Efficient Maximum Margin Clustering

Changshui Zhang, Bin Zhao

Dept. Automation, Tsinghua Univ.

MLA, Nov. 8, 2008
Nanjing, China

Outline

1. Motivation
2. Two-Class Maximum Margin Clustering
3. Multi-Class Maximum Margin Clustering
4. Related Works
5. Conclusions

Outline

1 Motivation

2 Two-Class Maximum Margin Clustering

3 Multi-Class Maximum Margin Clustering

4 Related Works

5 Conclusions

Support Vector Machine

Given $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$, $\mathbf{y} = (y_1, \ldots, y_n) \in \{-1, +1\}^n$, *SVM* finds a hyperplane $f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b$ by solving

$$\begin{aligned} min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ s.t. \quad & y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad i = 1, \ldots, n \end{aligned} \tag{1}$$

Maximum Margin Clustering [Xu et. al. 2004]

MMC targets to find not only the optimal hyperplane $(\mathbf{w}^*, b^*)$, but also the optimal labeling vector $\mathbf{y}^*$

$$\begin{aligned} \min_{\mathbf{y}\in\{-1,+1\}^n} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ s.t. \quad & y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad i = 1, \ldots, n \end{aligned} \tag{2}$$

Representative Works

Semi-definite programming [*Xu et. al. (NIPS 2004)*]

- Several relaxations made
- n^2 variables in SDP
- Time complexity $O(n^7)$

Representative Works

Semi-definite programming [*Valizadegan and Jin (NIPS 2006)*]

- Reduce number of variables from n^2 to n
- Time complexity $O(n^4)$
- Only 2-class scenario

Representative Works

Alternating optimization [*Zhang et. al. (ICML 2007)*]

- Involve a sequence of QPs
- Number of iterations not guaranteed theoretically
- Only 2-class scenario

Outline

1. Motivation
2. Two-Class Maximum Margin Clustering
3. Multi-Class Maximum Margin Clustering
4. Related Works
5. Conclusions

Problem Reformulation

Theorem

Maximum margin clustering is equivalent to

$$\begin{aligned} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ s.t. \quad & |\mathbf{w}^T\phi(\mathbf{x}_i) + b| \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad i = 1, \dots, n \end{aligned} \tag{3}$$

where the labeling vector $y_i = \mathrm{sign}(w^T\phi(x_i) + b)$.

Problem Reformulation

Theorem

Any solution $(\mathbf{w}^*, b^*)$ *to problem (4) is also a solution to problem (3) (and vice versa), with* $\xi^* = \frac{1}{n}\sum_{i=1}^{n}\xi_i^*$.

$$\begin{aligned} \min_{\mathbf{w},b,\xi\geq 0} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\xi \\ s.t. \quad & \forall\, \mathbf{c}\in\{0,1\}^n : \\ & \frac{1}{n}\sum_{i=1}^{n} c_i|\mathbf{w}^T\phi(\mathbf{x}_i)+b| \geq \frac{1}{n}\sum_{i=1}^{n} c_i - \xi \end{aligned} \tag{4}$$

Problem Reformulation

- Number of variables reduced by $2n - 1$
- Number of constraints increased from n to 2^n
- We can always find a polynomially sized subset of constraints, with which the solution of the relaxed problem fulfills all constraints from problem (4) up to a precision of ϵ.

Cutting Plane Algorithm [J. E. Kelley 1960]

- Starts with an empty constraint subset Ω
- Computes the optimal solution to problem (4) subject to the constraints in Ω
- Finds the most violated constraint in problem (4) and adds it into the subset Ω
- Stops when no constraint in (4) is violated by more than ϵ

$$\frac{1}{n}\sum_{i=1}^{n} c_i |\mathbf{w}^T\phi(\mathbf{x}_i)+b| \geq \frac{1}{n}\sum_{i=1}^{n} c_i - (\xi+\epsilon) \tag{5}$$

