

# Statistical Analysis of Subspace Methods and Associated Learning Algorithms

John Shawe-Taylor

School of Electronics and Computer Science  
University of Southampton  
jst@ecs.soton.ac.uk

June, 2006

Joint work with: Nello Cristianini, David Hardoon,  
Jaz Kandola, Yaoyong Li, Hongying Meng, Craig  
Saunders, Sandor Szedmak, Chris Williams, Alexei  
Vinokourov,

MFLT II Conference, June 2006

# STRUCTURE

1. Subspace methods as a learning problem
2. Example of kernel PCA
3. Analysis of kernel PCA and related bounds on empirical eigenvalues
4. Gram-Schmidt orthogonalisation – a sparse subspace method
5. Canonical correlation analysis – norm based bounds
6. SVM-2K – combining subspace with classification

## Aim:

- Problems of subspace identification can be viewed as learning
- Statistical learning theory can be applied to give bounds on ‘generalisation’
- Using subspaces for subsequent processing provides additional challenges – example of SVM-2K

## Challenge of high-dimensional data

- Curse of dimensionality and representational difficulties: seeking orthonormalised projection vectors  $\mathbf{w}_i$ ,  $i = 1, \dots, k$  defining a subspace  $V$  so that

$$P_{\mathbf{w}_i}(\mathbf{x}) = (\mathbf{w}_i \mathbf{w}_i') \mathbf{x}$$

or

$$P_V(\mathbf{x}) = (\mathbf{U}_k \mathbf{U}_k') \mathbf{x}$$

where  $\mathbf{U}_k$  is the matrix with columns  $\mathbf{w}_i$ .

- For subsequent processing we can use the mapping from the original space to the new  $k$ -dimensional representation:

$$\mathbf{x} \longmapsto \mathbf{U}_k' \mathbf{x}$$

## Example on Ionosphere data

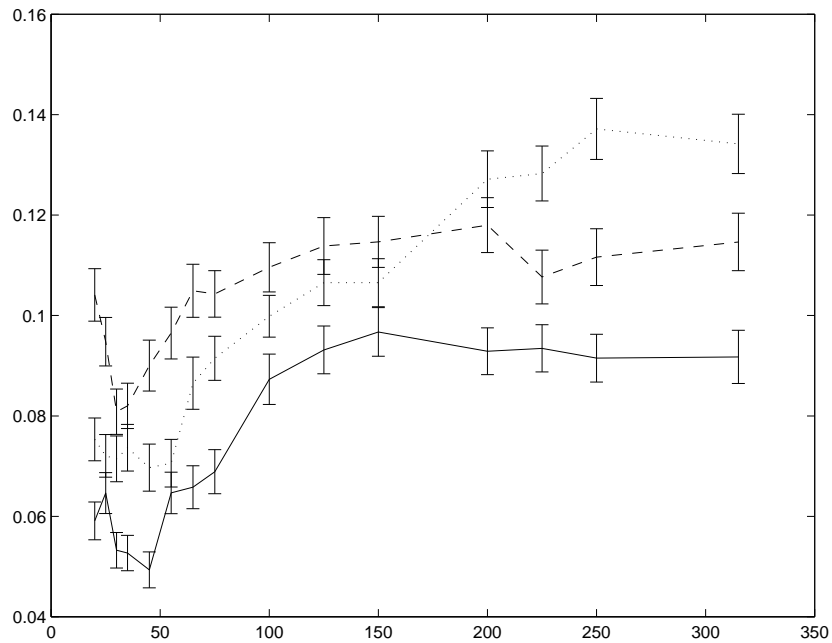


Figure 1: Error rates classifying Ionosphere data with varying polynomial kernels (2,3 and 4) and projecting into different dimensions using PCA.

## Representing the projections, I

- It may be hard to compute with explicit high-dimensional representations.
- Kernel representations provide a way round this difficulty when the input data is low-dimensional but we want to project it into a high-dimensional feature space for processing:

$$\begin{aligned} \mathbf{x} &\longmapsto \phi(\mathbf{x}) \text{ where} \\ \kappa(\mathbf{x}, \mathbf{z}) &= \langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle = \phi(\mathbf{x})' \phi(\mathbf{z}) \end{aligned}$$

## Representing the projections, II

- When the subspace is chosen in response to a set of ‘training’ data  $S = \{\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_m)\}$  it is natural that the vectors  $\mathbf{w}_i$  can be written as a linear combination of this data:

$$\mathbf{w}_i = \sum_{j=1}^m \alpha_{ij} \phi(\mathbf{x}_j) = \mathbf{X}' \alpha_i$$

where  $\mathbf{X}$  contains the training data as rows.

