the objective function of CLSR is written as:

$$\min_{Z,E} \|Z\|_F^2 + \frac{\lambda_e}{2}\|XZ - X\|_F^2 + \lambda_s\|E\|_1 \quad \text{s.t. } E = S \circ Z - L \tag{11}$$

where $E \in \mathbb{R}^{n\times n}$ denotes the sparse error, λ_e and λ_s are parameters to trade off other terms. In (11), the first two items are used to hold the global structure of data by Z and the third item introduces the pairwise constraints by L which is defined in (9).

3.3 Optimization

Equation (11) could be solved by the alternating direction method of multipliers (ADMM) [3] method. To start, we introduce an auxiliary matrix $A \in \mathbb{R}^{n\times n}$ for variables separation, then obtain

$$\min_{Z,E} \|Z\|_F^2 + \frac{\lambda_e}{2}\|XA - X\|_F^2 + \lambda_s\|E\|_1 \quad \text{s.t. } E = S \circ Z - L, Z = A \tag{12}$$

The augmented Lagrangian function of (12) can be written as:

$$\begin{aligned}\mathcal{L} = \min_{Z,E} \|Z\|_F^2 + \frac{\lambda_e}{2}\|XA - X\|_F^2 + \lambda_s\|E\|_1 + < Y_1, Z - A > \\ + < Y_2, E - S \circ Z + L > + \frac{\mu}{2}(\|Z - A\|_F^2 + \|E - S \circ Z + L\|_F^2)\end{aligned} \tag{13}$$

where $Y_1 \in \mathbb{R}^{n\times n}$ and $Y_2 \in \mathbb{R}^{n\times n}$ are two Lagrange multipliers. ADMM approach updates the variables Z, A and E alternately with other variables fixed, and we can get the updating rules as:

$$\begin{aligned}Z_{k+1} =& \arg\min_{Z} \|Z_k\|_F^2 + \frac{\mu_k}{2}\|Z_k - A_k + \frac{Y_1}{\mu_k}\|_F^2 + \frac{\mu_k}{2}\|S \circ Z_K - E_k - L - \frac{Y_2}{\mu_k}\|_F^2 \\ =& (\mathbf{1}/(\frac{2}{\mu_k} + \mathbf{1} + S)) \circ (A_k - \frac{Y_1}{\mu_k} + S \circ (\frac{Y_2}{\mu_k}) + S \circ E_k + S \circ L)\end{aligned} \tag{14}$$

$$\begin{aligned}A_{k+1} &= \arg\min_{A} \frac{\lambda_e}{2}\|XA_k - X\|_F^2 + \frac{\mu_k}{2}\|Z_{k+1} - A_k + \frac{Y_1}{\mu_k}\|_F^2 \\ &= (\lambda_e X^T X + \mu_k I)^{-1}(\lambda_e X^T X + \mu_k Z_{k+1} + Y_1)\end{aligned} \tag{15}$$

$$\begin{aligned}E_{k+1} =& \arg\min_{E} \lambda_s\|E_k\|_1 + \frac{\mu_k}{2}\|E_k - (S \circ Z_{k+1} - L - \frac{Y_2}{\mu_k})\|_F^2 \\ =& \mathcal{S}_{\frac{\lambda_s}{\mu_k}}(S \circ Z_{k+1} - L - \frac{Y_2}{\mu_k})\end{aligned} \tag{16}$$

where $\mathbf{1} \in \mathbb{R}^{n \times n}$ stands for an all-one matrix, and $\mathcal{S}_\mu(\cdot)$ is the shrinkage-thresholding operator [9] which is defined as:

$$\mathcal{S}_\mu(\nu) = sign(\nu)(|\nu| - \mu)_+ \tag{17}$$

The complete algorithm is summarized in Algorithm 1.

Algorithm 1. Solving Problem (11) via ADMM

Input: data matrix X, sampling matrix S, constraint matrix L, and parameters λ_e, λ_s.