The Most Violated Constraint

Theorem

The most violated constraint could be computed as follows

$$c_i = \begin{cases} 1 & \text{if } |\mathbf{w}^T\phi(\mathbf{x}_i)+b| < 1 \\ 0 & \textit{otherwise} \end{cases} \tag{6}$$

The feasibility of a constraint is measured by the corresponding value of ξ

$$\frac{1}{n}\sum_{i=1}^{n} c_i|\mathbf{w}^T\phi(\mathbf{x}_i)+b| \geq \frac{1}{n}\sum_{i=1}^{n} c_i - \xi \tag{7}$$

Enforcing the Class Balance Constraint

Enforce class balance constraint to avoid trivially "optimal" solutions

$$\min_{\mathbf{w},b,\xi\geq 0} \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\xi \tag{8}$$

$$s.t.\ \forall \mathbf{c}\in\Omega: \frac{1}{n}\sum_{i=1}^{n} c_i|\mathbf{w}^T\phi(\mathbf{x}_i)+b|\geq \frac{1}{n}\sum_{i=1}^{n} c_i-\xi$$

$$-l\leq \sum_{i=1}^{n}\left(\mathbf{w}^T\phi(\mathbf{x}_i)+b\right)\leq l$$

The Constrained Concave-Convex Procedure [A. J. Smola et.al. 2005]

Solve non-convex optimization problem whose objective function could be expressed as a difference of convex functions

$$\begin{aligned} \min_{\mathbf{z}} \quad & f_0(\mathbf{z}) - g_0(\mathbf{z}) \\ s.t. \quad & f_i(\mathbf{z}) - g_i(\mathbf{z}) \leq c_i \quad i = 1, \ldots, n \end{aligned} \tag{9}$$

where f_i and g_i are real-valued convex functions on a vector space $\mathcal{Z}$ and $c_i \in \mathcal{R}$ for all $i = 1, \ldots, n$.

The Constrained Concave-Convex Procedure

Given an initial point $\mathbf{z}_0$, the *CCCP* computes $\mathbf{z}_{t+1}$ from $\mathbf{z}_t$ by replacing $g_i(\mathbf{z})$ with its first-order Taylor expansion at $\mathbf{z}_t$

$$\begin{aligned} \min_{\mathbf{z}} \quad & f_0(\mathbf{z}) - T_1\{g_0, \mathbf{z}_t\}(\mathbf{z}) \\ s.t. \quad & f_i(\mathbf{z}) - T_1\{g_i, \mathbf{z}_t\}(\mathbf{z}) \leq c_i \quad i = 1, \ldots, n \end{aligned} \tag{10}$$

Optimization via the *CCCP*

By substituting first-order Taylor expansion into problem (8), we obtain the following *quadratic programming (QP)* problem:

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \ & \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\xi \qquad (11) \\ s.t. \ & \xi \geq 0 \\ & -l \leq \sum_{i=1}^{n}\left(\mathbf{w}^T\phi(\mathbf{x}_i)+b\right) \leq l \\ & \forall \mathbf{c}\in\Omega: \frac{1}{n}\sum_{i=1}^{n}c_i - \xi - \frac{1}{n}\sum_{i=1}^{n}c_i \mathrm{sign}(\mathbf{w}_t^T\phi(\mathbf{x}_i)+b_t)\left[\mathbf{w}^T\phi(\mathbf{x}_i)+b\right] \leq 0 \end{aligned}$$

Justification of *CPMMC*

Theorem

For any dataset $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ *and any* $\epsilon > 0$*, the CPMMC algorithm for maximum margin clustering returns a point* $(\mathbf{w}, b, \xi)$ *for which* $(\mathbf{w}, b, \xi + \epsilon)$ *is feasible in problem (4).*

Time Complexity Analysis

Theorem

Each iteration of CPMMC takes time $O(sn)$ for a constant working set size $|\Omega|$.

