AdaMKL: A Novel Biconvex Multiple Kernel Learning Approach

Ziming Zhang, Ze-Nian Li, Mark Drew
School of Computing Science, Simon Fraser University, Vancouver, B.C., Canada
{zxa27, li, mark}@cs.sfu.ca
Outline

- Background
- Adaptive Multiple Kernel Learning
- Experiments
- Conclusion
Background

- Multiple Kernel Learning
  - Aim to learn kernel coefficients and support vectors together

\[
\{K_{1\ldots M}\} \begin{cases} 
\forall i, & \gamma_i \geq 0 \\
\|\gamma\|_p = 1, & p = 1, 2, L
\end{cases} 
\]

\[
K_{opt} = \sum_{m=1}^{M} \gamma_m K_m
\]
Background

- Example: $L_p$-norm Multiple Kernel Learning [1]

\[
\begin{align*}
\min_{\gamma, \mathbf{w}, b, \xi} & \quad \frac{1}{2} \sum_{m} \frac{\| \mathbf{w}_m \|^2}{\gamma_m} + C \sum \xi_i \\
\text{s.t.} & \quad \forall i, y_i \left[ \sum_{m} \langle \mathbf{w}_m, \Phi_m(x_i) \rangle + b \right] \geq 1 - \xi_i \\
& \quad \xi_i \geq 0, C \geq 0 \\
& \quad \forall m, \gamma_m \geq 0, \| \mathbf{w}_m \|_p \leq 1
\end{align*}
\]

Adaptive Multiple Kernel Learning

\[ \begin{bmatrix} \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_M(x) \end{bmatrix} \]

weighting

\[ \begin{bmatrix} \theta_1 \cdot \phi_1(x) \\ \theta_2 \cdot \phi_2(x) \\ \vdots \\ \theta_M \cdot \phi_M(x) \end{bmatrix} \]

\[ f(x) = \sum_{m=1}^{M} \theta_m \langle w, \phi_m(x) \rangle + b \]
Adaptive Multiple Kernel Learning

- Biconvex functions
  - $f(x,y)$ is a **biconvex function** if $f_y(x)$ is convex and $f_x(y)$ is convex.
  - Example: $f(x,y) = x^2 + y^2 - 3xy$

- Biconvex optimization
  - At least one function in the objective functions and constraints is biconvex.
  - Local optima
Adaptive Multiple Kernel Learning

- Adaptive Multiple Kernel Learning (AdaMKL)
  - Aim to simplify the MKL learning process as well as keep the similar discriminative power of MKL using biconvex optimization.
  - Binary classification
    \[
    \min_{\theta, w, b, \xi} \quad \frac{1}{2} N_0(\theta) \|w\|_2^2 + C \sum_i \xi_i
    \]
    \[
    \text{s.t.} \quad \forall i, y_i \left[ \sum_m \theta_m \langle w_m, \phi_m(x_i) \rangle + b \right] \geq 1 - \xi_i
    \]
    \[
    \xi_i \geq 0, C \geq 0
    \]
    where \( N_0(\theta) = \|\theta\|_1^2, N_p(\theta) = \|\theta^2\|_{p \geq 1} \)

- Objective function:
  \[
  \sum_m \theta_m^2 \|w_m\|_2^2 \leq \|\theta^2\|_p \|w\|_2^2 \leq \|\theta^2\|_1 \|w\|_2^2 \leq \|\theta\|_1 \|w\|_2^2
  \]
Adaptive Multiple Kernel Learning

- Optimization
  - Learn $w$ by fixing $\theta$ using $N_0(\theta)$ norm
  - Learn $\theta$ by fixing $w$ using $L_1$ or $L_2$ norm of $\theta$
  - Repeat the two steps until converged