1. Initialize $Z_0 = A_0 = E_0 = Y_1 = Y_2 = 0$, $\mu_0 = 0.1, \mu_{max} = 10^4, \rho = 1.1, \epsilon = 10^{-2}$
2. **while** not converged **do**
3. Update Z, A and E by (14-16).
4. Update the multipliers Y_1, Y_2 as:
   $Y_1 = Y_1 + \mu(Z - A)$,
   $Y_2 = Y_2 + \mu(E - S \circ Z + L)$.
5. Update $\mu : \mu = \min(\rho\mu, \mu_{max})$.
6. Check the convergence conditions:
   $||E - S \circ Z + L||_\infty < \epsilon$ and $||Z - A||_\infty < \epsilon$.
7. **end while**

Output: Z_k, E_k

4 Labeled Sample Selection via CLSR

In many real applications, we need select a small part of data set as a labeled set. Usually, a natural and simple method, random sampling is adopted. However, this method cannot guarantee the quality of labeled samples. Based on the subspace assumption, we could select a more representative data subset to upgrade graph's performance in GSSL.

In the CLSR framework, it is convenient to use the basic LSR model for labeled sample selection. We randomly select c subsets $\{X_i\}_{i=1}^c$ from $X, X_i \in \mathbb{R}^{m \times p}$ and each subset contains p samples, $p \ll n$. We consider each subset as a tiny dictionary and use it to reconstruct the whole data set, consequently, the representative ability of each subset could be ranked by the corresponding reconstruction error, therefore, the smaller reconstruction error it has, the more representative it is. The reconstruction error can be solved by

$$\min_{Z_i, E_i} \lambda \|Z_i\|_F^2 + \|E_i\|_F^2 \quad \text{s.t. } E_i = X - X_i Z_i \tag{18}$$

where $X_i \in \mathbb{R}^{m \times p}$ denotes the selected subset, $E_i \in \mathbb{R}^{m \times n}$ is the reconstruction error, and $Z_i \in \mathbb{R}^{p \times n}$ is the coefficient matrix of X_i. Note problem (18) has a close-form solution

$$Z_i = (X_i^T X_i + \lambda I)^{-1} X_i^T X \tag{19}$$

The labeled sample selection method is summarized in Algorithm 2.

Algorithm 2. Labeled Sample Selection via Minimal Reconstruction Error

Input: data matrix X, selected subset $\{X_i\}_{i=1}^{c}$, parameter λ.

1. Initialize $\lambda = 10$
2. **for** $i = 1, \ldots, c$ **do**
3. Get Z_i by (19).
4. Computer the reconstruction error $r_i(X_i) = \|X - X_i Z_i\|_F^2$.
5. **end for**
6. Find X_i^* with minimal reconstruction error $\arg\min_i r_i(X_i)$.

Output: X_i^*

5 CLSR for Semi-Supervised Classification

In this section, we integrate CLSR with a popular label propagation approach, LGC [19], for semi-supervised classification. Define a label set $F = \{1, \ldots, k\}$, and an initial label matrix $Y \in \mathbb{R}^{m \times k}$ with $Y_{ij} = 1$ for x_i is labeled as j and $Y_{ij} = 0$ otherwise. The iterative scheme for propagation is

$$Y_{k+1} = \alpha \overline{W} Y_k + (1 - \alpha) Y_0 \tag{20}$$

where $\overline{W}$ is a normalized affinity matrix with $\overline{W} = D^{-1/2} W D^{-1/2}$ and D is a diagonal matrix whose diagonal entries are equal to the sum of corresponding rows. We fix the parameter α to 0.01 in following experiments. The detail of the algorithm is summarized in Algorithm 3.

Algorithm 3. CLSR for Semi-Supervised Classification

Input: data matrix X, initial label matrix Y, parameters $\lambda_s, \lambda_e, \lambda$.

1. Initialize $\lambda = 10$
2. Get the labeled subset X_i by Algorithm 2 or random sampling.
3. Generate the sampling matrix S by (6) and the constraint matrix L by (9).
4. Get the coefficient matrix Z by Algorithm 1.
5. Normalize all column vectors of Z to unit-norm, $z_i = z_i / \|z_i\|_2$.
6. Get the weight matrix W by $W = (|Z| + |Z|^T)/2$.
7. Compute the label matrix Y by (20).