Theorem

For any $\epsilon > 0$, $C > 0$, and any dataset $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, the CPMMC algorithm terminates after adding at most $\frac{CR}{\epsilon^2}$ constraints, where R is a constant number independent of n and s.

Time Complexity Analysis

Theorem

For any dataset $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ with n samples and sparsity of s, and any fixed value of $C > 0$ and $\epsilon > 0$, the CPMMC algorithm takes time $O(sn)$.

Clustering Error Comparison

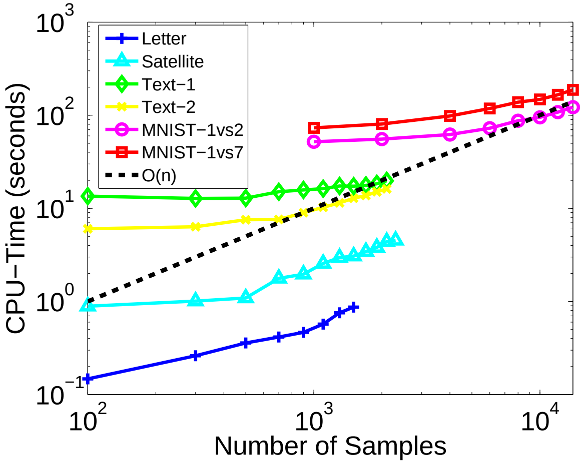
Data	Size	KM	NC	MMC	GMMC	IterSVR	CPMMC
Digits 3-8	357	5.32$\pm$ 0	35	10	5.6	3.36$\pm$ 0	**3.08**
Digits 1-7	361	0.55$\pm$ 0	45	31.25	2.2	0.55$\pm$ 0	**0.0**
Digits 2-7	356	3.09$\pm$ 0	34	1.25	0.5	**0.0$\pm$ 0**	**0.0**
Digits 8-9	354	9.32$\pm$ 0	48	3.75	16.0	3.67$\pm$ 0	**2.26**
Ionosphere	351	32$\pm$ 17.9	25	**21.25**	23.5	32.3$\pm$ 16.6	27.64
Letter	1555	17.94$\pm$ 0	23.2	-	-	7.2$\pm$ 0	**5.53**
Satellite	2236	4.07$\pm$ 0	4.21	-	-	3.18$\pm$ 0	**1.52**
Text-1	1980	49.47$\pm$0	6.21	-	-	**3.18$\pm$ 0**	5.00
Text-2	1989	49.62$\pm$0	8.65	-	-	6.01$\pm$ 1.82	**3.72**
UCI digits	1797	3.62	2.43	-	-	1.82	**0.62**
MNIST digits[1]	70000	10.79	10.08	-	-	7.59	**4.29**

[1] For UCI digits and MNIST datasets, we give a through comparison by considering all 45 pairs of digits 0- 9. For NC/MMC/GMMC/IterSVR, results on the digits and ionosphere data are simply copied from (Zhang et. al., 2007).

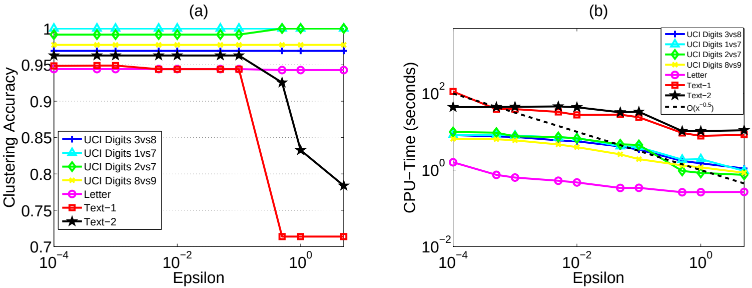
Speed of *CPMMC*

Data	KM	NC	GMMC	IterSVR	CPMMC
Digits 3-8	0.51	0.12	276.16	19.72	1.10
Digits 1-7	0.54	0.13	289.53	20.49	0.95
Digits 2-7	0.50	0.11	304.81	19.69	0.75
Digits 8-9	0.49	0.11	277.26	19.41	0.85
Ionosphere	0.07	0.12	273.04	18.86	0.78
Letter	0.08	2.24	-	2133	0.87
Satellite	0.19	5.01	-	6490	4.54
Text-1	66.09	6.04	-	5844	19.75
Text-2	52.32	5.35	-	6099	16.16

