Matching pursuit algorithms in machine learning

Zakria Hussain and John Shawe-Taylor

Centre for Computational Statistics and Machine Learning,
Department of Computer Science,
University College London

Workshop on Sparsity and Inverse Problems in Statistical Theory and Econometrics, 2008

Outline

- Matching pursuit kernel principal components analysis algorithm
- Show MPKPCA is a compression scheme
- Kernel matching pursuit
- Bound for KMP
- Experiments for MPKPCA and KMP bounds
- Extensions of MP to kernel canonical correlation analysis
- Experiments for MPKCCA
- Bound on each dimension for MPKCCA

Preliminaries

- Examples and outputs:

$$\mathbf{X} = \begin{pmatrix} \phi(\mathbf{x}_{11}) & \dots & \phi(\mathbf{x}_{1m}) \\ \vdots & \vdots & \vdots \\ \phi(\mathbf{x}_{n1}) & \dots & \phi(\mathbf{x}_{nm}) \end{pmatrix}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix},$$

  where data already assumed to be mapped into higher dimensional feature space using the mapping ϕ.

- Kernel matrix $\mathbf{K} = \mathbf{X}^\top \mathbf{X}$.
- Unit vector

$$\mathbf{e}_i = (0 \quad \dots \quad 0 \quad \overset{i}{1} \quad 0 \quad \dots \quad 0)^\top$$

Kernel principal components analysis

- Linear PCA with (primal) eigenvectors $\mathbf{w}$:

$$\max_{\mathbf{w}} \frac{\mathbf{w}^\top \mathbf{X}\mathbf{X}^\top \mathbf{w}}{\mathbf{w}^\top \mathbf{w}}$$

- Non-linear kernel PCA with (dual) eigenvectors $\boldsymbol{\alpha}$ by making substitution $\mathbf{w} = \mathbf{X}\boldsymbol{\alpha}$:

$$\max_{\boldsymbol{\alpha}} \frac{\boldsymbol{\alpha}^\top \mathbf{X}^\top \mathbf{X}\mathbf{X}^\top \mathbf{X}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\alpha}} = \max_{\boldsymbol{\alpha}} \frac{\boldsymbol{\alpha}^\top \mathbf{K}^\top \mathbf{K}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top \mathbf{K}\boldsymbol{\alpha}}$$

Sparse KPCA

- Let $\mathbf{i} = (i_1, \ldots, i_k)$ be a vector of indices pointing to k training examples in S *i.e.*, $S_{\mathbf{i}}$.
- Sparsely represented vector $\mathbf{w} = \mathbf{X}[:, \mathbf{i}]\tilde{\alpha}$, that is a linear combination of a small number of training examples indexed by vector $\mathbf{i}$, $|\tilde{\alpha}| = k$.
- we have the following SKPCA maximisation problem,

$$\max_{\tilde{\alpha}, \mathbf{i}} \frac{\tilde{\alpha}^\top \mathbf{X}[:, \mathbf{i}]^\top \mathbf{X}\mathbf{X}^\top \mathbf{X}[:, \mathbf{i}]\tilde{\alpha}}{\tilde{\alpha}^\top \mathbf{X}[:, \mathbf{i}]^\top \mathbf{X}[:, \mathbf{i}]\tilde{\alpha}},$$

with $\mathbf{K}[:, \mathbf{i}] = \mathbf{X}^\top \mathbf{X}[:, \mathbf{i}]$,

$$\max_{\tilde{\alpha}, \mathbf{i}} \frac{\tilde{\alpha}^\top \mathbf{K}[:, \mathbf{i}]^\top \mathbf{K}[:, \mathbf{i}]\tilde{\alpha}}{\tilde{\alpha}^\top \mathbf{K}[\mathbf{i}, \mathbf{i}]\tilde{\alpha}}$$

Matching pursuit kernel principal components analysis (MPKPCA)

Smola and Schölkopf (2000) solve this problem by using a matching pursuit technique – which turns out to be the following two-step procedure:

- Maximise:

$$i^* = \arg\max_i \frac{\mathbf{e}_i^\top \mathbf{K}^\top \mathbf{K} \mathbf{e}_i}{\mathbf{e}_i^\top \mathbf{K} \mathbf{e}_i} \qquad i = 1, \ldots, m,$$

- Deflate (orthogonalise):

$$\mathbf{K} = \mathbf{K} - \frac{\mathbf{K}[:, i^*]\mathbf{K}[:, i^*]^\top}{\mathbf{K}[i, i]}.$$