Output: Y

6 Experimental Results and Analysis

In this section, we evaluate the performance of CLSR and other popular graph construction methods on six public databases.

6.1 Datasets and Settings

We use two categories of public datasets in the experiments, including UCI data and image data (see Table 1).

1. **UCI data**[1]**.** We perform experiments on three UCI datasets including WDBC, Sonar and Parkinsons.
2. **Extended YaleB database**[2]**.** This face database contains 38 individuals under 9 poses and 64 illumination conditions. We choose the cropped images of first 10 individuals, and resize them to 48×42 pixels.
3. **ORL database**[3]**.** There are 40 distinct subjects and each of them has 10 different images. For some subjects, the images were taken at different times, varying the lighting, facial expressions and facial details. We resize them to 32×32 pixels.
4. **COIL20 database**[4]**.** This database consists of a set of gray-scale images with 20 objects. For each object, there are 72 images of size 32×32 pixels.

Table 1. Descriptions of datasets

Dataset	label Size	♯ of Features	♯ of Classes
WDBC	569	30	2
Sonar	208	60	2
Parkinsons	195	21	2
YaleB	640	2016	10
ORL	400	1024	40
COIL20	1440	1024	20

We compare following six graph construction algorithms. There are some parameters in each algorithm, and we tune the parameters on each dataset for every algorithm and record the best results.

1. ***k*-NN:** the Euclidean distance is used as similarity metric, and the Gaussian kernel is used to reweight the edges. The number of nearest neighbors is set to 5 for ***k*-NN5**, and 15 for ***k*-NN15**, respectively. The scale parameter of Gaussian kernel is set as [22]
2. **ESK:** Following the lines of [15], a low-rank kernel is learned as the affinity matrix. ESK model also use the Gaussian kernel to initialize the weight matrix.
3. **LSR:** Compared with CLSR, LSR [13] does not consider the pairwise constraints in graph leaning process.
4. **LRR:** Following [10], we construct the low-rank graph and adopt $\ell_{2,1}$-norm to model "sample-specific" corruptions.
5. **NNLRS:** Following [22], we construct the non-negative low-rank and sparse graph.
6. **CLSR:** In CLSR, the neighbor relations are encoded as the additional pairwise constraints for reflecting the local data structure. In the experiments, the sizes of nearest neighbors are set to 0, 5 and 15, respectively.

[1] `http://archive.ics.uci.edu/ml/`
[2] `http://vision.ucsd.edu/~leekc/ExtYaleDatabase/ExtYaleB.html`
[3] `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`
[4] `http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php`

6.2 Results and Discussions

All experiments are repeated 20 times, for each dataset, the label rate varies from 10% to 40%. Table 2 lists the average accuracies.

From Table 2 we can get following observations.

1. LSR, LRR and NNLRS generally outperform k-NN and ESK on YaleB and ORL datasets, as these datasets have roughly subspace structures. Correspondingly, datasets WDBC, Parkinsons cater to Euclidean distance-based measurement, so k-NN, ESK can work well on these datasets.
2. NNLRS usually achieves better performance than LSR and LRR, owing to it considers both sparseness and low-rankness of the graph.
3. ESK generally outperforms k-NN with the increasing of the sampling percentage, which testifies the effectiveness of integrating pairwise constraints into the graph learning process.
4. In most cases, CLSR outperforms other algorithms, since it takes advantage of both the self-expressive similarity and local constraints to enhance the model's flexibility and performance.

