Lecture 9: Multi Kernel SVM

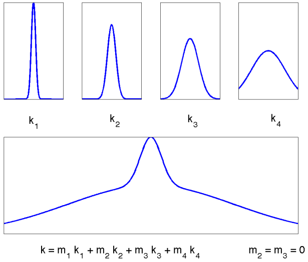
Stéphane Canu
stephane.canu@litislab.eu

Sao Paulo 2014

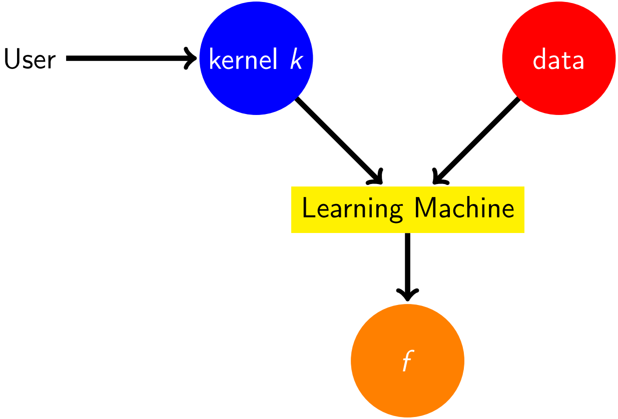
April 16, 2014

Roadmap

1. Tuning the kernel: MKL
   - The multiple kernel problem
   - Sparse kernel machines for regression: SVR
   - SimpleMKL: the multiple kernel solution

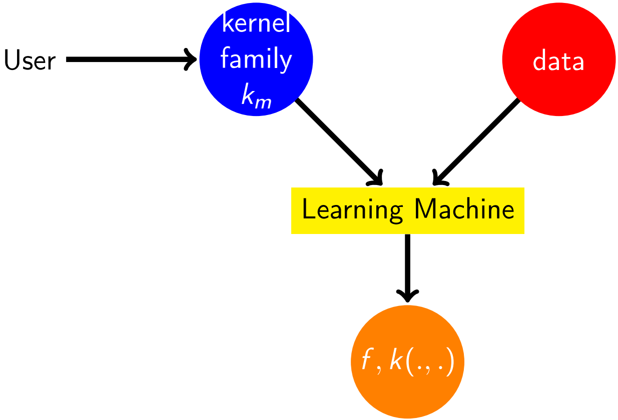


Standard Learning with Kernels



http://www.cs.nyu.edu/~mohri/icml2011-tutorial/tutorial-icml2011-2.pdf

Learning Kernel framework



http://www.cs.nyu.edu/~mohri/icml2011-tutorial/tutorial-icml2011-2.pdf

from SVM

- SVM: single kernel k

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i \quad k \; (\mathbf{x}, \mathbf{x}_i) + b$$

$$=$$

http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf

from SVM → to Multiple Kernel Learning (MKL)

- SVM: single kernel k
- MKL: set of M kernels $k_1, \ldots, k_m, \ldots, k_M$
  - learn classier and combination weights
  - can be cast as a convex optimization problem

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i \sum_{m=1}^{M} d_m \, k_m(\mathbf{x}, \mathbf{x}_i) + b \qquad \sum_{m=1}^{M} d_m = 1 \text{ and } 0 \leq d_m$$

$$=$$

http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf

from SVM → to Multiple Kernel Learning (MKL)

- SVM: single kernel k
- MKL: set of M kernels $k_1, \dots, k_m, \dots, k_M$
  - learn classier and combination weights
  - can be cast as a convex optimization problem

$$\begin{aligned} f(\mathbf{x}) &= \sum_{i=1}^{n} \alpha_i \sum_{m=1}^{M} d_m \; k_m(\mathbf{x}, \mathbf{x}_i) + b \qquad \sum_{m=1}^{M} d_m = 1 \text{ and } 0 \leq d_m \\ &= \sum_{i=1}^{n} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b \qquad \text{with} \quad K(\mathbf{x}, \mathbf{x}_i) = \sum_{m=1}^{M} d_m \; k_m(\mathbf{x}, \mathbf{x}_i) \end{aligned}$$

http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf

Multiple Kernel

The model

$$f(x) = \sum_{i=1}^{n} \alpha_i \sum_{m=1}^{M} d_m k_m(x, x_i) + b, \qquad \sum_{m=1}^{M} d_m = 1 \text{ and } 0 \leq d_m$$