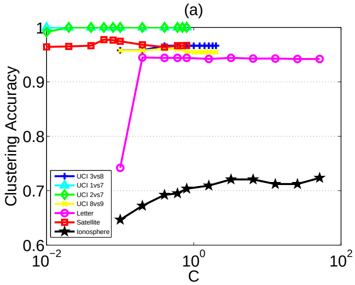
Dataset Size *n* vs. Speed



ϵ vs. Accuracy & Speed



C vs. Accuracy & Speed



Outline

1. Motivation
2. Two-Class Maximum Margin Clustering
3. Multi-Class Maximum Margin Clustering
4. Related Works
5. Conclusions

Multi-Class Support Vector Machine [Crammer & Singer 2001]

Given a point set $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ and their labels $\mathbf{y} = (y_1, \ldots, y_n) \in \{1, \ldots, k\}^n$, *SVM* defines a weight vector $\mathbf{w}_p$ for each class $p \in \{1, \ldots, k\}$ and classifies sample $\mathbf{x}$ by $y^* = \arg\max_{y \in \{1,\ldots,k\}} \mathbf{w}_y^T \mathbf{x}$ with the weight vectors obtained as

$$\min_{\mathbf{w}_1,\ldots,\mathbf{w}_k,\xi} \quad \frac{1}{2}\beta \sum_{p=1}^{k} ||\mathbf{w}_p||^2 + \sum_{i=1}^{n} \xi_i \tag{12}$$

$$s.t. \quad \forall i = 1, \ldots, n, r = 1, \ldots, k$$

$$\mathbf{w}_{y_i}^T \mathbf{x}_i + \delta_{y_i,r} - \mathbf{w}_r^T \mathbf{x}_i \geq 1 - \xi_i$$

Multi-Class Maximum Margin Clustering

Similar with the binary clustering scenario

$$\begin{aligned}\min_{\mathbf{y}} \min_{\mathbf{w}_1,\ldots,\mathbf{w}_k,\xi} \quad & \frac{1}{2}\beta\sum_{p=1}^{k}||\mathbf{w}_p||^2+\frac{1}{n}\sum_{i=1}^{n}\xi_i \\ s.t. \quad & \forall i=1,\ldots,n, r=1,\ldots,k \\ & \mathbf{w}_{y_i}^T\mathbf{x}_i+\delta_{y_i,r}-\mathbf{w}_r^T\mathbf{x}_i\geq 1-\xi_i\end{aligned} \tag{13}$$

Problem Reformulation I

Theorem

$$\begin{aligned}
\min_{\mathbf{w}_1,\ldots,\mathbf{w}_k,\xi} \quad & \frac{1}{2}\beta\sum_{p=1}^{k}||\mathbf{w}_p||^2+\frac{1}{n}\sum_{i=1}^{n}\xi_i \\
s.t. \quad & \forall i=1,\ldots,n, r=1,\ldots,k \\
& \sum_{p=1}^{k}\mathbf{w}_p^T\mathbf{x}_i\prod_{q=1,q\neq p}^{k}I_{(\mathbf{w}_p^T\mathbf{x}_i>\mathbf{w}_q^T\mathbf{x}_i)}+\prod_{q=1,q\neq r}^{k}I_{(\mathbf{w}_r^T\mathbf{x}_i>\mathbf{w}_q^T\mathbf{x}_i)}-\mathbf{w}_r^T\mathbf{x}_i\geq 1-\xi_i
\end{aligned} \tag{14}$$

where $I(\cdot)$ is the indicator function and the label for sample $\mathbf{x}_i$ is determined as $y_i=\sum_{p=1}^{k}p\prod_{q=1,q\neq p}^{k}I_{(\mathbf{w}_p^T\mathbf{x}_i>\mathbf{w}_q^T\mathbf{x}_i)}$

