Matching Pursuit Kernel Fisher Discriminant Analysis

Tom Diethe, Zakria Hussain, David Roi Hardoon, John Shawe-Taylor

Department of Computer Science, University College London

Sparsity in Machine Learning and Statistics
2^{nd} April 2009

Introduction

- Algorithm motivated by methods from signal processing and ML
- Greedy alternative to L_1 optimisation (e.g. Lasso) to achieve sparsity
- Generalisation error bound (sample compression + Rademacher complexity)
- Competitive with state-of-the-art techniques

Matching Pursuit

- Proposed in the signal processing literature [Mallat and Zhang, 1993]
- Decompose signal into sparse set of basis functions (atoms) from a given dictionary
- In Orthogonal Matching Pursuit (OMP) each time an atom is chosen the remaining weight vectors are projected into an orthogonal space
- Kernel Matching Pursuit (KMP) [Vincent and Bengio, 2002] has been proposed as kernel counterpart of MP/OMP

Notation

$\mathbf{x} \in \mathbb{R}^n$	Examples
$y \in \{-1, 1\}$	Labels
$\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_m)'$	inputs as row vectors
$\mathbf{w}$	Primal weight vector
$\mathbf{e}$	Unit vector
$\mathbf{K}$	Kernel matrix has entries $\mathbf{K}[i,j] = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$
$\phi(\mathbf{x})$	Feature map
$\mathbf{K}[:, i]$	ith column of $\mathbf{K}$
$\mathbf{i} = \{i_1, \ldots, i_k\}$	set of indices
$\mathbf{K}[\mathbf{i}, \mathbf{i}]$	square matrix defined by index set $\mathbf{i}$

Framework for Machine Learning

- Generic (Orthogonal) Matching Pursuit framework for ML described by [Hussain and Shawe-Taylor, 2008]
- Previously applied to KPCA and KCCA

Generic matching pursuit algorithm

Input: data $\mathbf{X}$, sparsity parameter $k > 0$

1. initialise $\mathbf{i} = ()$
2. **for** $i = 1$ to k **do**
3. set $\mathbf{i}_i$ to index of maximiser of a loss function that uses $\mathbf{X}$
4. deflate vectors in $\mathbf{X}$ according to basis vector chosen
5. **end for**
6. use final $\mathbf{i}$ to construct subspace and carry out learning in this low dimensional subspace to find (sparse) parameters

Output: final set $\mathbf{i}$ and (sparse) parameters to make predictions, create low dimensional projection, etc.

MPKFDA - Derivation I

- Using the notation from [Shawe-Taylor and Cristianini, 2004], we have the following maximisation problem for FDA:

$$\hat{\mathbf{w}} = \max_{\mathbf{w}} \frac{\mathbf{w}'\mathbf{X}'\mathbf{y}\mathbf{y}'\mathbf{X}\mathbf{w}}{\mathbf{w}'\mathbf{X}'\mathbf{B}\mathbf{X}\mathbf{w}} \tag{1}$$

- Apply the Nyström method of low-rank approximation of the Gram matrix [Williams and Seeger, 2001]

$$\begin{aligned}\tilde{\mathbf{K}} &= \mathbf{K}[:,\mathbf{i}]\mathbf{K}[\mathbf{i},\mathbf{i}]^{-1}\mathbf{K}[:,\mathbf{i}]' \\ &= \mathbf{K}[:,\mathbf{i}]\mathbf{R}'\mathbf{R}\mathbf{K}[:,\mathbf{i}]',\end{aligned}$$

where $\mathbf{R}$ is the Cholesky decomposition of $\mathbf{K}[\mathbf{i},\mathbf{i}]^{-1}$ such that $\mathbf{R}'\mathbf{R} = \mathbf{K}[\mathbf{i},\mathbf{i}]^{-1}$

MPKFDA - Derivation II

- Rather than use the full $[m \times m]$ low rank approximation, it would be preferable to work in the $[k \times k]$ space where $k \ll m$. In order to do this we treat $\mathbf{K}[:, \mathbf{i}]\mathbf{R}'$ as a new input $\mathbf{X}$ in FDA, which in effect means we are projecting into a k-dimensional subspace. Within this space we can view

$$\tilde{\mathbf{\Sigma}}_k = \mathbf{R}\mathbf{K}[:, \mathbf{i}]'\mathbf{K}[:, \mathbf{i}]\mathbf{R}',$$

as a form of covariance matrix within this space. This trick allows us to perform nonlinear discriminant analysis on a sparse subspace using standard linear FDA