Table 2. Average accuracies (mean and standard deviation) of different graphs integrated with LGC label propagation strategy (The best results are highlighted in bold)

Dataset	k-NN5	k-NN15	ESK	LSR	LRR	NNLRS	CLSR
WDBC(10%)	93.56±1.26	93.54±0.80	92.41±1.50	89.01±1.70	91.14±1.48	91.11±1.43	**94.27±1.23**
WDBC(20%)	94.02±0.56	94.08±0.67	94.20±1.15	91.86±1.34	93.27±1.21	92.41±0.87	**95.09±0.39**
WDBC(30%)	94.84±0.55	94.87±0.63	94.90±0.74	93.51±1.10	94.15±1.03	93.59±0.76	**96.18±0.27**
WDBC(40%)	95.52±0.51	95.10±0.59	95.60±0.43	94.75±0.89	95.34±0.94	94.64±0.60	**96.85±0.26**
Sonar(10%)	73.44±4.21	73.41±4.48	67.18±4.34	67.90±4.32	68.37±7.55	71.06±3.98	**74.40±3.19**
Sonar(20%)	75.60±3.38	76.87±3.08	76.54±3.56	74.25±3.06	75.10±3.20	76.39±3.11	**81.38±2.99**
Sonar(30%)	79.71±2.03	78.32±3.25	81.82±3.13	79.46±2.42	80.94±2.92	81.39±2.05	**85.60±1.66**
Sonar(40%)	83.55±1.39	85.20±2.76	85.37±2.24	83.54±1.85	83.31±2.01	85.38±1.13	**88.75±0.94**
Parkinsons(10%)	75.82±6.23	72.66±3.93	75.27±6.18	67.12±3.54	74.11±3.33	76.10±3.48	**77.95±3.26**
Parkinsons(20%)	79.24±5.49	72.00±3.17	82.44±5.10	74.43±3.29	77.58±2.34	80.72±1.88	**83.59±2.19**
Parkinsons(30%)	80.57±4.66	72.29±2.93	85.91±4.78	79.24±2.80	81.66±1.86	82.87±1.55	**87.44±1.06**
Parkinsons(40%)	81.19±3.98	72.10±2.44	88.34±3.81	82.37±2.10	84.68±2.12	86.26±1.19	**89.05±0.85**
YaleB(10%)	69.06±2.25	66.98±6.48	60.23±2.51	87.80±1.74	87.88±2.11	**88.86±2.35**	88.85±1.65
YaleB(20%)	75.91±2.17	74.72±1.53	71.96±2.12	94.21±1.05	94.08±0.96	93.70±1.02	**94.37±0.88**
YaleB(30%)	79.58±1.89	78.64±2.11	77.82±1.23	96.28±0.94	96.20±0.79	95.39±0.58	**96.65±0.43**
YaleB(40%)	79.92±1.72	79.96±1.76	81.69±0.83	97.39±0.73	97.00±0.57	96.45±0.41	**97.69±0.35**
ORL(10%)	40.47±3.39	44.88±3.03	53.50±3.71	49.95±3.78	50.49±3.85	52.58±4.23	**54.52±3.07**
ORL(20%)	63.88±2.10	66.05±3.36	72.60±2.33	72.15±2.79	71.95±3.24	75.10±2.92	**76.65±2.39**
ORL(30%)	78.63±5.88	78.95±5.55	82.98±2.35	83.75±2.19	83.98±2.73	84.33±2.73	**85.50±2.14**
ORL(40%)	86.17±3.19	84.52±3.93	88.20±2.01	91.53±1.65	90.19±2.24	90.95±2.38	**92.50±1.91**
COIL20(10%)	86.12±0.81	85.80±1.01	86.24±1.21	80.10±1.52	79.38±2.54	81.39±1.55	**88.12±1.07**
COIL20(20%)	88.24±0.76	86.23±0.91	88.50±1.18	87.85±1.16	87.39±1.18	87.26±1.06	**90.84±0.73**
COIL20(30%)	89.88±0.85	87.82±1.02	90.69±0.77	91.35±0.92	90.57±1.11	89.96±0.81	**92.93±0.69**
COIL20(40%)	90.17±0.80	88.61±0.84	92.10±0.60	93.28±0.71	92.98±0.92	92.47±0.74	**94.58±0.55**