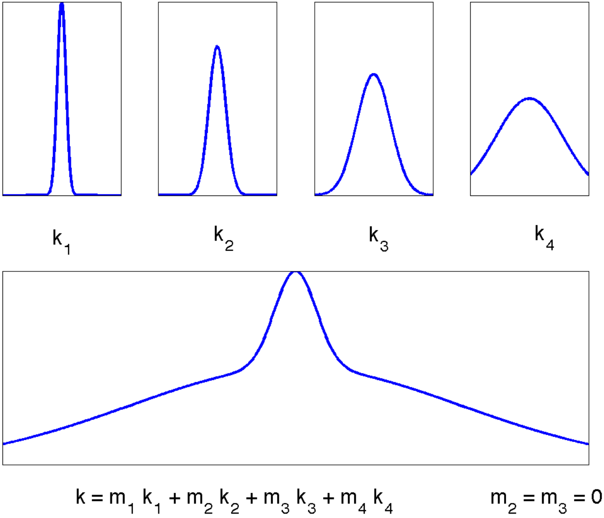
Given M kernel functions $k_1, \ldots, k_M$ that are potentially well suited for a given problem, find a positive linear combination of these kernels such that the resulting kernel k is "optimal"

$$k(\mathbf{x}, \mathbf{x}') = \sum_{m=1}^{M} d_m k_m(\mathbf{x}, \mathbf{x}'), \text{ with } d_m \geq 0, \sum_m d_m = 1$$

Learning together

The kernel coefficients d_m and the SVM parameters α_i, b.

Multiple Kernel: illustration



Multiple Kernel Strategies

- Wrapper method (Weston et al., 2000; Chapelle et al., 2002)
  - solve SVM
  - gradient descent on d_m on criterion:
    - margin criterion
    - span criterion
- Kernel Learning & Feature Selection
  - use Kernels as dictionary
- Embedded Multi Kernel Learning (MKL)

Multiple Kernel functional Learning

The problem (for given C)

$$\begin{aligned} \min_{f\in\mathcal{H},b,\xi,d} \quad & \frac{1}{2}\|f\|_{\mathcal{H}}^2 + C\sum_i \xi_i \\ \text{with} \quad & y_i\big(f(x_i)+b\big) \geq 1+\xi_i \;; \quad \xi_i \geq 0 \quad \forall i \\ & \sum_{m=1}^{M} d_m = 1 \;, \quad d_m \geq 0 \quad \forall m \;, \end{aligned}$$

$$f = \sum_m f_m \qquad \text{and} \qquad k(\mathbf{x},\mathbf{x}') = \sum_{m=1}^{M} d_m k_m(\mathbf{x},\mathbf{x}'), \text{ with } d_m \geq 0$$

The functional framework

$$\mathcal{H} = \bigoplus_{m=1}^{M} \mathcal{H}'_m \qquad \langle f,g\rangle_{\mathcal{H}'_m} = \frac{1}{d_m}\langle f,g\rangle_{\mathcal{H}_m}$$

Multiple Kernel functional Learning

The problem (for given C)

$$\begin{array}{ll}\min\limits_{\{f_m\},b,\xi,d} & \frac{1}{2}\sum\limits_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum\limits_i \xi_i \\ \text{with} & y_i\big(\sum\limits_m f_m(x_i) + b\big) \geq 1 + \xi_i \; ; \quad \xi_i \geq 0 \quad \forall i \\ & \sum\limits_m d_m = 1 \; , \quad d_m \geq 0 \quad \forall m \; ,\end{array}$$

