Prototype Vector Machine for Large Scale Semi-supervised Learning

Kai Zhang[1] James T. Kwok[2] Bahram Parvin[1]

[1]Life Science Division, Lawrence Berkeley National Lab

[2]Department of Computer Science and Engineering
Hong Kong University of Science and Technology

Outline

Outline

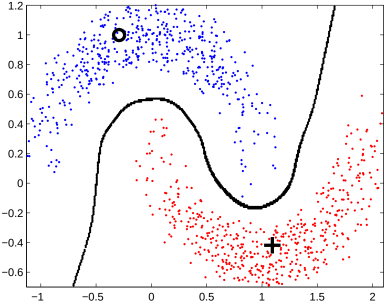
Semi-supervised Learning

Setting:

- limited supervision: $\{x_i, y_i\}_{i=1}^{l}$
- unlabeled data: $\{x_i\}_{i=l+1}^{n}$

Goal:

- prediction using both labeled and unlabeled samples



Transductive SVM

Transductive SVM

$$\min_{\{\overrightarrow{y}_i\}_{i=1}^u, w, b, \{\xi_i^*\}_{i=1}^u, \{\xi_i\}_{i=1}^l} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^l \xi_i + C^*\sum_{i=l+1}^n \xi_i^*$$

$$\text{s.t.} \quad y_i(w'x_i + b) \leq 1 - \xi_i$$

$$y_i^*(w'x_i + b) \leq 1 - \xi_i^*$$

- transductive SVM (text classification) [Joachims et al. 1999]
- linear SVM [Fung and Mangasarian 2001]
- SDP relaxations [Bie and Cristianini 2004] [Xu et al. 2008]
- CCCP optimization [Collobert et al. 2006]

Graph-based Methods

Graph Regularization (transductive)

$$\min_{\mathbf{f}=[\mathbf{f}_l' \ \mathbf{f}_u']'} \underbrace{\operatorname{tr}(\mathbf{f}'\mathcal{S}\mathbf{f})}_{\textit{smoothness}} + \underbrace{C_1 L(\mathbf{f}_l, \mathbf{Y}_l)}_{\textit{loss}} + \underbrace{C_2 \|\mathbf{f}_u\|_F^2}_{\textit{complexity}} \tag{1}$$

- $\mathcal{S}$: (normalized) Graph Laplacian

Examples:

- local and global consistency [Zhou et al. 2003]
- Gaussian fields and harmonic function [Zhu et al. 2003]
- nonparametric function induction [Delalleau et al. 2005]

Graph-based Methods

Manifold Regularization (inductive)

$$\min_f \sum_{i=1}^{l} L(f(\mathbf{x}_i), \mathbf{y}_i) + \gamma_A \|f\|_{\mathcal{K}} + \gamma_I \|f\|_{\mathcal{G}}$$

$$\Rightarrow f(\mathbf{x}) = \sum_{i=1}^{l+u} \alpha_i K(\mathbf{x}, \mathbf{x}_i)$$

- manifold regularization [Belkin 2002]
  - Lap-RLS, Lap-SVM

Fast graph-based SSL Methods

Fast algorithms ($O(m^2n)$)

- Harmonic mixture [Zhu et al. 2002]
  - combine generative model with graph-method
- Nonparametric function induction [Delalleau et al. 2005]
  - label reconstruction by landmark points
  - ignores important regularization
- Nyström method [Gustavo et al. 2007]
  - speed up kernel matrix inverse

Survey

- Semi-supervised learning literature survey [Zhu]
- Large scale semi-supervised learning [Weston]

Outline

1. Semi-supervised Learning
   - Transductive SVM
   - Graph-based Methods
   - Scaling up graph-based SSL
2. Prototype Vector Machine
   - Approximation via Prototypes
   - Low-rank Approximation Prototype
   - Label Reconstruction Prototype
   - Optimization
3. Experiments
4. Conclusion

Observation

Regularization: bottleneck of graph-based SSL

- manipulation of $n \times n$ kernel matrix
  - multiplication
  - inverse
- lead to complex model
  - spans over labelled and unlabeled data $f(\mathbf{x}) = \sum_{i=1}^{l+u} \alpha_i K(\mathbf{x}, \mathbf{x}_i)$
  - slow training and testing

Basic Idea

Basic idea: approximate regularization via **prototypes**

1. Low-rank approximation prototypes
   - preserve structures of kernel matrix
   - crucial for manifold regularization
   - less space
2. Label-reconstruction prototypes
   - reduce model complexity
   - fast testing

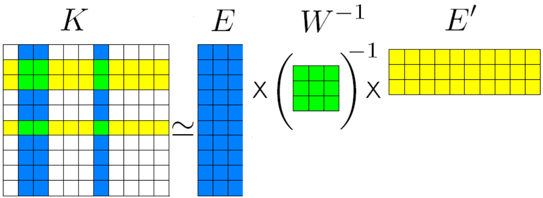
Low-rank Approximation

Given $n \times n$ kernel matrix K (on $\mathcal{X}$)

