Multiple Kernel Learning and Feature Space Denoising

Fei Yan, Josef Kittler and Krystian Mikolajczyk
Overview of the talk

- Kernel methods
  - Kernel methods: an overview
  - Three examples: kernel PCA, SVM, and kernel FDA
  - Connection between SVM and kernel FDA
- Multiple kernel learning
  - MKL: motivation
  - $\ell_p$ regularised multiple kernel FDA
  - The effect of regularisation norm in MKL
- MKL and feature space denoising
- Conclusions
Kernel Methods: an overview

- Kernel methods: one of the most active areas in ML
- Key idea of kernel methods:
  - Embed data in input space into high dimensional feature space
  - Apply linear methods in feature space
- Input space can be: vector, string, graph, etc.
- Embedding is implicit via a kernel function $k(\cdot, \cdot)$, which defines dot product in feature space
- Any algorithm that can be written with only dot products is “kernelisable”
What is PCA

- Principal component analysis (PCA): an orthogonal basis transformation
- Transform correlated variables into uncorrelated ones (principal components)
- Can be used for dimensionality reduction
- Retains as much variance as possible when reducing dimensionality
How PCA works

- Given \( m \) centred vectors: \( \tilde{X} = (\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_m) \)
  - \( X: \tilde{d} \times m \) data matrix,
  - Eigen decomposition of covariance \( \tilde{C} = \tilde{X}\tilde{X}^T: \tilde{C}\tilde{V} = \tilde{V}\tilde{\Omega} \)
    - Diagonal matrix \( \tilde{\Omega} \): eigenvalues
    - \( \tilde{V} = (\tilde{v}_1, \tilde{v}_2, \cdots) \): eigenvectors, orthogonal basis sought
- Data can now be projected onto orthogonal basis
- Projecting only onto leading eigenvectors \( \Rightarrow \) dimensionality reduction with minimum variance loss
Kernelising PCA

- If we knew explicitly the mapping from input space to feature space $x_i = \phi(\tilde{x}_i)$:
- we could map all data: $X = \phi(\tilde{X})$, where $X$ is $d \times m$
- diagonalise the covariance in feature space $C = XX^T$: $X^T CV = X^T V \Omega$: or $KA = A \Delta$ where $K = X^T X$ and $A = X^T V$
  - Diagonal matrix $\Delta$: eigenvalues
  - $V = (v_1, v_2, \cdots)$: orthogonal basis in feature space
- However... we have $\phi(\cdot)$ only implicitly via:
  $<\phi(\tilde{x}_i), \phi(\tilde{x}_j)> = k(\tilde{x}_i, \tilde{x}_j)$
- Kernelised PCA
Kernelising PCA

- Kernel matrix $K$: evaluation of kernel function on all pairs of samples; symmetric, positive semi-definite (PSD)
- Connection between $C$ and $K$:
  - $C = XX^T$ and $K = X^T X$
  - $C$ is $d \times d$ and $K$ is $m \times m$
- $C$ is not explicitly available but $K$ is
- So we diagonalise $K$ instead of $C$: $K = A\Delta A^T$
  - $A = (\alpha_1, \alpha_2, \cdots)$: eigenvectors
Kernelising PCA

- Using the connection between $C$ and $K$, we have:
  - $C$ and $K$ have the same eigenvalues
  - Their $i$th eigenvectors are related by: $v_i = X\alpha_i$
- $v_i$ is still not explicitly available: $\alpha_i$ is, but $X$ is not
- However... we are interested in projection onto the orthogonal basis, not the basis itself
- Projection onto $v_i$: $X^T v_i = X^T X \alpha_i = K \alpha_i$
- Both $K$ and $\alpha_i$ are available.
Kernel FDA

- Kernel Fisher discriminant analysis: another supervised learning technique
- Seeking the projection $\mathbf{w}$ maximising Fisher criterion

$$\max_{\mathbf{w}} \frac{\mathbf{w}^T \frac{m}{m^+m^-} S_B \mathbf{w}}{\mathbf{w}^T (S_T + \lambda I) \mathbf{w}}$$  \hspace{1cm} (1)