Problem Reformulation II

Theorem

Problem (14) can be equivalently formulated as problem (15), with $\xi^* = \frac{1}{n}\sum_{i=1}^{n}\xi_i^*$.

$$\min_{\mathbf{w}_1,\ldots,\mathbf{w}_k,\xi} \frac{1}{2}\beta\sum_{p=1}^{k}||\mathbf{w}_p||^2+\xi \tag{15}$$

$$s.t.\ \forall \mathbf{c}_i \in \{\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_k\},\ i=1,\ldots,n$$

$$\frac{1}{n}\sum_{i=1}^{n}\left\{\mathbf{c}_i^T\mathbf{e}\sum_{p=1}^{k}\mathbf{w}_p^T\mathbf{x}_i z_{ip}+\sum_{p=1}^{k}c_{ip}(z_{ip}-\mathbf{w}_p^T\mathbf{x}_i)\right\}\geq\frac{1}{n}\sum_{i=1}^{n}\mathbf{c}_i^T\mathbf{e}-\xi$$

where $z_{ip} = \prod_{q=1,q\neq p}^{k} I_{(\mathbf{w}_p^T\mathbf{x}_i > \mathbf{w}_q^T\mathbf{x}_i)}$ *and each constraint* $\mathbf{c}$ *is represented as a* $k \times n$ *matrix* $\mathbf{c} = (\mathbf{c}_1, \ldots, \mathbf{c}_n)$.

Problem Reformulation

- Number of variables reduced by $2n - 1$
- Number of constraints increased from nk to $(k + 1)^n$
- Targets to finding a small subset of constraints, with which the solution of the relaxed problem fulfills all constraints from problem (15) up to a precision of ϵ.

Cutting Plane Algorithm [J. E. Kelley 1960, T. Joachims 2006]

- Starts with an empty constraint subset Ω
- Computes the optimal solution to problem (15) subject to the constraints in Ω
- Finds the most violated constraint in problem (15) and adds it into the subset Ω
- Stops when no constraint in (15) is violated by more than ϵ

$$\forall \mathbf{c}_i \in \{\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_k\}^n,\ i = 1, \ldots, n \tag{16}$$

$$\frac{1}{n}\sum_{i=1}^{n}\left\{\mathbf{c}_i^T\mathbf{e}\sum_{p=1}^{k}\mathbf{w}_p^T\mathbf{x}_i z_{ip}+\sum_{p=1}^{k}c_{ip}(z_{ip}-\mathbf{w}_p^T\mathbf{x}_i)\right\}\geq\frac{1}{n}\sum_{i=1}^{n}\mathbf{c}_i^T\mathbf{e}-\xi-\epsilon$$

The Most Violated Constraint

Theorem

Define $p^* = \arg\max_p(\mathbf{w}_p^T\mathbf{x}_i)$ *and* $r^* = \arg\max_{r\neq p^*}(\mathbf{w}_r^T\mathbf{x}_i)$ *for* $i = 1,\ldots,n$*, the most violated constraint could be calculated as follows*

$$\mathbf{c}_i = \begin{cases} \mathbf{e}_{r^*} & \textit{if } (\mathbf{w}_{p^*}^T\mathbf{x}_i - \mathbf{w}_{r^*}^T\mathbf{x}_i) < 1 \\ \mathbf{0} & \textit{otherwise} \end{cases}, \; i = 1,\ldots,n \tag{17}$$