MPKFDA - Derivation III

- We can define the following maximisation problem for a dual sparse version of FDA by setting $\mathbf{w} = \mathbf{X}'\mathbf{e}_i$ where $\mathbf{e}_i$ is the i^{th} unit vector of length m, and substituting into the FDA problem described above (ignoring constants) to yield:

$$\begin{aligned} \max_i \rho_i &= \frac{\mathbf{e}_i'\mathbf{X}\mathbf{X}'\mathbf{y}\mathbf{y}\mathbf{X}\mathbf{X}'\mathbf{e}_i}{\mathbf{e}_i'\mathbf{X}\mathbf{X}'\mathbf{B}\mathbf{X}\mathbf{X}'\mathbf{e}_i} \\ &= \frac{\mathbf{K}[:,i]'\mathbf{y}\mathbf{y}'\mathbf{K}[:,i]}{\mathbf{K}[:,i]'\mathbf{B}\mathbf{K}[:,i]} \end{aligned}$$

- Maximising the quantity above leads to maximisation of the Fisher Discriminant ratio (FDR) corresponding to $\mathbf{e}_i$, and hence a sparse subset of the original KFDA problem
- We would like to find the optimal set of indices $\mathbf{i}$

MPKFDA - Derivation IV

- Proceed in a greedy manner (Matching Pursuit) similarly to [Smola and Schölkopf, 2000] and [Vincent and Bengio, 2002]
- Choose basis vectors that maximise the FDR iteratively until some pre-specified number of k vectors are chosen
- After finding the best index i orthogonalise $\mathbf{K}$ by setting $\boldsymbol{\tau} = \mathbf{K}[:, i]$, and deflating:

$$\mathbf{K} = \left(\mathbf{I} - \frac{\boldsymbol{\tau}\boldsymbol{\tau}'}{\boldsymbol{\tau}'\boldsymbol{\tau}}\right)\mathbf{K}$$

- This ensures that remaining basis vectors are chosen from an orthogonal space to those bases already chosen

MPKFDA - Derivation V

- After choosing the k training examples, giving $\mathbf{i} = (i_1, \ldots, i_k)$, we can define:

$$\mathbf{RK}[:, \mathbf{i}]'$$

  as a new data matrix.

- We then train FDA in this new projected space to find a k-dimensional weight vector $\mathbf{w}_k$. Given index j of a test point $\mathbf{x}_j$, and using the train-test kernel on this point $\mathbf{K}[j, \mathbf{i}]$ and its projection $\phi(\mathbf{x}_j) = \mathbf{RK}[j, \mathbf{i}]'$, we can make predictions using the FDA prediction function,

$$f(\mathbf{x}_j) = \operatorname{sgn}\left(\langle \tilde{\mathbf{w}}, \phi(\mathbf{x}_j) \rangle + b\right) \tag{2}$$

MPKFDA - Algorithm

Matching Pursuit Kernel Fisher Discriminant Analysis

Input: kernel $\mathbf{K}$, sparsity parameter $k > 0$, training labels $\mathbf{y}$

1. calculate matrix $\mathbf{B}$
2. initialise $\mathbf{i} = (\ )$
3. **for** $i = 1$ to k
4. set $\mathbf{i}_i$ to index of max $\frac{\mathbf{K}[:,i]'\mathbf{y}\mathbf{y}'\mathbf{K}[:,i]}{\mathbf{K}[:,i]'\mathbf{B}\mathbf{K}[:,i]}$
5. $\boldsymbol{\tau} = \mathbf{K}[:, \mathbf{i}_i]$ to deflate kernel matrix like so:
6. $\mathbf{K} = \left(\mathbf{I} - \frac{\boldsymbol{\tau}\boldsymbol{\tau}'}{\boldsymbol{\tau}'\boldsymbol{\tau}}\right)\mathbf{K}$
7. **end for**
8. calculate the projection $\mathbf{R}\mathbf{K}[:, \mathbf{i}]'$ where $\mathbf{R}$ is the Cholesky decomposition of $\mathbf{K}[\mathbf{i}, \mathbf{i}]^{-1}$ and $\mathbf{i} = (\mathbf{i}_1, \ldots, \mathbf{i}_k)$
9. train FDA in projected space to find sparse weight vector $\tilde{\mathbf{w}}$ and make predictions

Output: final set $\mathbf{i}$, (sparse) weight vector $\tilde{\mathbf{w}}$, bias term b

UCI/DELVE/STATLOG

- 13 benchmark datasets from UCI, DELVE & STATLOG, 100 predefined splits (20 for image and splice) [Mika et al., 1999]
- Compare with KFDA & SVM using RBF kernel, c.v. to select the params (RBF width , C param. in SVM, and sparsity k in MPKFDA), coarse & fine