Next, we study the effectiveness of sample selection strategy based on minimal reconstruction error. We first randomly select 50 labeled subsets from each dataset, and then sort them in ascending order to form a subset-residual array according to the representative residual of each subset. Secondly, these labeled subsets are used as the supervisory information for classification and the average accuracies are recorded. Furthermore, another two results are listed for comparison, one is the average accuracy of the top 10% of the array (denoted as AT-10%),

the other is the average accuracy of the lowest 10% of the array (denoted as AL-10%). The percentage of labeled samples is 5% on WDBC, Parkinsons, YaleB, COIL20 and Sonar, because the selection strategy could be useful in case that there are only limited labeled samples available, especially, we select 20% of samples from ORL, since there are only 10 samples in each class of ORL.

The results are plotted in Fig. 1(a-f). It shows that our method is almost effective for all graph construction approaches on each dataset, except Parkinsons. The result on Parkinsons is unstable. The reason is that there are two classes in Parkinsons, but its imbalance ratio is nearly 3. In this case, our method tends to select more samples from the majority class to minimize the total reconstruction error, which leads to that the selected samples are incapable of capturing the true geometric structure of the dataset. We balance the sizes of two classes by randomly selecting some samples from the majority class, and the result shown in Fig. 1(g) is consistent with the other datasets'.

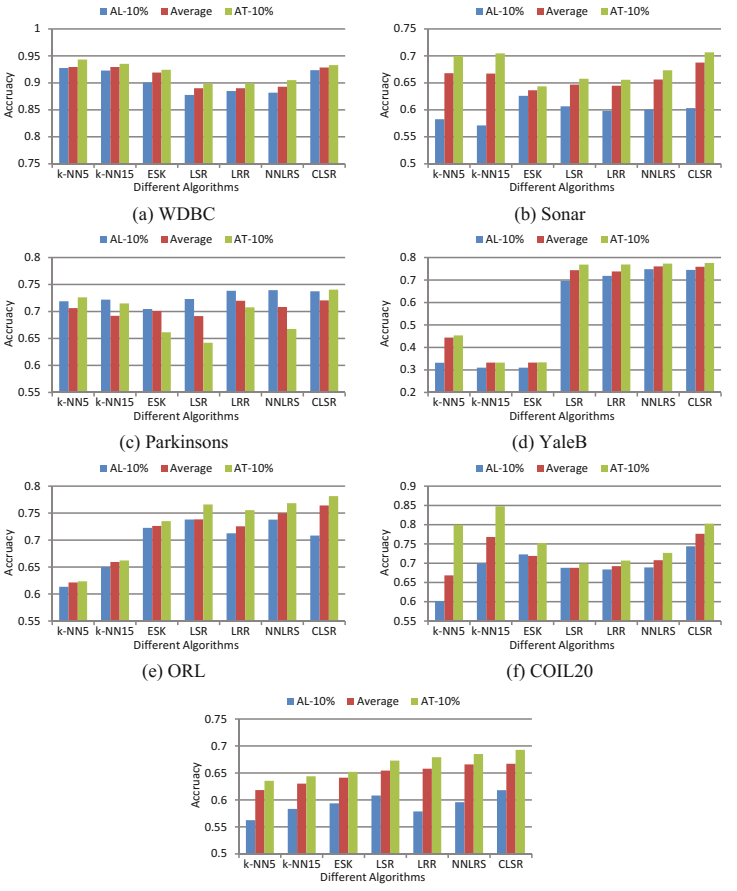

Fig. 1. Classification results of all graph construction algorithms on each dataset after applying sample selection strategy

7 Conclusion

We propose a new graph based semi-supervised learning approach called CLSR, which utilizes the pairwise constraints to guide the graph learning process. Beside the labeled information, there constraints also bring in local neighbor relations to enhance the graph's flexibility. In addition, based on CLSR, we design a labeled sample selection strategy which is used to select more representative points as a labeled set. Experimental results on real world datasets demonstrate the effectiveness of our method. Furthermore, given a small size of labeled set (e.g., 5% of total samples), our sample selection strategy could generally improve the performance of several state-of-the-art methods on most of the datasets used in the experiments.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Grant 61375062, Grant 61370129, and Grant 61033013, the Ph.D Programs Foundation of Ministry of Education of China under Grant 20120009110006, the National 863 project under Grant 2012AA040912, the Opening Project of State Key Laboratory of Digital Publishing Technology, and the Fundamental Research Funds for the Central Universities.