- $m$: numbers of samples
- $m^+$ and $m^-$: numbers of positive and negative samples
- $S_B$ and $S_T$: between class and total scatters
- $\lambda$: regularisation parameter
Kernel FDA

- It can be proved that (3) is equivalent to

\[
\min_w \| (XP)^T w - a \|^2 + \lambda \| w \|^2
\]  

(2)

- \( P \) and \( a \): constants determined by labels

- (4) is equivalent to its Lagrangian dual:

\[
\min_\alpha \frac{1}{4} \alpha^T (I + \frac{1}{\lambda} K) \alpha - \alpha^T a
\]  

(3)

- (5) depends only on \( K \) (and labels): FDA can be kernelised
MKL: motivation

A recap on kernel methods:
- Embed (implicitly) into (very high dimensional) feature space
- Implicitly: only need dot product in feature space, i.e., the kernel function $k(\cdot, \cdot)$
- Apply linear methods in the feature space
- Easy balance of capacity (empirical error) and generalisation (norm $\mathbf{w}^T \mathbf{w}$)

These sound nice but what kernel function to use?
- This choice is critically important, for it completely determines the embedding
MKL: motivation

- Ideal case: learn kernel function from data
- If that is hard, can we learn a good combination of given kernel matrices: the multiple kernel learning problem
- Given $n$ $m \times m$ kernel matrices, $K_1, \cdots, K_n$
- Most MKL formulations consider linear combination:

$$K = \sum_{j=1}^{n} \beta_j K_j, \quad \beta_j \geq 0$$

(4)

- Goal of MKL: learn the “optimal” weights $\beta \in \mathbb{R}^n$
MKL: motivation

- Kernel matrix $K_j$: pairwise dot products in feature space $j$
- Geometrical interpretation of unweighted sum $K = \sum_{j=1}^{n} K_j$:
  - Cartesian product of the feature spaces
- Geometrical interpretation of weighted sum $K = \sum_{j=1}^{n} \beta_j K_j$:
  - Scale feature spaces with $\sqrt{\beta_j}$, then take Cartesian product
- Learning kernel weights: seeking the “optimal” scaling
MKL: motivation

- Some example definitions of “optimality”:
  - Soft margin $\Rightarrow$ multiple kernel SVM
  - Fisher criterion $\Rightarrow$ multiple kernel FDA
  - Other objectives: kernel alignment, KL divergence, etc.
- Next we propose an $\ell_p$ regularised MK-FDA
  - Learn kernel weights $\beta$ by maximising Fisher Criterion
  - Regularise $\beta$ with a general $\ell_p$ norm for any $p \geq 1$
  - Better performance than single kernel and fixed norm MK-FDA
\( \ell_p \) MK-FDA: min-max formulation

- We rewrite the kernel FDA primal problem:
  \[
  \max_w \; \frac{w^T m^+ m^{-1} S_B w}{w^T (S_T + \lambda I) w} \tag{5}
  \]

- And its Lagrangian dual:
  \[
  \min_\alpha \; \frac{1}{4} \alpha^T (I + \frac{1}{\lambda} K) \alpha - \alpha^T a \tag{6}
  \]

- For multikernel FDA, \( K \) can be chosen from a kernel set \( \mathcal{K} \):
  \[
  \max_{K \in \mathcal{K}} \min_\alpha \; \frac{1}{4} \alpha^T (I + \frac{1}{\lambda} K) \alpha - \alpha^T a \tag{7}
  \]
\( \ell_p \) MK-FDA: min-max formulation