Treated as a bi-level optimization task

$$\begin{array}{ll}\min\limits_{d\in\mathbb{R}^M} & \left\{\begin{array}{ll}\min\limits_{\{f_m\},b,\xi} & \frac{1}{2}\sum\limits_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum\limits_i \xi_i \\ \text{with} & y_i\big(\sum\limits_m f_m(x_i) + b\big) \geq 1 + \xi_i \; ; \quad \xi_i \geq 0 \quad \forall i\end{array}\right. \\ \text{s.t.} & \sum\limits_m d_m = 1 \; , \quad d_m \geq 0 \quad \forall m \; ,\end{array}$$

Multiple Kernel representer theorem and dual

The Lagrangian:

$$\mathcal{L} = \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i - \sum_i \alpha_i\Big(y_i\big(\sum_m f_m(x_i) + b\big) - 1 - \xi_i\Big) - \sum_i \beta_i \xi_i$$

Associated KKT stationarity conditions:

$$\nabla_m \mathcal{L} = 0 \quad \Leftrightarrow \quad \frac{1}{d_m} f_m(\bullet) = \sum_{i=1}^n \alpha_i y_i k_m(\bullet, \mathbf{x}_i) \qquad m = 1, M$$

Representer theorem

$$f(\bullet) = \sum_m f_m(\bullet) = \sum_{i=1}^n \alpha_i y_i \underbrace{\sum_m d_m k_m(\bullet, \mathbf{x}_i)}_{K(\bullet, \mathbf{x}_i)}$$

We have a standard SVM problem with respect to function f and kernel K.

Multiple Kernel Algorithm

Use a Reduced Gradient Algorithm[1]

$$\begin{aligned} \min_{d \in \mathbb{R}^M} \quad & J(d) \\ \text{s.t.} \quad & \sum_m d_m = 1 \;, \quad d_m \geq 0 \quad \forall m \;, \end{aligned}$$

SimpleMKL algorithm

set $d_m = \frac{1}{M}$ for $m = 1, \dots, M$
while stopping criterion not met **do**
  compute $J(d)$ using an QP solver with $K = \sum_m d_m K_m$
  compute $\frac{\partial J}{\partial d_m}$, and projected gradient as a descent direction D
  $\gamma \leftarrow$ compute optimal stepsize
  $d \leftarrow d + \gamma D$
end while

$\longrightarrow$ Improvement reported using the Hessian

[1] Rakotomamonjy et al. JMLR 08

Computing the reduced gradient

At the optimal the primal cost = dual cost

$$\underbrace{\frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i}_{\text{primal cost}} = \underbrace{\frac{1}{2}\alpha^\top G\alpha - \mathbf{e}^\top\alpha}_{\text{dual cost}}$$

with $G = \sum_m d_m G_m$ where $G_{m,ij} = k_m(\mathbf{x}_i, \mathbf{x}_j)$

Dual cost is easier for the gradient

$$\nabla_{d_m} J(\mathbf{d}) = \frac{1}{2}\alpha^\top G_m \alpha$$

Reduce (or project) to check the constraints $\sum_m d_m = 1 \;\rightarrow\; \sum_m D_m = 0$

$$D_m = \nabla_{d_m} J(\mathbf{d}) - \nabla_{d_1} J(\mathbf{d}) \qquad \text{and} \qquad D_1 = -\sum_{m=2}^{M} D_m$$

Complexity

For each iteration:

- SVM training: $O(nn_{sv} + n_{sv}^3)$.
- Inverting $K_{sv,sv}$ is $O(n_{sv}^3)$, but might already be available as a by-product of the SVM training.
- Computing H: $O(Mn_{sv}^2)$
- Finding d: $O(M^3)$.

The number of iterations is usually less than 10.

$\longrightarrow$ When $M < n_{sv}$, computing d is not more expensive than QP.

MKL on the 101-caltech dataset

Performance of recent methods applied to Caltech-101. Note that (*) combines [Gehler et al. ICCV'09] and our features.