- find $K \approx GG'$, $G \in \mathbb{R}^{n \times m}$ $(m \ll n)$

Nyström Method

1. Choose $m \ll n$ columns $E_{n \times m}$
   - corresponds to landmark set $\mathcal{Z}$, $_{|\mathcal{Z}|=m}$
   - $W_{m \times m}$: kernel matrix on $\mathcal{Z}$
2. Reconstruct by $K \approx EW^{-1}E'$



Low-rank Approximation

$\mathbf{z}_i's \in \mathcal{Z}$: low-rank approximation prototypes

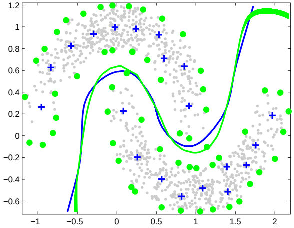
- can be chosen as k-means clustering centers for
  - Gaussian
  - linear
  - polynomial

detailed analysis in [Zhang et. al. 2008]

Nyström low-rank approximation quality depends on the encoding power of landmark points.

Label Reconstruction

A small set of prototypes (with labels estimated) can reconstruct the overall label landscape.



Label reconstruction: $g(\mathbf{x}) = \sum_{i=1}^{k} \mathbf{f}_i K(\mathbf{x}, \mathbf{v}_i)$ or $\mathbf{f} = H\mathbf{f}_v$
$\mathbf{v}_i$'s: label reconstruction prototypes

Information Theoretic Measures

Using g to approximate f:

$$\min_{\beta_i, \mathbf{v}_i} D(\underbrace{\sum_{i=1}^{l+u} \alpha_i K(\mathbf{x}, \mathbf{x}_i)}_{f(\mathbf{x})}, \underbrace{\sum_{i=1}^{m} \beta_i K(\mathbf{x}, \mathbf{v}_i)}_{g(\mathbf{x})})$$

- α_i's unknown

alternative: basis in f should be well-coded by those in g.

$$\mathcal{Q} = \sum_{i=1}^{l+u} \sum_{j=1}^{k} \min D_{KL} \left[K(\mathbf{x}, \mathbf{x}_i) || K(\mathbf{x}, \mathbf{v}_j) \right]$$

Gaussian kernel K $\Rightarrow \mathcal{Q} = \frac{1}{4h^2} \sum_i \sum \min \|\mathbf{x}_i - \mathbf{v}_j\|^2 \Rightarrow k$-means centers as $\mathbf{v}_j$'s.

Rephrasing Optimization with Prototypes

Two types of prototypes

1. low-rank approximation $K \approx \mathbf{E}W^{-1}\mathbf{E}'$
   - $E \in \mathbb{R}^{n\times m}$, $W \in \mathbb{R}^{m\times m}$,
2. label reconstruction $f \approx \mathbf{H}\mathbf{f}_v$
   - $f \in \mathbb{R}^{n\times 1}$; $\mathbf{f}_v \in \mathbb{R}^{k\times 1}$,$\mathbf{H} \in \mathbb{R}^{n\times k}$

Regularization can be approximated by

$$\mathbf{f}^\top \mathcal{S}\mathbf{f} \approx \mathbf{f}_v' \underbrace{\mathbf{H}'(\tilde{D} - \mathbf{E}W^{-1}\mathbf{E}^\top)\mathbf{H}}_{O((m+k)^2 n)} \mathbf{f}_v$$

L_2 Loss Function

- multiclass, L_2-loss function
- labels $\mathbf{Y}_l \in \mathbb{R}^{l \times C}$,

$$\min_{\mathbf{f}_v \in \mathbb{R}^{m \times k}} \operatorname{tr}\left((\mathbf{H}\mathbf{f}_v)'\mathcal{S}(\mathbf{H}\mathbf{f}_v)\right) + C_1\|\mathbf{H}_l\mathbf{f}_v - \mathbf{Y}_l\|_F^2 + C_2\|\mathbf{H}_u\mathbf{f}_v\|_F^2$$

training $$\mathbf{f}_v^* = (\mathbf{H}'\mathcal{S}\mathbf{H} + C_1\mathbf{H}_l'\mathbf{H}_l + C_2\mathbf{H}_u'\mathbf{H}_u)^{-1}\mathbf{E}_l'\mathbf{Y}_l$$

testing $$\mathbf{f} = \mathbf{H}\mathbf{f}_v$$

$$O(n(m+k)^2) \text{ time}$$

Hinge Loss Function

- binary, $\mathbf{Y}_l \in \{\pm 1\}^{l \times 1}$, Hinge loss,
- $\mathbf{H}_l = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_l]^\top$
- $A = \mathbf{H}^\top \mathcal{S} \mathbf{H} + C_2 \mathbf{H}_u^\top \mathbf{H}_u \in \mathbb{R}^{k \times k}$
- $\mathbf{Q} = \mathbf{H}_l A^{-1} \mathbf{H}_l^\top \odot \mathbf{Y}_l \mathbf{Y}_l^\top \in \mathbb{R}^{l \times l}$

$$
\begin{array}{llll}
\textit{Primal} & \displaystyle\min_{\mathbf{f}_v \in \mathbb{R}^{m \times 1}} & \displaystyle\frac{1}{2} \mathbf{f}_v^\top A \mathbf{f}_v + C_1 \sum_{i=1}^{l} \xi_i \\
 & \text{s.t.} & y_i \mathbf{e}_i^\top \mathbf{f}_v \geq 1 - \xi_i, \quad \xi_i \geq 0
\end{array}
$$

$$
\begin{array}{lll}
\textit{Dual} & \max & \displaystyle -\frac{1}{2} \beta^\top \mathbf{Q} \beta + \mathbf{1}_l^\top \beta \\
 & \text{s.t.} & 0 \leq \beta_i \leq C_1, \; i = 1, 2, ..., l.
\end{array}
$$