	KFDA		MPKFDA			SVM		
	Error	s.d.	Error	s.d.	k	Error	s.d.	k
Banana	0.1069	0.0047	0.1101	0.0071	31	0.1068	0.0047	122
Breast Cancer	0.2886	0.0468	0.3174	0.0447	19	0.2603	0.0473	113
Diabetes	0.2596	0.0203	0.2543	0.0189	18	0.2332	0.0175	260
Flare Solar	0.3500	0.0168	0.3457	0.0220	19	0.3239	0.0179	557
German	0.2672	0.0248	0.2808	0.0205	27	0.2345	0.0215	392
Heart	0.2125	0.0327	0.1599	0.0312	13	0.1543	0.0326	98
Image	0.0092	0.0187	0.0136	0.0278	39	0.0061	0.0124	27
Ringnorm	0.0685	0.0108	0.0573	0.0302	15	0.0164	0.0012	216
Splice	0.0397	0.0801	0.0314	0.0633	37	0.0223	0.0450	110
Thyroid	0.0392	0.0208	0.0699	0.0310	29	0.0520	0.0208	87
Titanic	0.2259	0.0247	0.2468	0.0528	70	0.2256	0.0110	76
Twonorm	0.0253	0.0022	0.0253	0.0016	14	0.0280	0.0024	231
Waveform	0.1228	0.0053	0.1027	0.0046	13	0.1031	0.0047	121
Mean	0.1550	0.0237	0.1550	0.0274	26.5	0.1359	0.0184	185.3

High Dimensional Data

- Results from the NIPS 2003 challenge datasets
  [Guyon et al., 2004] ARCENE, DEXTER and DOROTHEA
- Main advantage of MPKFDA is in high dimensions
- Compare with KFDA & SVM using RBF kernel. 5-fold c.v.
- MPKFDA outpeforms KFDA & SVM, sparse solutions.

	KFDA	MPKFDA		SVM	
	Error	Error	k	Error	k
Arcene	0.2000	0.1800	40	0.2600	80
Dexter	0.1133	0.0800	40	0.0733	257
Dorothea	0.0971	0.0571	11	0.0686	711
Mean	0.1368	0.1057	30.3	0.1340	349.3

Conclusions and Further Work

- Performance competitive with regular KFDA and SVM
- Improved performance on high dimensional datasets
- Application to logistic regression
- Speeding up the algorithm
  - Currently $O(km^3)$
  - Using empirical bound as in Minimax paper [Lanckriet et al., 2003]
  - Approximating ρ by reduced set approach [Strohmann et al., 2003]
- Tightening the bound

Selected References


Guyon, I., Hur, A. B., Gunn, S., and Dror, G. (2004).
Result analysis of the NIPS 2003 feature selection challenge.
In *Advances in Neural Information Processing Systems 17*, pages 545–552. MIT Press.


Hussain, Z. and Shawe-Taylor, J. (2008).
Theory of matching pursuit.
Neural Information Processing Systems.


Lanckriet, G. R., Ghaoui, L. E., Bhattacharyya, C., and Jordan, M. I. (2003).
A robust minimax approach to classification.
J. Mach. Learn. Res., 3:555–582.


Mallat, S. and Zhang, Z. (1993).
Matching pursuit with time-frequency dictionaries.
IEEE Transactions on Signal Processing, 41(12):3397–3415.


Mika, S., Rätsch, G., Weston, J., Schölkopf, B., and Müller, K. (1999).
Fisher discriminant analysis with kernels.
In Y. H. Hu, J. Larsen, E. W. and Douglas, S., editors, *Proc. NNSP'99*, pages 41–48. IEEE.


Shawe-Taylor, J. and Cristianini, N. (2004).
Kernel Methods for Pattern Analysis.
Cambridge University Press, Cambridge, U.K.


Smola, A. J. and Schölkopf, B. (2000).
Sparse greedy matrix approximation for machine learning.
In *Proceedings of 17th International Conference on Machine Learning*, pages 911–918. Morgan Kaufmann, San Francisco, CA.


Strohmann, T., Belitski, A., Grudic, G., and DeCoste, D. (2003).
Sparse greedy minimax probability machine classification.
In *Neural Information Processing Systems 2003 (NIPS 2003)*. MIT Press.


Vincent, P. and Bengio, Y. (2002).
Kernel matching pursuit.
Machine Learning, 48(1-3):165–187.


Williams, C. K. I. and Seeger, M. (2001).
Using the nyström method to speed up kernel machines.
In *Advances in Neural Information Processing Systems*, volume 13, pages 682–688. MIT Press.