- This makes the new coordinates of  $\mathbf{x}$  given by

$$\mathbf{w}_i' \phi(\mathbf{x}) = \alpha_i' \mathbf{X} \phi(\mathbf{x}) = \alpha_i' \mathbf{k}$$

where  $\mathbf{k}$  is the vector with entries  $\kappa(\mathbf{x}_j, \mathbf{x})$ .

## PCA: criterion

- Criterion used by PCA is maximising variance:

$$\begin{aligned}\max_{\mathbf{w}: \|\mathbf{w}\|=1} \hat{\mathbb{E}}[P_{\mathbf{w}}(\phi(\mathbf{x}))^2] &= \max_{\mathbf{w}: \|\mathbf{w}\|=1} \frac{1}{m} \sum_{i=1}^m (\mathbf{w}' \phi(\mathbf{x}))^2 \\ &= \max_{\mathbf{w}: \|\mathbf{w}\|=1} \frac{1}{m} \mathbf{w}' \mathbf{X}' \mathbf{X} \mathbf{w}.\end{aligned}$$

- Solution  $\mathbf{w}$  is eigenvector of  $C = \mathbf{X}'\mathbf{X}$ . For  $\mathbf{u}$ ,  $\lambda$  eigenvalue pair of the matrix  $\mathbf{K} = \mathbf{X}\mathbf{X}'$

$$\mathbf{K}\mathbf{u} = \mathbf{X}\mathbf{X}'\mathbf{u} = \lambda\mathbf{u}$$

$$\Rightarrow C(\mathbf{X}'\mathbf{u}) = \mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{u}) = \lambda\mathbf{X}'\mathbf{u}$$

$$\text{So } \mathbf{w} = \frac{1}{\sqrt{\lambda}}\mathbf{X}'\mathbf{u} \text{ is an eigenvector of } C.$$



## PCA: analysis

- Kernel PCA computes the principal components of the data in the feature space using a dual representation for the eigenvectors.
- It can then project new data into the subspace spanned by the first  $k$  empirical eigenvectors.
- Can be used to clean data or simply find a low dimensional representation.
- Critical question for the application of the technique is how much of the new data is captured in the projected subspace.

## Kernel PCA learns a subspace

- Kernel PCA identifies the  $k$  dimensional subspace as that which maximises the amount of information captured for the training data.
- Hence, subspace is biased towards the training set and no guarantee that it will 'generalise' well, particularly in high dimensions?
- Can we place theoretical bounds on how well it will perform?