- Consider linear combination: \( \mathcal{K} = \{ K = \sum_{i=1}^{n} \beta_i K_i : \beta \geq 0 \} \)
- \( \beta \) must be regularised in order for (9) to be meaningful
- We propose a general \( \ell_p \) regularisation for any \( p \geq 1 \):
  \( \mathcal{K} = \{ K = \sum_{i=1}^{n} \beta_i K_i : \beta \geq 0, \| \beta \|_p \leq 1 \} \)
- Substituting into (9), the \( \ell_p \) MK-FDA problem becomes:

\[
\max_{\beta} \min_{\alpha} \quad \frac{1}{4\lambda} \alpha^T \sum_{i=1}^{n} \beta_i K_i \alpha + \frac{1}{4} \alpha^T \alpha - \alpha^T a \\
\text{s.t.} \quad \beta \geq 0, \quad \| \beta \|_p \leq 1
\]
\( \ell_p \) MK-FDA: SIP formulation

- Semi-infinite program (SIP):
  - Finite number of variables, infinite many constraints
  - Efficient algorithms exist for solving SIP

- Min-max formulation (10) can be reformulated as a SIP:

\[
\max_{\theta, \beta} \quad \theta \\
\text{s.t.} \quad \beta \geq 0, \quad ||\beta||_p \leq 1, \quad S(\alpha, \beta) \geq \theta \quad \forall \alpha \in \mathbb{R}^m
\]

where

\[
S(\alpha, \beta) = \frac{1}{4\lambda} \alpha^T \sum_{i=1}^n \beta_i K_i \alpha + \frac{1}{4} \alpha^T \alpha - \alpha^T a
\]
\( \ell_p \) MK-FDA: solving the SIP with column generation

- **Column generation:**
  - Divide SIP into inner and outer subproblems
  - Alternate between the two subproblems till convergence

- **Inner subproblem:**
  - unconstrained quadratic program

- **Outer subproblem:**
  - quadratically constrained linear program

- Very efficient, and convergence is guaranteed
Effect of regularisation norm: simulation

- Sample from two heavily overlapping Gaussian distributions
- Error rate of single kernel FDA with RBF kernel: $\sim 0.43$
- Generate $n$ kernels, apply $\ell_1$ and $\ell_2$ MK-FDAs, i.e. set $p = 1$ and $p = 2$ in $\ell_p$ MK-FDA
Effect of regularisation norm: simulation

**Figure:** Error rate of $\ell_1$ MK-FDA and $\ell_2$ MK-FDA

- Both outperform single kernel, more kernels $\Rightarrow$ lower error:
  - More kernels means more dimensions, better separability
- More kernels $\Rightarrow$ more advantageous $\ell_2$ is over $\ell_1$. Why?
Effect of regularisation norm: simulation

**Figure:** Leant kernel weights. Left: $n = 5$. Right: $n = 30$.

- **Reason:** when $n$ is large, $\ell_1$ regularisation gives sparse solution, resulting in loss of information
Pascal VOC 2008 development set:
- 20 object classes \( \Rightarrow \) 20 binary problems
- Mean average precision (MAP) as performance metric

30 “informative” kernels:
- Colour SIFTs as local descriptors
- Bag-of-words model for kernel construction

Mix informative kernels with 30 random kernels
- 31 runs in total
- 1st run: 0 informative + 30 random
- 31st run: 30 informative + 0 random
Effect of regularisation norm: Pascal VOC 2008

Figure: Learnt kernel weights with various kernel mixture.

- Again, $\ell_1$ gives sparse solution and $\ell_2$ non-sparse
- A hypothesis: when most kernels are informative sparsity is a bad thing and vice versa
Effect of regularisation norm: Pascal VOC 2008

Figure: MAP vs. number of informative kernels
We have seen the behaviour of $\ell_1$ and $\ell_2$ MK-FDAs

A principle for selecting regularisation norm:
- High intrinsic sparsity in base kernels: use small norm
- Low intrinsic sparsity: use large norm

But how do we know the intrinsic sparsity?
Simple idea: try various norms, choose the best on validation

$\ell_p$ MK-FDA allows us to do this
Effect of regularisation norm: Pascal VOC 2007