Method	15 train	30 train
LP-beta(*) P. Gehler and S. Nowozin, ICCV'09.	74.6 ± 1.0	82.1 ± 0.3
Group-sensitive multiple kernel learning for object categorization. J. Yang, Y. Li, Y. Tian, L. Duan, and W. In Proc. ICCV, 2009.	73.2	84.3
Bayesian localized multiple kernel learning. M. Christoudias, R. Urtasun, and T. Darrell. *Technical report, UC Berkeley,* 2009.	73.0 ± 1.3	NA
In defense of nearest-neighbor based image classification. O. Boiman, E. Shechtman, and M. Irani. In *Proc. CVPR*, 2008.	72.8	≈79
This method.	71.1 ± 0.6	78.2 ± 0.4
On feature combination for multiclass object classification. P. Gehler and S. Nowozin. In *Proc. ICCV*, 2009.	70.4 ± 0.8	77.7 ± 0.3
Recognition using regions. C. Gu, J. J. Lim, P. Arbelàez, and J. Malik. In *Proc. CVPR*, 2009.	65.0	73.1
SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. H. Zhang, A. C. Berg, M. Maire, and J. Malik. In *Proc. CVPR*, 2006.	59.06 ± 0.56	66.23 ± 0.48

http://www.robots.ox.ac.uk/~vgg/software/MKL/

Support vector regression (SVR)

the t-insensitive loss

$$\begin{cases} \min\limits_{f \in \mathcal{H}} & \frac{1}{2}\|f\|^2_{\mathcal{H}} \\ \text{with} & |f(\mathbf{x}_i) - y_i| \leq t, \; i = 1, n \end{cases}$$

The support vector regression introduce slack variables

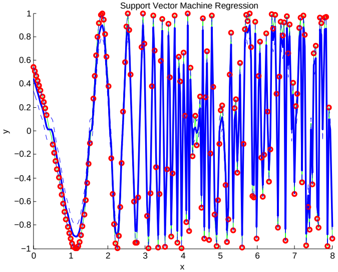
$$(SVR) \quad \begin{cases} \min\limits_{f \in \mathcal{H}} & \frac{1}{2}\|f\|^2_{\mathcal{H}} + C \sum |\xi_i| \\ \text{with} & |f(\mathbf{x}_i) - y_i| \leq t + \xi_i \quad 0 \leq \xi_i \quad i = 1, n \end{cases}$$

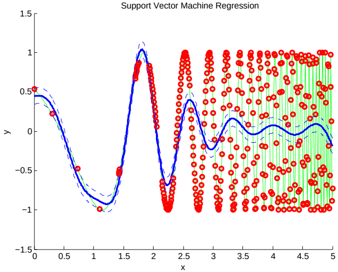
- a typical multi parametric quadratic program (mpQP)
- piecewise linear regularization path

$$\alpha(C, t) = \alpha(C_0, t_0) + (\frac{1}{C} - \frac{1}{C_0})\mathbf{u} + \frac{1}{C_0}(t - t_0)\mathbf{v}$$

- 2d Pareto's front (the tube width and the regularity)

Support vector regression illustration





C large $\qquad$ C small

- there exists other formulations such as LP SVR...

Multiple Kernel Learning for regression

The problem (for given C and t)

$$\begin{array}{ll}\min_{\{f_m\},b,\xi,d} & \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i \\ \text{s.t.} & \left|\sum_m f_m(x_i) + b - y_i\right| \le t + \xi_i \quad \forall i \xi_i \ge 0 \quad \forall i \\ & \sum_m d_m = 1 \;, \quad d_m \ge 0 \quad \forall m \;,\end{array}$$

regularization formulation

$$\begin{array}{ll}\min_{\{f_m\},b,d} & \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \max\left(\left|\sum_m f_m(x_i) + b - y_i\right| - t, 0\right) \\ & \sum_m d_m = 1 \;, \quad d_m \ge 0 \quad \forall m \;,\end{array}$$