References

1. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15(6), 1373–1396 (2003)
2. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. The Journal of Machine Learning Research 7, 2399–2434 (2006)
3. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning 3(1), 1–122 (2011)
4. Chapelle, O., Weston, J., Schölkopf, B.: Cluster kernels for semi-supervised learning. In: Advances in Neural Information Processing Systems, pp. 585–592 (2002)
5. Chen, X., Cai, D.: Large scale spectral clustering with landmark-based representation. In: The 25th Conference on Artificial Intelligence, AAAI 2011 (2011)
6. Elhamifar, E., Vidal, R.: Sparse subspace clustering. In: Proceedings of the 22th Conference on Computer Vision and Pattern Recognition, pp. 2790–2797. IEEE (2009)
7. Jebara, T., Wang, J., Chang, S.F.: Graph construction and b-matching for semi-supervised learning. In: Proceedings of the 26th International Conference on Machine Learning, pp. 441–448. ACM (2009)
8. Li, Z., Liu, J., Tang, X.: Pairwise constraint propagation by semidefinite programming for semi-supervised classification. In: Proceedings of the 25th International Conference on Machine Learning, pp. 576–583 (2008)
9. Lin, Z., Chen, M., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. UIUC Technical report UILU-ENG-09-2215 (2010)

10. Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., Ma, Y.: Robust recovery of subspace structures by low-rank representation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 171–184 (2013)
11. Liu, W., He, J., Chang, S.F.: Large graph construction for scalable semi-supervised learning. In: Proceedings of the 27th International Conference on Machine Learning, pp. 679–686 (2010)
12. Liu, W., Wang, J., Chang, S.F.: Robust and scalable graph-based semisupervised learning. Proceedings of the IEEE 100(9), 2624–2638 (2012)
13. Lu, C.-Y., Min, H., Zhao, Z.-Q., Zhu, L., Huang, D.-S., Yan, S.: Robust and efficient subspace segmentation via least squares regression. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VII. LNCS, vol. 7578, pp. 347–360. Springer, Heidelberg (2012)
14. Peng, X., Zhang, L., Yi, Z.: Scalable sparse subspace clustering. In: IEEE Proceedings of the 26th Conference on Computer Vision and Pattern Recognition (2013)
15. Shang, F., Jiao, L., Liu, Y., Tong, H.: Semi-supervised learning with nuclear norm regularization. Pattern Recognition 46(8), 2323–2336 (2013)
16. Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science 290(5500), 2319–2323 (2000)
17. Yan, S., Wang, H.: Semi-supervised learning by sparse representation. In: SDM, pp. 792–801 (2009)
18. Zhang, L., Yang, M., Feng, X.: Sparse representation or collaborative representation: Which helps face recognition? In: Proceedings of the 12th International Conference on Computer Vision, pp. 471–478. IEEE (2011)
19. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. Advances in Neural Information Processing Systems 16(16), 321–328 (2004)
20. Zhu, X.: Semi-supervised learning literature survey. Technical report, Department of Computer Science, University of Wisconsin-Madison (2006)
21. Zhu, X., Ghahramani, Z., Lafferty, J., et al.: Semi-supervised learning using gaussian fields and harmonic functions. In: Proceedings of the 20th International Conference on Machine Learning, vol. 3, pp. 912–919 (2003)
22. Zhuang, L., Gao, H., Lin, Z., Ma, Y., Zhang, X., Yu, N.: Non-negative low rank and sparse graph for semi-supervised learning. In: Proceedings of the 25th Conference on Computer Vision and Pattern Recognition, pp. 2328–2335. IEEE (2012)