- Kernel coefficient constraints

$$\begin{align*}
\max_{\alpha} & \quad \sum_{i} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \left[ \sum_m \frac{\theta^2_m}{N_p(\theta)} K_m(x_i, x_j) \right] \\
\text{s.t.} & \quad \forall i, 0 \leq \alpha_i \leq C, \sum_i \alpha_i y_i = 0
\end{align*}$$

(w-Dual)

\[
\gamma_m = \frac{\theta^2_m}{N_p(\theta)} \geq 0, \quad \|\gamma\|_p = 1
\]
Adaptive Multiple Kernel Learning

- **Complexity**
  - Same as quadratic programming

- **Convergence**
  - If hard-margin cases \((C = +1)\) can be solved at the initialization stage, then AdaMKL will converge to a local minimum.
  - If at either step our objective function converged, then AdaMKL has converged to a local minimum.
Adaptive Multiple Kernel Learning

**L_p-norm MKL**

\[
\min_{\theta,w,b,\xi} \frac{1}{2} \sum_{m} \left\| w_m \right\|_{2}^{2} + C \sum_{i} \xi_i
\]

s.t.
\[
\forall i, y_i \left[ \sum_{m} \langle w_m, \phi_m(x_i) \rangle + b \right] \geq 1 - \xi_i
\]

\[
\theta_m \geq 0, \left\| \theta \right\|_p \leq 1
\]

\[
\xi_i \geq 0, C \geq 0
\]

- Convex
- Kernel coefficient norm condition
- Gradient search, Semi-infinite programming (SIP), etc

**AdaMKL**

\[
\min_{\theta,w,b,\xi} \frac{1}{2} N_p(\theta) \left\| w \right\|_{2}^{2} + C \sum_{i} \xi_i
\]

s.t.
\[
\forall i, y_i \left[ \sum_{m} \theta_m \langle w_m, \phi_m(x_i) \rangle + b \right] \geq 1 - \xi_i
\]

\[
\xi_i \geq 0, C \geq 0
\]

\[
\max_{\alpha} \sum_{i} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \left[ \sum_{m} \frac{\theta_m^2}{N_p(\theta)} K_m(x_i, x_j) \right]
\]

s.t.
\[
\forall i, 0 \leq \alpha_i \leq C, \sum_{i} \alpha_i y_i = 0
\]

- Biconvex
- Kernel coefficient conditions hidden in dual
- Quadratic programming
Experiments

- 4 specific AdaMKL: $N_0 L_1$, $N_1 L_1$, $N_1 L_2$, $N_2 L_2$, where “N” and “L” denote the types of norm used for learning $w$ and $\Theta$.
- 2 experiments
  - Toy example: $C = 10^5$ without tuning, 10 Gaussian kernels, randomly sampled from 2D Gaussian distributions
    - Positive samples: mean [0 0], covariance [0.3 0; 0 0.3], 100 samples
    - Negative samples: mean [-1 -1] and [1 1], covariance [0.1 0; 0 0.1] and [0.2 0; 0 0.2], 100 samples, respectively.
  - 4 benchmark datasets: breast-cancer, heart, thyroid, and titanic (downloaded from http://ida.first.fraunhofer.de/projects/bench/)
    - Gaussian kernels + polynomial kernels
    - 100, 140, 60, 40 kernels for corresponding datasets, respectively
Experiments - Toy example
Experiments - Toy example
Experiments - Benchmark datasets

(a) Breast-Cancer: [69.64 ~ 75.23]
(b) Heart: [79.71 ~ 84.05]
(c) Thyroid: [95.20 ~ 95.80]
(d) Titanic: [76.02 ~ 77.58]
Experiments - Benchmark datasets

(a) Breast-Cancer

(b) Heart

(c) Thyroid

(d) Titanic
Conclusion

- Biconvex optimization for MKL
  - Arbitrary $L_p$ norm of kernel coefficient constraint, which is actually hidden in the dual without consideration explicitly.
  - Easy to optimize, fast to converge, lower computational time but similar performance as traditional convex optimization based MKL
Thank you !!!