## Fraction of norm captured

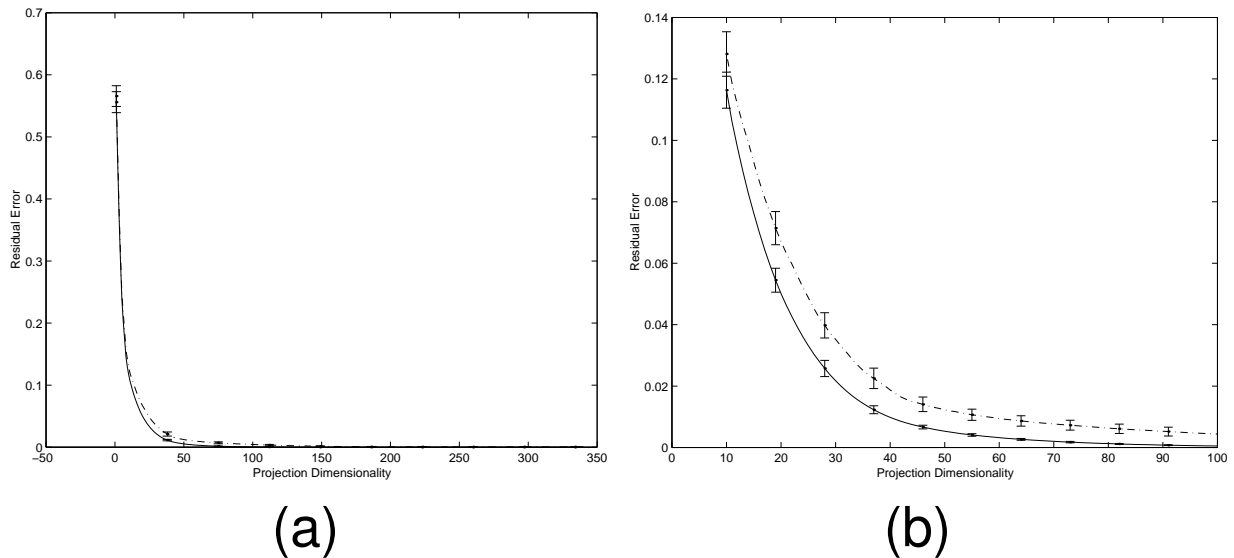


Figure 2: Fraction of the squared norm not captured by the first  $k$  eigenvectors against  $k$ . Continuous line is fraction for training set, while the dashed line is for the test set. (a) shows the full spectrum, while (b) zooms in. (Breast cancer data with a cubic polynomial kernel.)

## Eigenvalues as projection norms

- Can characterise the subspace  $\hat{V}_k$  spanned by the first  $k$  eigenvectors

$$\hat{V}_k = \operatorname{argmax}_{V:\dim(V)=k} \sum_{j=1}^m \|P_V(\phi(\mathbf{x}_j))\|^2$$

with the value of the maximum giving the sum of the relevant eigenvalues

$$\begin{aligned} \sum_{i=1}^k \hat{\lambda}_i(M) &= \sum_{j=1}^m \|P_{\hat{V}_k}(\phi(\mathbf{x}_j))\|^2 = m\hat{\mathbb{E}} \left[ \|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 \right] \\ &= \sum_{j=1}^m \|\phi(\mathbf{x}_j)\|^2 - \sum_{j=1}^m \|P_{\hat{V}_k}^\perp(\phi(\mathbf{x}_j))\|^2. \end{aligned}$$

## Process eigenvalues as projection norms

- The same analysis extends to process eigenvalues and eigenfunctions for the operator  $\mathcal{K}(\cdot)$ :

$$\mathcal{K}(f)(\mathbf{x}) = \int_{\mathcal{X}} f(\mathbf{x}') \kappa(\mathbf{x}, \mathbf{x}') dp(\mathbf{x}').$$

$$\begin{aligned} \sum_{i=1}^k \lambda_i(\mathcal{K}(f)) &= \max_{\dim(V)=k} \mathbb{E} [\|P_V(\phi(\mathbf{x}))\|^2] \\ &= \mathbb{E} [\|\phi(\mathbf{x})\|^2] - \min_{\dim(V)=k} \mathbb{E} [\|P_V^\perp(\phi(\mathbf{x}))\|^2], \end{aligned}$$

- Note that for the process eigenvalues the corresponding expectation is with respect to underlying distribution generating the data.

## Core problem considered

The questions that interest us boil down to the relationships between the following quantities:

$$\hat{\mathbb{E}} \left[ \|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 \right] = \frac{1}{m} \sum_{i=1}^k \hat{\lambda}_i$$

$$\mathbb{E} \left[ \|P_{V_k}(\phi(\mathbf{x}))\|^2 \right] = \sum_{i=1}^k \lambda_i$$

$$\mathbb{E} \left[ \|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 \right] \quad \text{and} \quad \hat{\mathbb{E}} \left[ \|P_{V_k}(\phi(\mathbf{x}))\|^2 \right].$$

## First inequalities

- Our first two observations follow simply from the characterisation of the eigenvalues as maxima

$$\hat{\mathbb{E}} \left[ \|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 \right] = \frac{1}{m} \sum_{i=1}^k \hat{\lambda}_i \geq \hat{\mathbb{E}} \left[ \|P_{V_k}(\phi(\mathbf{x}))\|^2 \right],$$

$$\mathbb{E} \left[ \|P_{V_k}(\phi(\mathbf{x}))\|^2 \right] = \sum_{i=1}^k \lambda_i \geq \mathbb{E} \left[ \|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 \right]$$

- Can show it's an approximate chain: upper right close to lower left..

## Learning the eigen-subspace

**Theorem 1.** *The projection norm  $\|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2$  is a linear function  $\hat{f}$  in a feature space  $\hat{F}$  for which the kernel function is given by*

$$\hat{\kappa}(\mathbf{x}, \mathbf{z}) = \kappa(\mathbf{x}, \mathbf{z})^2.$$

*Furthermore the 2-norm of the function  $\hat{f}$  is  $\sqrt{k}$ .*

- Proof idea: consider SVD  $X = U\Sigma V'$  then projection onto space spanned by first  $k$  eigenvectors is given by  $U_k U_k' \phi(\mathbf{x})$ , hence

$$\hat{f}(\mathbf{x}) = \|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 = \phi(\mathbf{x})' U_k U_k' U_k U_k' \phi(\mathbf{x}) = \phi(\mathbf{x})' U_k U_k' \phi(\mathbf{x})$$

