Sharing Features in Multi-class Boosting via Group Sparsity

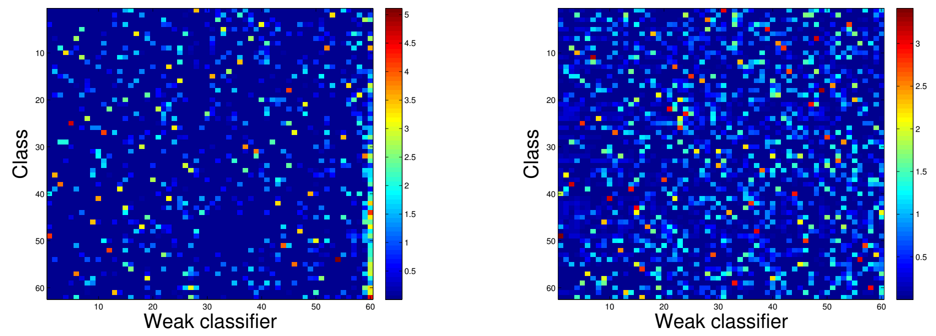
Sakrapee Paisitkriangkrai Chunhua Shen Anton van den Hengel

School of Computer Science, The University of Adelaide, Adelaide, SA 5005, Australia

THE UNIVERSITY of ADELAIDE

Introduction

- We propose a new formulation, termed MultiBoost$^{\text{group}}$, for multi-class boosting that promotes feature sharing across classes by enforcing group sparsity regularization.



(a) ℓ_1 (b) Group sparsity $\ell_{1,2}$

- Our derivation for designing fully corrective multi-class boosting methods is applicable to general $\ell_{p,q}$ $(p, q \geq 1)$ mixed-norms.
- We also propose the use of the alternating direction method of multipliers (ADMM) [1] to efficiently solve the involved optimization problems, which is much faster than using standard interior-point solvers.
- We empirically show that sharing features across classes can further improve classification performance and efficiency.

Multi-class boosting

- Let $F_{y_i}(\cdot)$ be the response of the linear classifier corresponding to y_i (the true class label) when applied to training instance x_i.
- The multi-class margins for the instance x_i thus can be defined as:

$$F_{y_i}(x_i) - F_r(x_i), \forall r \neq y_i.$$

- We want to maximize this term in the framework of large-margin learning. By employing hinge loss with $\ell_{1,2}$ mixed-norm regularization, the optimization problem can be written as,

$$\min_{W,\xi} \sum_{i=1}^{m} \xi_i + \nu \|W\|_{1,2} \text{ s.t. } \delta_{r,y_i} + H_{i:}\boldsymbol{w}_{y_i} \geq 1 + H_{i:}\boldsymbol{w}_r - \xi_i, \forall i, r;\ W \geq 0; \boldsymbol{\xi} \geq 0.$$

where ν is the regularization parameter. $\delta_{s,t}$ is an indication operator ($\delta_{s,t} = 1$ if $s = t$ and $\delta_{s,t} = 0$, otherwise). The matrix $H \in \mathbb{Z}^{m \times n}$, is made up of the binary outputs of the weak classifiers. $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_k$ are the coefficients of the linear classifier. We define the matrix $W = [\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_k] \in \mathbb{R}^{n \times k}$ such that each column of W, $\boldsymbol{w}_r$, contains coefficients of the linear classifier for class r and each row of W, $W_{j:}$, consists of the coefficients for the weak classifier $h_j(\cdot)$ for all class labels.

- The Lagrange dual problem can be written as,

$$\min_{U,Q} \sum_{i,r} U_{ir}\delta_{r,y_i} \tag{1}$$
$$\text{s.t. } \sum_i (\delta_{r,y_i} - U_{ir}) H_{i:} \leq \nu Q_{:r}, \forall r;\ \sum_r U_{ir} = 1, \forall i;\ U \geq 0; \quad \|Q_{j:}\|_2 \leq 1, \forall j.$$

Since there can be infinitely many constraints, we need to use column generation to solve (1) [2, 3].

- The subproblem for generating weak classifiers is

$$h^*(\cdot) = \operatorname*{argmax}_{h(\cdot) \in \mathcal{H}, r} \sum_{i=1}^{m} (\delta_{r,y_i} - U_{ir}) h(x_i).$$

where $h^*(\cdot)$ is the one that most violates the first constraint in the dual.