Outline

Experimental Setting

- methods compared
  - LGC: local and global consistency;
  - Lap-RLS: Laplacian-regularized RLS;
  - NYS-LGC: Nyström-based LGC;
  - NFI: nonparametric function induction;
  - PVM(1): L_2 loss;
  - PVM(2) Hinge loss
- 15 data sets (semi-supervised learning, libsvm)
- Gaussian kernel ($m = k$).
- $m = 0.1n$ for $n \leq 3000$; $m = 200$ for larger n
- 50 labels per class; randomly repeat 30 times

Benchmark Data

Classification errors of different algorithms.

Data(#cls)	LGC	LAP-RLS	NYS-LGC	NFI	PVM(1)	PVM(2)
g241c(2)	**21.92**	**22.02**	24.19	28.07	24.50	23.21
g241d(2)	28.10	**22.36**	30.98	30.82	25.15	24.85
digit1(2)	5.74	5.74	6.68	9.83	4.18	**3.72**
USPS(2)	**4.57**	6.11	9.72	5.49	5.29	6.35
coil_2(2)	14.37	10.83	16.90	13.98	**11.69**	14.85
coil(6)	**12.38**	21.17	18.75	30.93	13.41	–
BCI(2)	44.43	**29.16**	45.45	45.67	33.59	31.65
Text(2)	**23.09**	23.99	34.40	32.54	30.4	26.29
usps3589(4)	**2.46**	4.54	6.89	7.14	3.66	–
splice(2)	22.85	**19.78**	30.56	34.56	23.47	25.32
dna(3)	27.31	17.72	29.53	43.38	**15.87**	–
svmgd1a(2)	–	–	6.32	14.21	**5.24**	6.08
usps-full(10)	–	–	17.68	14.43	**7.35**	–
satimage(6)	–	–	16.36	19.27	**14.97**	–

Benchmark Data

Time consumptions (seconds) of different algorithms.

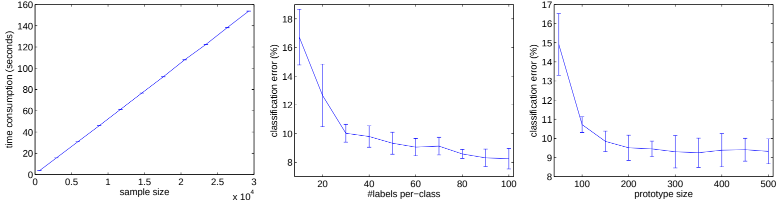
Data(n/dim)	LGC	LAP-RLS	NYS-LGC	NFI	PVM(1)	PVM(2)
g241c(1500/241)	140.84	129.86	0.86	0.48	3.30	3.19
g241d(1500/241)	129.78	142.65	0.84	0.49	3.31	3.16
digit1(1500/241)	140.51	131.08	0.84	0.48	3.31	3.15
USPS(1500/241)	139.23	131.59	0.74	0.47	3.28	3.14
$coil_2$(1500/241)	151.36	120.48	0.87	0.48	3.26	3.47
coil(1500/241)	146.92	115.22	0.79	0.49	3.35	–
BCI(400/117)	3.08	1.94	0.53	0.22	0.71	1.09
Text(1500/11960)	139.67	216.37	9.14	13.26	30.24	34.24
2-moon(1000/2)	49.76	16.11	0.026	0.24	0.083	0.21
usps3589(719/64)	13.94	13.13	0.15	0.086	0.37	–
splice(3175/60)	1622.51	1439.51	2.49	0.83	4.87	4.24
dna(3186/180)	1566.91	1463.75	3.07	1.22	8.92	–
svmgd1a(7089/4)	–	–	3.22	1.66	8.06	5.38
usps-full(7291/256)	–	–	3.96	2.87	22.48	–
satimage(6435/36)	–	–	3.34	2.57	11.56	–

Case Study

Five-class classification

- MNIST digits 3,5,6,8,9
- n = 29270; dim = 784
- algorithm properties
  - scalability
  - performance over # labels
  - performance over prototype size

Properties of PVM(1)



From left to right: time v.s. sample size; error v.s. #labels; error v.s.#prototypes.

Outline

Conclusions

- Conclusion
  - Computational bottleneck of Graph-based SSL
    - the regularization term
    - alleviated by using prototype approximations
- Future work
  - prototype selection
    - under different kernels
    - using label information
  - different label reconstruction schemes

Q & A

Thank you!