# Linear projections

So if  $\alpha_{ij} = (U_k U_k')_{ij}$  we have

$$\|P_{\hat{V}_k}(\phi(\mathbf{x}))\|^2 = \sum_{ij=1}^{N_F} \alpha_{ij} \phi(\mathbf{x})_i \phi(\mathbf{x})_j = \sum_{ij=1}^{N_F} \alpha_{ij} \hat{\phi}(\mathbf{x})_{ij},$$

that is a linear function in the space with features the products of pairs of original features – this is just the feature space corresponding to the quadratic kernel

$$\hat{\kappa}(\mathbf{x}, \mathbf{z}) = \kappa(\mathbf{x}, \mathbf{z})^2.$$

## Computing the norm of $\hat{f}$

We can compute the Frobenius norm of the  $k$  dimensional projection

$$\begin{aligned}\|\hat{f}\|^2 &= \sum_{i,j=1}^{N_F} \alpha_{ij}^2 = \|U_k U_k'\|_F^2 \\ &= \left\langle \sum_{i=1}^k \mathbf{w}_i \mathbf{w}_i', \sum_{j=1}^k \mathbf{w}_j \mathbf{w}_j' \right\rangle_F \\ &= \sum_{i,j=1}^k (\mathbf{w}_i' \mathbf{w}_j)^2 = k\end{aligned}$$

where the  $\mathbf{w}_i$  are the columns of  $U_k$  that is the feature space eigenvectors.

## Function space for measuring residual

- We consider the function class  $\hat{\mathcal{F}}_{\sqrt{k}}$  with respect to the kernel

$$\hat{\kappa}(\mathbf{x}, \mathbf{z}) = \kappa(\mathbf{x}, \mathbf{z})^2,$$

augmenting the corresponding primal weight vectors with one further dimension while augmenting the corresponding input vectors with a feature

$$\begin{aligned} \|\phi(\mathbf{x})\|^2 k^{-0.25} &= \kappa(\mathbf{x}, \mathbf{x}) k^{-0.25} = k^{-0.25} \sqrt{\hat{\kappa}(\mathbf{x}, \mathbf{x})} \\ &= \|\hat{\phi}(\mathbf{x})\| k^{-0.25} \end{aligned}$$

## Associated function space

- We now apply the Rademacher theorem to the class

$$\begin{aligned}\hat{F} &= \left\{ f_\ell : (\hat{\phi}(\mathbf{x}), \|\hat{\phi}(\mathbf{x})\| k^{-0.25}) \right. \\ &\quad \left. \mapsto (\|\hat{\phi}(\mathbf{x})\| - f(\hat{\phi}(\mathbf{x}))) R^{-2} \mid f \in \hat{\mathcal{F}}_{\sqrt{k}} \cap \mathcal{P} \right\} \\ &\subseteq R^{-2} \hat{\mathcal{F}}'_{\sqrt{k+\sqrt{k}}},\end{aligned}$$

## Rademacher complexity

Following Bartlett and Mendelson (2002): The empirical Rademacher complexity of  $\mathcal{F}$  is the random variable

$$\hat{R}_m(\mathcal{F}) = \mathbb{E}_\sigma \left[ \sup_{f \in \mathcal{F}} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(\mathbf{x}_i) \right| \middle| \mathbf{x}_1, \dots, \mathbf{x}_m \right],$$

Measures ability of class to align with noise in the form of the Rademacher random variables  $\sigma$ .

## Rademacher bound

**Theorem 2.** *With probability at least  $1 - \delta$  over samples of length  $m$  every  $f \in \mathcal{F}$  satisfies*

$$\mathbb{E}_{\mathcal{D}} [f(\mathbf{z})] \leq \hat{\mathbb{E}} [f(\mathbf{z})] + \hat{R}_m(\mathcal{F}) + \sqrt{\frac{18 \ln(2/\delta)}{m}}.$$

Taking the quadratic kernel and augmenting with one extra feature equal to the norm squared we consider the class:

$$\left\{ f_{\ell} : (\hat{\phi}(\mathbf{x}), \|\hat{\phi}(\mathbf{x})\|) \mapsto \mathcal{S}(\|\hat{\phi}(\mathbf{x})\| - f(\hat{\phi}(\mathbf{x}))) \mid f \in \hat{\mathcal{F}}_{\sqrt{k}} \right\} \\ \subseteq \mathcal{S} \circ \hat{\mathcal{F}}'_{\sqrt{k+1}},$$

where  $\mathcal{S}$  is truncated scaled linear function.