Logistic loss and faster training of multi-class boosting

- We can also design a boosting algorithm for optimizing the logistic loss. The learning problem can be expressed as:

$$\min_{W,V,\rho} \sum_{i,r} \log(1 + \exp(-\rho_{ir})) + \nu \|W\|_{1,2} \text{ s.t. } \rho_{ir} = H_{i:}\boldsymbol{w}_{y_i} - H_{i:}\boldsymbol{w}_r, \forall i, \forall r,\ W \geq 0.$$

- The Lagrange dual problem is

$$\max_{U,Q} \ -\sum_{i,r} \left[U_{ir} \log(U_{ir}) + (1 - U_{ir}) \log(1 - U_{ir}) \right] \tag{2}$$
$$\text{s.t. } \sum_i \left[\delta_{r,y_i} \left(\sum_l U_{il} \right) - U_{ir} \right] H_{i:} \leq \nu Q_{:r}, \forall r;\ \|Q_{j:}\|_2 \leq 1, \forall j.$$

- Since real-world data consists of a large number of samples and classes, we want to speed up the training time. We achieve this by simplify the margin, $\rho_{i,r}$, as $y_{ir} H_{i:} \boldsymbol{w}_r$ where $y_{ir} = 1$ if $y_i = r$ and $y_{ir} = -1$, otherwise.
- Each $\mathbf{w}_r$ can now be solved independently with the use of ADMM.

MultiBoost with shared weak classifiers via group sparsity

Input:
1) A set of examples $\{\boldsymbol{x}_i, y_i\}, i = 1 \cdots m$;
2) The maximum number of weak classifiers, T;
Output: A multi-class classifier $F(\boldsymbol{x}) = \operatorname*{argmax}_r \sum_{j=1}^{T} W_{jr} h_j(\boldsymbol{x})$;

Initilaize:
1) $t \leftarrow 0$;
2) Initialize sample weights, $U_{ir} = 1/(mk)$;

while $t < T$ **do**
  1) Train a weak learner, $h_t(\cdot) =$

$$\begin{cases} \operatorname*{argmax}_{h(\cdot), r} \sum_{i=1}^{m} \left[\delta_{r,y_i} - U_{ir} \right] h(\boldsymbol{x}_i), & \text{hinge loss} \\ \operatorname*{argmax}_{h(\cdot), r} \sum_{i=1}^{m} \left[\delta_{r,y_i} \left(\sum_l U_{il} \right) - U_{ir} \right] h(\boldsymbol{x}_i), & \text{logistic} \end{cases}$$

  $\forall r, \forall h(\cdot) \in \mathcal{H}$;
  2) If the stopping criterion has been met, we exit the loop.
  if $\left\| \sum_{i=1}^{m} \left[\delta_{r,y_i} - U_{ir} \right] h(\boldsymbol{x}_i) \right\|_2 < \nu + \epsilon$ **then**
    break; (hinge loss)
  if $\left\| \sum_{i=1}^{m} \left[\delta_{r,y_i} \left(\sum_l U_{il} \right) - U_{ir} \right] h(\boldsymbol{x}_i) \right\|_2 < \nu + \epsilon$ **then**
    break; (logistic loss)
  3) Add the best weak learner, $h_t(\cdot)$, into the current set;
  4) Solve the objective problem

Hinge loss:
$$\begin{aligned} \min_{W,V,\boldsymbol{\xi}} \quad & \sum_{i=1}^{m} \xi_i + \nu \|V\|_{1,2} \\ \text{s.t.} \quad & \delta_{r,y_i} + H_{i:}\boldsymbol{w}_{y_i} \geq 1 + H_{i:}\boldsymbol{w}_r - \xi_i, \forall i, r \\ & V = W; W \geq 0; \boldsymbol{\xi} \geq 0. \end{aligned}$$