Enforcing the Class Balance Constraint

To avoid trivially “optimal” solutions

$$\min_{\mathbf{w}_1,\ldots,\mathbf{w}_k,\xi\geq 0} \quad \frac{1}{2}\beta\sum_{p=1}^{k}||\mathbf{w}_p||^2+\xi \tag{18}$$

$$s.t. \quad \frac{1}{n}\sum_{i=1}^{n}\left\{\mathbf{c}_i^T\mathbf{e}\sum_{p=1}^{k}\mathbf{w}_p^T\mathbf{x}_i z_{ip}+\sum_{p=1}^{k}c_{ip}(z_{ip}-\mathbf{w}_p^T\mathbf{x}_i)\right\}$$

$$\geq \frac{1}{n}\sum_{i=1}^{n}\mathbf{c}_i^T\mathbf{e}-\xi,\ \forall[\mathbf{c}_1,\ldots,\mathbf{c}_n]\in\Omega$$

$$-l\leq\sum_{i=1}^{n}\mathbf{w}_p^T\mathbf{x}_i-\sum_{i=1}^{n}\mathbf{w}_q^T\mathbf{x}_i\leq l,\ \forall p,q=1,\ldots,k$$

Optimization via the *CCCP*

Calculate the subgradients

$$\partial_{\mathbf{w}_r}\left\{\frac{1}{n}\sum_{i=1}^{n}\left[\mathbf{c}_i^T\mathbf{e}\sum_{p=1}^{k}\mathbf{w}_p^T\mathbf{x}_i z_{ip}+\sum_{p=1}^{k}c_{ip}z_{ip}\right]\right\}\Bigg|_{\mathbf{w}=\mathbf{w}^{(t)}} \tag{19}$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbf{c}_i^T\mathbf{e}z_{ip}^{(t)}\mathbf{x}_i \qquad \forall r = 1,\ldots,k$$

By substituting first-order Taylor expansion into problem (18), we obtain a *quadratic programming (QP)* problem.

Justification of *CPM3C*

Theorem

For any dataset $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ *and any* $\epsilon > 0$*, the CPM3C algorithm returns a point* $(\mathbf{w}_1, \ldots, \mathbf{w}_k, \xi)$ *for which* $(\mathbf{w}_1, \ldots, \mathbf{w}_k, \xi + \epsilon)$ *is feasible.*

Clustering Accuracy Comparison

Data	KM	NC	MMC	CPM3C
Dig 0689	42.23	93.13	94.83	**96.63**
Dig 1279	40.42	90.11	91.91	**94.01**
Cora-DS	28.24	36.88	-	**43.75**
Cora-HA	34.02	42.00	-	**59.75**
Cora-ML	27.08	31.05	-	**45.58**
Cora-OS	23.87	23.03	-	**58.89**
Cora-PL	33.80	33.97	-	**46.83**
WK-CL	55.71	61.43	-	**71.95**
WK-TX	45.05	35.38	-	**69.29**
WK-WT	53.52	32.85	-	**77.96**
WK-WC	49.53	33.31	-	**73.88**
20-news	35.27	41.89	-	**70.63**
RCVI	27.05	-	-	**61.97**

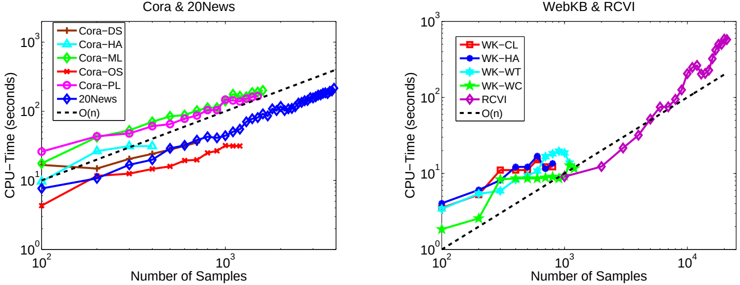
Rand Index Comparison

Data	KM	NC	MMC	CPM3C
Dig 0689	0.696	0.939	0.941	**0.968**
Dig 1279	0.681	0.909	0.913	**0.943**
Cora-DS	0.589	**0.744**	-	0.735
Cora-HA	0.385	0.659	-	**0.692**
Cora-ML	0.514	0.720	-	**0.754**
Cora-OS	0.518	0.522	-	**0.721**
Cora-PL	0.643	0.675	-	**0.703**
WK-CL	0.603	0.602	-	**0.728**
WK-TX	0.604	0.514	-	**0.707**
WK-WT	0.616	0.581	-	**0.747**
WK-WC	0.581	0.509	-	**0.752**
20-news	0.581	0.496	-	**0.782**
RCVI	0.471	-	-	**0.698**