## Interpreting the result

- Hence, function measures norm squared residual.
- LHS is just expected loss of projection.
- First term on RHS is empirical loss.
- Second term is empirical Rademacher complexity which is bounded by

$$\hat{R}_m(\mathcal{S} \circ \hat{\mathcal{F}}'_{\sqrt{k+1}}) \leq \frac{4}{R^2} \sqrt{\frac{k+1}{m}} \sqrt{\frac{2}{m} \sum_{i=1}^m \kappa(\mathbf{x}_i, \mathbf{x}_i)^4}.$$

## Putting it all together

gives the final bound:

**Theorem 3.** *If we perform PCA in the feature space defined by a kernel  $\kappa(\mathbf{x}, \mathbf{z})$  and project new data onto the space  $\hat{V}_k$ , with probability greater than  $1 - \delta$  the expected squared residual is bounded by*

$$\mathbb{E} \left[ \|P_{\hat{V}_k}^\perp(\phi(\mathbf{x}))\|^2 \right] \leq \frac{1}{m} \hat{\lambda}^{>k}(S) + 4\sqrt{\frac{k+1}{m}} \sqrt{\frac{2}{m} \sum_{i=1}^m \kappa(\mathbf{x}_i, \mathbf{x}_i)} + R^2 \sqrt{\frac{18 \ln(2/\delta)}{m}},$$

where the support of the distribution is in a ball of radius  $R$ .

## Implications

Note that the bound implies a bound on the difference between the process and empirical eigenvalues since

$$\mathbb{E} \left[ \|P_{\hat{V}_k}^\perp(\phi(\mathbf{x}))\|^2 \right] = \sum_{i>k} \lambda_i \leq \mathbb{E} \left[ \|P_{\hat{V}_k}^\perp(\phi(\mathbf{x}))\|^2 \right]$$

implying with probability  $1 - \delta$  that

$$\sum_{i>k} \lambda_i - \frac{1}{m} \hat{\lambda}^{>k}(S) \leq 4 \sqrt{\frac{k+1}{m}} \sqrt{\frac{2}{m} \sum_{i=1}^m \kappa(\mathbf{x}_i, \mathbf{x}_i)^4} + R^2 \sqrt{\frac{18 \ln(2/\delta)}{m}},$$

## Gram-Schmidt orthonormalisation

- PCA maximises the variance overall, but can be expensive to compute for large matrices.
- An alternative is to perform a greedy choice by choosing largest residual norm.

```
for  $i = 1$  to  $m$  do norm[ $i$ ] =  $\kappa(x_i, x_i)$ ;  
for  $j = 1$  to  $T$  do  
   $i_j = \operatorname{argmax}_i(\text{norm}[i])$ ;  
   $s[j] = \sqrt{\text{norm}[i_j]}$ ;  
  for  $i = 1$  to  $m$  do  
     $\psi[i, j] = \left( \kappa(x_i, x_{i_j}) - \sum_{t=1}^{j-1} \psi[i, t] * \psi[i_j, t] \right) / s[j]$ ;  
    norm[ $i$ ] = norm[ $i$ ] -  $\psi(i, j) * \psi(i, j)$ ;  
  end;  
end;  
return  $\psi[i, j]$  as the  $j$ -th feature of input  $i$ ;
```

## Cholesky is dual Gram-Schmidt

- We have deliberately written the algorithm in dual form – we observe that it is identical to the incomplete Cholesky decomposition of the kernel matrix.

Table 1: F1 scores for text classification using kPCA and the Gram-Schmidt technique (GSK) compared with SVM

QUERY	kPCA		GSK		SVM	
	MEAN	SD	MEAN	SD	MEAN	SD
20	0.436	0.129	0.473	0.117	0.433	0.122
23	0.374	0.124	0.589	0.127	0.317	0.131

## Sparsity bound for G-S

- We can use the sparsity bound approach used for compression schemes/ set cover machine: consider all  $\binom{m}{k}$  sets of  $k$  indices, and for each consider performing a G-S on these examples.
- We can now view the remaining  $m - k$  examples as test examples providing an estimate of the error of the function  $f$  that measures the residual of the projection into this space:

$$\mathbb{E}_{\mathcal{D}} [f(\mathbf{z})] \leq \hat{\mathbb{E}} [f(\mathbf{z})] + R \sqrt{\frac{k \ln \frac{m}{k} + \ln \frac{2}{\delta}}{2(m - k)}}.$$

- This has a similar form to the PCA bound but tighter constants and no crossterms.