Equivalently

$$\min_{\{f_m\},b,\xi,d} \quad \sum_i \max\Big(\big|\sum_m f_m(x_i) + b - y_i\big| - t, 0\Big) + \frac{1}{2C}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + \mu\sum_m |d_m|$$

Multiple Kernel functional Learning

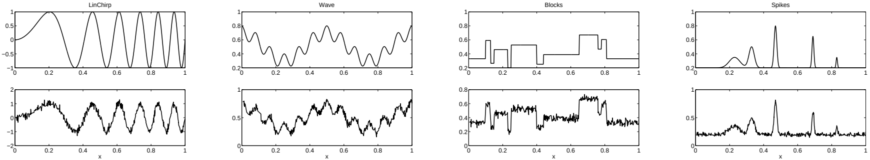
The problem (for given C and t)

$$\begin{aligned} \min_{\{f_m\},b,\xi,d} \quad & \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i \\ \text{s.t.} \quad & \Big|\sum_m f_m(x_i) + b - y_i\Big| \leq t + \xi_i \quad \forall i \xi_i \geq 0 \quad \forall i \\ & \sum_m d_m = 1\,, \quad d_m \geq 0 \quad \forall m \,, \end{aligned}$$

Treated as a bi-level optimization task

$$\begin{aligned} \min_{d \in \mathbb{R}^M} \quad & \left\{ \begin{aligned} \min_{\{f_m\},b,\xi} \quad & \frac{1}{2}\sum_m \frac{1}{d_m}\|f_m\|^2_{\mathcal{H}_m} + C\sum_i \xi_i \\ \text{s.t.} \quad & \Big|\sum_m f_m(x_i) + b - y_i\Big| \geq t + \xi_i \quad \forall i \\ & \xi_i \geq 0 \quad \forall i \end{aligned} \right. \\ \text{s.t.} \quad & \sum_m d_m = 1\,, \quad d_m \geq 0 \quad \forall m \,, \end{aligned}$$

Multiple Kernel experiments



	Single Kernel	Kernel *Dil*		Kernel *Dil-Trans*	
Data Set	Norm. MSE (%)	#Kernel	Norm. MSE	#Kernel	Norm. MSE
LinChirp	1.46 ± 0.28	7.0	1.00 ± 0.15	21.5	$\mathbf{0.92} \pm \mathbf{0.20}$
Wave	0.98 ± 0.06	5.5	$\mathbf{0.73} \pm \mathbf{0.10}$	20.6	0.79 ± 0.07
Blocks	1.96 ± 0.14	6.0	2.11 ± 0.12	19.4	$\mathbf{1.94} \pm \mathbf{0.13}$
Spike	6.85 ± 0.68	6.1	6.97 ± 0.84	12.8	$\mathbf{5.58} \pm \mathbf{0.84}$

Table: Normalized Mean Square error averaged over 20 runs.

Conclusion on multiple kernel (MKL)

- MKL: Kernel tuning, variable selection...
  - extention to classification and one class SVM
- SVM KM: an efficient Matlab toolbox (available at MLOSS)[2]
- Multiple Kernels for Image Classification: Software and Experiments on Caltech-101[3]
- new trend: Multi kernel, Multi task and ∞ number of kernels

[2] http://mloss.org/software/view/33/

[3] http://www.robots.ox.ac.uk/~vgg/software/MKL/

Bibliography

- A. Rakotomamonjy, F. Bach, S. Canu & Y. Grandvalet. SimpleMKL. J. Mach. Learn. Res. 2008, 9:2491–2521.
- M. Gönen & E. Alpaydin Multiple kernel learning algorithms. J. Mach. Learn. Res. 2008;12:2211-2268.

- http://www.cs.nyu.edu/~mohri/icml2011-tutorial/tutorial-icml2011-2.pdf
- http://www.robots.ox.ac.uk/~vgg/software/MKL/
- http://www.nowozin.net/sebastian/talks/ICCV-2009-LPbeta.pdf