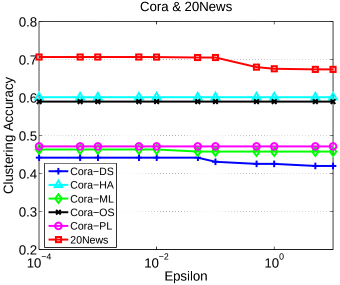
Speed Comparison

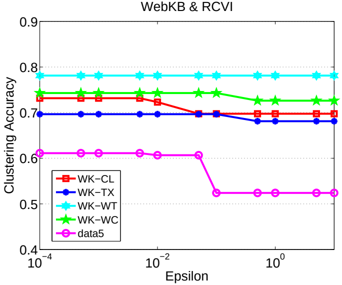
Data	KM	CPM3C
Dig 0689	34.28	9.66
Dig 1279	17.78	17.47
Cora-DS	839.67	35.31
Cora-HA	204.43	24.35
Cora-ML	22781	69.04
Cora-OS	47931	13.98
Cora-PL	7791.4	165.0
WK-CL	672.69	9.534
WK-TX	766.77	10.53
WK-WT	4135.2	10.67
WK-WC	1578.2	9.041
20-news	2387.8	215.6
RCVI	428770	587.9

Dataset Size *n* vs. Speed

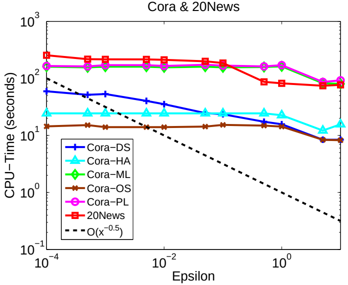


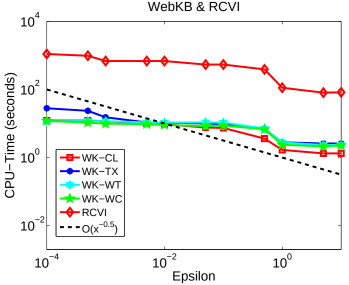
ϵ vs. Accuracy





ϵ vs. Speed





Outline

1. Motivation
2. Two-Class Maximum Margin Clustering
3. Multi-Class Maximum Margin Clustering
4. Related Works
5. Conclusions

Semi-Supervised Support Vector Machine

Given $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_l, \mathbf{x}_{l+1}, \cdots, \mathbf{x}_n\}$, where the first l points in $\mathcal{X}$ are labeled as $y_i \in \{-1, +1\}$ and the remaining $u = n - l$ points are unlabeled

$$\begin{aligned} \min_{y_{l+1},\ldots,y_n} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C_l}{n}\sum_{i=1}^{l}\xi_i + \frac{C_u}{n}\sum_{j=l+1}^{n}\xi_j \\ s.t. \quad & y_i[\mathbf{w}^T\phi(\mathbf{x}_i)+b] \geq 1-\xi_i,\ \forall i=1,\ldots,l \\ & y_j[\mathbf{w}^T\phi(\mathbf{x}_j)+b] \geq 1-\xi_j,\ \forall j=l+1,\ldots,n \\ & \xi_i \geq 0,\ \forall i=1,\ldots,l \\ & \xi_j \geq 0,\ \forall j=l+1,\ldots,n \end{aligned} \tag{20}$$