## Canonical correlation analysis

- Consider the situation where we have two views of each item, eg translations of a document, image and caption, etc. Can consider as two projections:

$$\phi_a(\mathbf{x}) \longleftarrow \mathbf{x} \longrightarrow \phi_b(\mathbf{x})$$

- Seek directions  $\mathbf{w}_a$  and  $\mathbf{w}_b$  such that correlation over the training set is maximised:

$$\begin{aligned} \rho &= \frac{\hat{\mathbb{E}} \left[ \mathbf{w}_a^T \phi_a(\mathbf{x}) \phi_b(\mathbf{x})^T \mathbf{w}_b \right]}{\sqrt{\hat{\mathbb{E}} \left[ \mathbf{w}_a^T \phi_a(\mathbf{x}) \phi_a(\mathbf{x})^T \mathbf{w}_a \right] \hat{\mathbb{E}} \left[ \mathbf{w}_b^T \phi_b(\mathbf{x}) \phi_b(\mathbf{x})^T \mathbf{w}_b \right]}} \\ &= \frac{\mathbf{w}_a^T \mathbf{X}'_a \mathbf{X}'_b \mathbf{w}_b}{\sqrt{\mathbf{w}_a^T \mathbf{X}'_a \mathbf{X}_a \mathbf{w}_a \mathbf{w}_b^T \mathbf{X}'_b \mathbf{X}_b \mathbf{w}_b}}, \end{aligned}$$



## Canonical correlation analysis

- Danger of overfitting since in high-dimensional spaces we can find a perfect fit for  $\mathbf{w}_b$  whatever  $\mathbf{w}_a$  we choose.
- Consider the function:

$$g_{\mathbf{w}_a, \mathbf{w}_b}(\mathbf{x}) = \left\| \mathbf{w}_a^T \phi_a(\mathbf{x}) - \mathbf{w}_b^T \phi_b(\mathbf{x}) \right\|^2,$$

if  $\|\mathbf{w}_a\| \leq A$  and  $\|\mathbf{w}_b\| \leq B$  we can bound

$$\begin{aligned} \mathbb{E} \left[ g_{\mathbf{w}_a, \mathbf{w}_b}(\mathbf{x}) \right] &\leq \sum_{i=1}^k 2(1 - \rho_i) + 3R(A^2 + B^2) \sqrt{\frac{\ln(\frac{2}{\delta})}{2m}} \\ &\quad + \frac{4(A^2 + B^2)k}{m} \sqrt{\sum_{i=1}^m (\kappa_a(x_i, x_i) + \kappa_b(x_i, x_i))^2} \end{aligned}$$



## Regularised Canonical correlation analysis

- Hence problem of overfitting can be tackled by controlling norms. Leads to generalised eigenvalue equation:

$$\begin{pmatrix} \mathbf{0} & \mathbf{K}_a \mathbf{K}_b \\ \mathbf{K}_b \mathbf{K}_a & \mathbf{0} \end{pmatrix} \begin{pmatrix} \alpha_a \\ \alpha_b \end{pmatrix} = \lambda \begin{pmatrix} (1 - \tau_a) \mathbf{K}_a^2 + \tau_a \mathbf{K}_a & \mathbf{0} \\ \mathbf{0} & (1 - \tau_b) \mathbf{K}_b^2 + \tau_b \mathbf{K}_b \end{pmatrix} \begin{pmatrix} \alpha_a \\ \alpha_b \end{pmatrix}.$$

where  $\tau_a$  and  $\tau_b$  are the regularisation parameters applied to the two norms.

## Space identified by CCA

- Subspace identified by CCA should capture the common semantics of the two views – effectively cleaning each of any view-specific noise.
- The subspace projection can then be used for different types of further processing.
- An example with English/Japanese documents performing mate retrieval (1000 training documents): the accuracy rates averaged over 2000 test documents.