Figure: Learnt kernel weights on validation set with various $p$ value.
$p = \{1, 1 + 2^{-6}, 1 + 2^{-5}, 1 + 2^{-4}, 1 + 2^{-3}, 1 + 2^{-2}, 1 + 2^{-1}, 2, 3, 4, 8, 10^6\}$, and increases from left to right, top to bottom.
Effect of regularisation norm: Pascal VOC 2007

**Figure:** APs on validation set and test set with various $p$ value. Left column: “dinningtable” class. Right column: “cat” class.
As expected, the smaller the $p$, the more sparse the learnt weights

- $p = 10^6$ is practically $\ell_\infty$, i.e. uniform weighting
- Performance on validation and test sets matches well
  - A good $p$ value on validation set is also good on test set
  - This means the optimal $p$, or the intrinsic sparsity, can be learnt
Effect of regularisation norm: Pascal VOC 2007

Table: Comparing $\ell_p$ MK-FDA and fixed norm MK-FDAs

<table>
<thead>
<tr>
<th></th>
<th>$\ell_1$ MK-FDA</th>
<th>$\ell_2$ MK-FDA</th>
<th>$\ell_\infty$ MK-FDA</th>
<th>$\ell_p$ MK-FDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAP</td>
<td>54.85</td>
<td>54.79</td>
<td>54.64</td>
<td><strong>55.61</strong></td>
</tr>
</tbody>
</table>

- By learning optimal $p$ (intrinsic sparsity) for each class, $\ell_p$ MK-FDA outperforms fixed norm MK-FDA
- $\sim 1\%$ improvement is significant: leading methods in VOC challenges differ only by a few tenths of a percent
MKL and Denoising: Experimental setup

- PASCAL VOC07 dataset, same 33 kernels as before
- Use kernel PCA for dimensionality reduction (denoising) in feature space
- Questions to be answered:
  - Can denoising improve single kernel performance?
  - Can denoising improve MKL performance?
  - How MKL behaviour differs on original kernels and denoised kernels?
MKL and Denoising: Single kernel performance

Figure: AP vs. variance kept in kernel PCA. Two kernels as examples.

- Choosing denoising level using a validation set ⇒ better single kernel performance (compared to original kernel)
In general, denoised kernels are better than original ones

- $\ell_p$ is better than fixed norm, on both original and denoised
- Advantage of $\ell_p$ is much smaller with denoised kernels. Why?
MKL and Denoising: Learnt kernel weight vs. noise level

**Figure:** Spearman’s coefficient between learnt kernel weights and variance kept in denoising. All 20 problems in PASCAL VOC07.

- Spearman’s coefficient: measure ranking correlation
Positive coefficients on most problems (16 out of 20):
- The more noisy a kernel, the lower weight it gets
- MKL essentially works by removing noise?
- Maybe this is why $\ell_p$ not as advantageous on denoised kernels?
- Maybe MKL should be done on per dimension basis instead of per kernel basis?
- Linear combination assigns same weight to all dimensions in a feature space: it cannot remove noise completely
- Maybe only nonlinear MKL can be optimal?
Conclusions

- A brief introduction to kernel methods
  - The kernel trick
  - Three examples: kernel PCA, SVM, and kernel FDA
  - Connection between SVM and kernel FDA
- Proposed an MKL method: $\ell_p$ regularised MK-FDA
  - Regularisation norm plays an important role in MKL
  - $\ell_p$ MK-FDA allows to learn intrinsic sparsity of base kernels ⇒ better performance than fixed norm MKL
Conclusions

- Investigated connection between MKL and feature space denoising
  - Denoising improves both single kernel and MKL performance
  - Positive correlation between weights and variance kept: the more noisy a kernel is, the lower its learnt weight
  - Linear kernel combination cannot take care of feature space denoising automatically
  - MKL should be done on per dimension basis instead of per kernel basis?
  - The optimal (non-linear) MKL is yet to be developed