CutS3VM: Fast S3VM Algorithm

Theorem

Problem (20) is equivalent to

$$\min_{\mathbf{w},\xi_i} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}+\frac{C_l}{n}\sum_{i=1}^{l}\xi_i+\frac{C_u}{n}\sum_{j=l+1}^{n}\xi_j \tag{21}$$
$$\begin{aligned} s.t.\quad & y_i[\mathbf{w}^T\phi(\mathbf{x}_i)+b]\geq 1-\xi_i,\ \forall i=1,\dots,l \\ & |\mathbf{w}^T\phi(\mathbf{x}_j)+b|\geq 1-\xi_j,\ \forall j=l+1,\dots,n \\ & \xi_i\geq 0,\ \forall i=1,\dots,l \\ & \xi_j\geq 0,\ \forall j=l+1,\dots,n \end{aligned}$$

where the labels $y_j, j=l+1,\dots,n$ *are calculated as* $y_j=\mathrm{sign}(\mathbf{w}^T\phi(\mathbf{x}_j)+b)$.

CutS3VM: Fast S3VM Algorithm

Theorem

Problem (21) can be equivalently formulated as

$$\min_{\mathbf{w},\xi\geq 0} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}+\xi \tag{22}$$

$$s.t. \quad \frac{1}{n}\left\{C_l\sum_{i=1}^{l}c_iy_i[\mathbf{w}^T\phi(\mathbf{x}_i)+b]+C_u\sum_{j=l+1}^{n}c_j|\mathbf{w}^T\phi(\mathbf{x}_j)+b|\right\}$$

$$\geq \frac{1}{n}\left\{C_l\sum_{i=1}^{l}c_i+C_u\sum_{j=l+1}^{n}c_j\right\}-\xi, \quad \forall\, \mathbf{c}\in\{0,1\}^n$$

and any solution $\mathbf{w}^*$ to problem (22) is also a solution to problem (21) (vice versa), with $\xi^* = \frac{C_l}{n}\sum_{i=1}^{l}\xi_i^* + \frac{C_u}{n}\sum_{j=l+1}^{n}\xi_j^*$.

Maximum Margin Embedding

- Traditional embedding methods find the optimal subspace by minimizing some form of average loss or cost
- MME directly finds the most discriminative subspace, where clusters are most well-separated
- MME is insensitive to the actual probability distribution of patterns lying further away from the separating hyperplanes

Maximum Margin Embedding

$$\begin{aligned} \min_{\mathbf{y},\mathbf{w},b,\xi_i\geq 0} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ s.t. \quad & y_i(\mathbf{w}^T\phi(\mathbf{x}_i)+b) \geq 1-\xi_i,\ i=1,\ldots,n \\ & \mathbf{A}^T\mathbf{w} = 0 \\ & \mathbf{y} \in \{-1,+1\}^n \end{aligned} \tag{23}$$

where $A = [\mathbf{w}^1, \ldots, \mathbf{w}^{r-1}]$ constrains that $\mathbf{w}$ should be orthogonal to all previously calculated projecting vectors.

Outline

1 Motivation

2 Two-Class Maximum Margin Clustering

3 Multi-Class Maximum Margin Clustering

4 Related Works

5 Conclusions

Conclusions

Improvements

- No loss in clustering accuracy
- Major improvement on speed
- Handle large real-world datasets efficiently

Conclusions

Future works

- Automatically tune the parameters
- Even larger dataset

References

- Bin Zhao, Fei Wang, Changshui Zhang. Maximum Margin Embedding. ICDM 2008
- Bin Zhao, Fei Wang, Changshui Zhang. CutS3VM: A Fast Semi-Supervised SVM Algorithm. KDD 2008.
- Bin Zhao, Fei Wang, Changshui Zhang. Efficient Multiclass Maximum Margin Clustering. ICML 2008.
- Bin Zhao, Fei Wang, Changshui Zhang. Efficient Maximum Margin Clustering Via Cutting Plane Algorithm. SDM 2008

Thanks for Listening