#Eigenvectors	5	10	50	100	200	500	1000
Test docs as queries							
KCCA(E→J)	0.050	0.154	0.401	0.471	0.534	0.486	0.377
KCCA(J→E)	0.084	0.173	0.369	0.448	0.461	0.369	0.272
LSI(E→J)	0.037	0.095	0.296	0.376	0.431	0.393	0.247
LSI(J→E)	0.029	0.079	0.212	0.294	0.362	0.304	0.170

## Processing with subspaces

- We have seen several examples of using subspaces inferred on a training set being used in a subsequence analysis, eg retrieval or classification
- Standard learning theory requires the function space to be fixed before the sample is generated and so cannot be applied to these cases
- We now present an example of a two stage learning algorithm for which analysis can be undertaken

## SVM-2k: definition

- In the patent example we could envisage a two-phase process: first identify the subspace using KCCA and then train an SVM on the elicited features to perform cross-lingual classification.
- SVM-2k attempts to combine these two stages into one.
- Train two linear functions (SVMs)  $f_a$  and  $f_b$  on the two views, but add a further series of constraints:

$$|f_a(x_i) - f_b(x_i)| \leq \epsilon + \eta_i, i = 1, \dots, m$$

and add  $D \sum_{i=1}^m \eta_i$  to the objective.  
Note: this is before thresholding.

## SVM-2K

- Two kernels and associated constraints:

$$\begin{aligned} \min L &= \frac{1}{2} \|w_A\|^2 + \frac{1}{2} \|w_B\|^2 + C^A \sum \xi_i^A + C^B \sum \xi_i^B + D \sum \eta_i \\ \text{s.t. } & |\langle w_A, \phi_A(x_i) \rangle - \langle w_B, \phi_B(x_i) \rangle| \leq \eta_i + \epsilon \\ & y_i \langle w_A, \phi_A(x_i) \rangle \geq 1 - \xi_i^A \\ & y_i \langle w_B, \phi_B(x_i) \rangle \geq 1 - \xi_i^B \\ & \xi_i^A \geq 0 \quad \xi_i^B \geq 0 \quad \eta_i \geq 0 \end{aligned}$$

- Let  $\hat{w}_A, \hat{w}_B$  be the solution to this optimisation problem. The final decision function is then

$$\begin{aligned} f(x) &= 0.5 (\langle \hat{w}_A, \phi_A(x) \rangle + \langle \hat{w}_B, \phi_B(x) \rangle) \\ &= 0.5 (f_A(x) + f_B(x)). \end{aligned}$$

## Rademacher Analysis

First observe that an application of the Rademacher bound shows that

$$\begin{aligned}\mathbb{E}_x[|f_A(x) - f_B(x)|] &\leq \mathbb{E}_x[|\langle \hat{w}_A, \phi_A(x) \rangle - \langle \hat{w}_B, \phi_B(x) \rangle|] \\ &\leq \epsilon + \frac{1}{m} \sum_{i=1}^m \eta_i + \frac{2C}{m} \sqrt{\text{tr}(K_A) + \text{tr}(K_B)} + 3\sqrt{\frac{\ln(2/\delta)}{m}} =: D\end{aligned}$$

with probability at least  $1 - \delta$ . Hence, the class of functions used is

$$\mathcal{F}_C = \left\{ f \mid f : x \rightarrow 0.5 \left( \sum_{i=1}^m [\alpha_A^i \kappa_A(x_i, x) + \alpha_B^i \kappa_B(x_i, x)] \right), \right. \\ \left. \alpha_A' K_A \alpha_A \leq C^2, \alpha_B' K_B \alpha_B \leq C^2, \mathbb{E}_x[|f_A(x) - f_B(x)|] \leq D \right\}$$

## Rademacher bounds for SVM-2K

- Consider the function of two weight vectors  $w_A$  and  $w_B$ ,

$$D(w_A, w_B) := \mathbb{E}_x [|\langle w_A, \phi_A(x) \rangle - \langle w_B, \phi_B(x) \rangle|]$$

- Our Rademacher complexity is therefore

$$\begin{aligned} \hat{R}_m^*(\mathcal{F}_C) &= \mathbb{E}_\sigma \left[ \sup_{f \in \mathcal{F}_C} \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right] \\ &= \mathbb{E}_\sigma \left[ \sup_{\substack{\|w_A\| \leq C \\ \|w_B\| \leq C \\ D(w_A, w_B) \leq D}} \frac{1}{m} \sum_{i=1}^m \sigma_i [\langle w_A, \phi_A(x_i) \rangle + \langle w_B, \phi_B(x_i) \rangle] \right] \end{aligned}$$