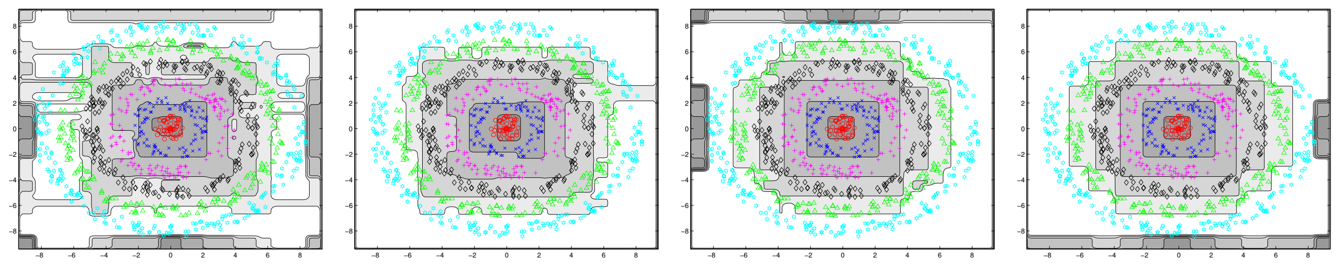
Logistic loss:
$$\begin{aligned} \min_{W,V,\boldsymbol{\rho}} \quad & \sum_{i=1}^{m} \sum_{r=1}^{k} \log(1 + \exp(-\rho_{ir})) + \nu \|V\|_{1,2} \\ \text{s.t.} \quad & \rho_{ir} = H_{i:}\boldsymbol{w}_{y_i} - H_{i:}\boldsymbol{w}_r, \forall i, \forall r \\ & V = W; W \geq 0. \end{aligned}$$

  5) Update sample weights (dual variables);
  6) $t \leftarrow t + 1$;

Experiments

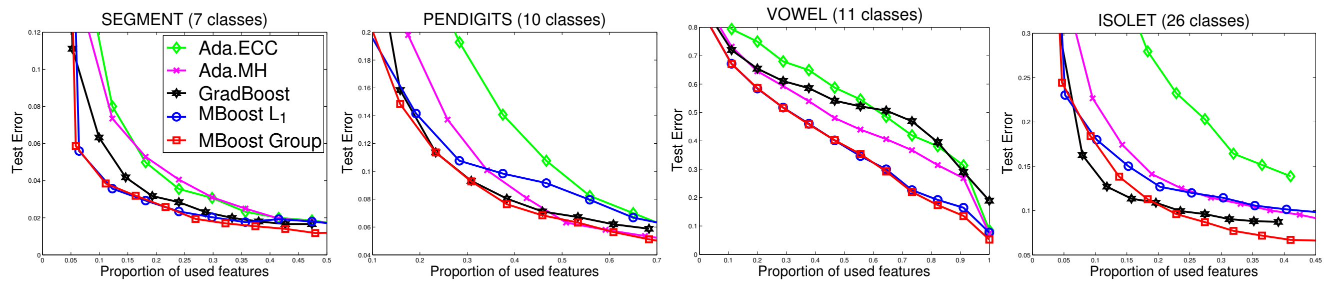
We compare the performance between MultiBoost$^{\text{group}}$ with AdaBoost.MH, AdaBoost.ECC, GradBoost, AdaBoost.SIP and MultiBoost$^{\ell_1}$.

Artificial data



(a) AdaBoost.ECC (b) AdaBoost.MH (c) MultiBoost (d) MultiBoost$^{\text{group}}$

UCI data sets



Distribution of shared weak classifiers on handwritten data sets

ABCDETC	'0 – 15'	'16 – 30'	'31 – 45'	'46 – 62'
MultiBoost (Shen and Hao)	99.8%	0.2%	0%	0%
MultiBoost-Group (ours)	0%	81.3%	18.7%	0%
MultiBoost-FAST	0%	65.7%	33.5%	0.7%

Table: The distribution of shared weak classifiers. For example, '31 – 45' indicates that the weak classifier is being shared among 31 to 45 classes.

Scene recognition

methods	# features used	accuracy (%)
SAMME (Zhu *et al.*)	1000	70.9 (0.40)
JointBoost (Torralba *et al.*)	1000	72.2 (0.70)
MultiBoost (Shen and Hao)	1000	76.0 (0.48)
AdaBoost.SIP (Zhang *et al.*)	1000	75.7 (0.10)
AdaBoost.ECC (Guruswami and Sahai)	1000	76.5 (0.67)
AdaBoost.MH (Schapire and Singer)	1000	77.6 (0.59)
MultiBoost-Group (ours)	1000	77.8 (0.77)
MultiBoost-FAST (ours)	**1000**	**79.2** (0.82)
Linear SVM	6200	76.3 (0.88)
Nonlinear SVM (HIK)	**6200**	**81.4** (0.60)

Table: Recognition rate on Scene15 data sets. All experiments are run 5 times. The average accuracy mean and standard deviation (in percentage) are reported.

References

[1] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations & Trends in Mach. Learn.*, 3(1), 2011.

[2] A. Demiriz, K. P. Bennett, and J. Shawe-Taylor. Linear programming boosting via column generation. *Mach. Learn.*, 46(1-3):225–254, 2002.

[3] Chunhua Shen and Hanxi Li. On the dual formulation of boosting algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010.