## Rademacher bounds for SVM-2K

A reverse Rademacher theorem shows that for weight vectors  $w_A$  and  $w_B$  satisfying  $D(w_A, w_B) \leq D$ , with probability at least  $1 - \delta$  we have

$$\begin{aligned}\hat{D}(w_A, w_B) &:= \mathbb{E}_S[|\langle w_A, \phi_A(x) \rangle - \langle w_B, \phi_B(x) \rangle|] \\ &\leq D + \frac{2C}{m} \sqrt{\text{tr}(K_A) + \text{tr}(K_B)} + 3\sqrt{\frac{\ln(2/\delta)}{m}} \\ &\leq \epsilon + \frac{1}{m} \sum_{i=1}^m \eta_i + \frac{4C}{m} \sqrt{\text{tr}(K_A) + \text{tr}(K_B)} \\ &\quad + 6\sqrt{\frac{\ln(2/\delta)}{m}} =: \hat{D}\end{aligned}$$

## Evaluating the bound

- By an application of McDiarmid can fix one random evaluation  $\sigma$ , so Rademacher complexity bounded by

$$\hat{R}_m^*(\mathcal{F}_C) \leq \sup_{\substack{\|w_A\| \leq C \\ \|w_B\| \leq C \\ \hat{D}(w_A, w_B) \leq \hat{D}}} \frac{1}{m} \sum_{i=1}^m \sigma_i [\langle w_A, \phi_A(x_i) \rangle + \langle w_B, \phi_B(x_i) \rangle] \\ + RC \sqrt{\frac{2}{m} \log \frac{1}{\delta}}$$

Evaluating this bound involves solving an optimisation problem

- In practice significant reductions in complexity are achieved with corresponding improvement in classification accuracy.

## SVM-2k: results

- Results obtained classifying patents from a Japanese patent dataset with paired English translations.
- Results are average precision as percent – i.e. higher is better.
- Note that SVM-2k<sub>j</sub> is performing crosslingual classification, i.e. only uses Japanese text – and often does better than a SVM in original language

	pSVM	kcca_SVM	SVM	SVM-2k <sub>j</sub>	Concat	SVM-2k
1	59.4±3.9	60.3±2.8	66.6±2.8	66.1± 2.6	67.5±2.3	67.5±2.1
2	71.1±4.5	68.4±4.4	73.0±4.0	74.8±4.7	73.9±4.0	75.1±4.1
3	16.7±1.2	13.1±1.0	18.8±1.6	20.8±1.9	21.5±1.9	22.5±1.7
7	74.9±1.8	76.0±1.2	76.7±1.3	77.5±1.4	79.0±1.2	80.7±1.5
12	75.0±0.8	73.6±0.8	76.8±1.0	77.6±0.7	76.8±0.6	78.4±0.6
14	76.0±1.6	71.5±1.5	80.9±1.3	82.2±1.3	81.4±1.4	82.7±1.3

## SVM-2k: semi-supervised

- In the definition of the restriction placed on the two functions  $f_A$  and  $f_B$  we didn't need to use the same data as for training.
- note that we don't need labels to evaluate  $\hat{D}(w_A, w_B)$  so can use unlabelled data for this
- Gives semi-supervised learning algorithm with even larger reductions in Rademacher complexity.

## SVM-2k: semi-supervised results

- Results with image classification – object detection
- Two views are interest points and image patches

		<b>Airplanes</b>	<b>Faces</b>	<b>Motorbikes</b>
<b>SVM on</b>	mean(std)	91.4(2.8)	96.8(1.3)	94.1(1.3)
<b>two feats</b>	Rad. comp.	588	339	574
<b>SVM_2K</b>	mean(std)	<b>91.9(2.1)</b>	<b>97.4(1.3)</b>	<b>94.3(1.1)</b>
	Rad. comp.	236	34	197

Accuracies(%), standard deviation and estimation of Rademacher Complexities of the SVM acting on two feature sets and the SVM\_2K in three image classes. 5% of the cases were labeled.

## Semi-supervised framework

- General example of semi-supervised framework proposed by Blum and Balcan.
- Learn but restrict functions  $f$  to have some low average over test data when measured by a compatibility function  $\chi$ .
- In our case functions are averaged pairs  $0.5(f_A + f_B)$  with the compatibility measured by  $|f_A - f_B|$ .
- Example of more general ‘luckiness’, in which we posit properties such as large margin that if observed imply a much lower complexity of function space.

# Conclusions

- Subspace methods as learning allows analysis within standard frameworks
- Example of kernel PCA also illuminates relation between process and empirical eigenvalues
- Can get similar bounds using sparsity arguments for Gram-Schmidt orthonormalisation
- Canonical correlation analysis can also be analysed within the framework suggesting an appropriate regularisation strategy
- The question of combining with postprocessing led to SVM-2k algorithm and analysis