MPKPCA projection

An orthogonal projection $P_{\mathbf{i}}(\phi(\mathbf{x}_j))$ of a feature vector $\phi(\mathbf{x}_j)$ into a subspace defined by $\mathbf{i}$ in the kernel defined feature space:

$$\mathbf{K}[j, \mathbf{i}]\mathbf{K}[\mathbf{i}, \mathbf{i}]^{-1}\mathbf{K}[\mathbf{i}, j],$$

with $\mathbf{K}[j, \mathbf{i}]$ denoting the kernel entries between the index set $\mathbf{i}$ and the feature vector $\phi(\mathbf{x}_j)$.

Generalisation error bound for MPKPCA

- Let $\mathcal{A}(S)$ be the function output by learning algorithm $\mathcal{A}$ on training set S. A sample compression scheme is a reconstruction function Φ mapping a compression set $\Lambda(S)$ to some set of functions $\mathcal{H}$ such that

$$\mathcal{A}(S) = \Phi(\Lambda(S)).$$

- The MPKPCA projection defines a compression scheme *e.g.*,

$$\mathbf{K}[j, \mathbf{i}]\mathbf{K}[\mathbf{i}, \mathbf{i}]^{-1}\mathbf{K}[j, \mathbf{i}]^{\top}$$

only requires information from $\mathbf{i}$ and the new test example $\phi(\mathbf{x}_j)$.

Sample compression bound

Introduced by Littlestone and Warmuth (1986). Bounds the generalisation error using the size of the compression set *i.e.*, number of examples in $S_{\mathbf{i}}$.

Theorem (Sample compression bound)

Consider a compression scheme $\Phi(\Lambda(S))$. *For any probability distribution* $\mathcal{D}$ *on* $\mathcal{X} \times \{-1, 1\}$, *with probability* $1 - \delta$ *over* m *random examples* S, *the true error* $\mathrm{err}(f)$ *of a hypothesis* $f \in \mathcal{H}$ *defined by a compression set of size* d *can be upper bounded by,*

$$\mathrm{err}(f) \quad \leq \quad \frac{1}{m-d}\left[d \ln\left(\frac{em}{d}\right) + \ln\left(\frac{m}{\delta}\right)\right].$$

Generalisation error bound for MPKPCA

Theorem (Hussain and Shawe-Taylor (2008))

Let $\mathcal{A}_k$ be any learning algorithm having a reconstruction function that maps compression sets to subspaces. Let m be the size of the training set S, let k be the size of the compression set, let $\hat{\mathcal{E}}_{m-k}[\ell(\mathcal{A}_k(S))]$ be the residual loss between the $m-k$ points outside of the compression set and their projections into a subspace, then with probability $1-\delta$, the expected loss $\mathcal{E}[\ell(\mathcal{A}_k(S))]$ of algorithm $\mathcal{A}_k$ given any training set S can be bounded by,

$$\mathcal{E}[\ell(\mathcal{A}_k(S))] \leq \min_{1\leq t\leq k}\left[\hat{\mathcal{E}}_{m-t}[\ell(\mathcal{A}_t(S))]+\sqrt{\frac{R^2}{2(m-t)}\left[t\ln\left(\frac{em}{t}\right)+\ln\left(\frac{2m}{\delta}\right)\right]}\right],$$

where $\ell(\cdot)\geq 0$ and $R=\sup\ell(\cdot)$.

Bound specialised for the Gaussian kernel

Let the corresponding loss function (residual) be defined as

$$\ell(\mathcal{A}_k(S))(\mathbf{x}) \quad = \quad \|\mathbf{x} - P_{\mathbf{i}_{1\ldots k}}(\mathbf{x})\|^2,$$

where $\mathbf{x}$ is a test point and $P_{\mathbf{i}_{1\ldots k}}(\mathbf{x})$ its projection into the subspace determined by the set $\mathbf{i}_1, \ldots, \mathbf{i}_k$ of indices returned by $\mathcal{A}_k(S)$.

Corollary (Sample compression bound for sparse KPCA)

Using a Gaussian kernel and all of the definitions from the Theorem above, we get the following bound:

$$\mathcal{E}[\ell(\mathcal{A}_k(S))] \quad \leq \quad \min_{1\leq t\leq k} \left[\frac{1}{m-t} \sum_{i=1}^{m-t} \|\mathbf{x}_i - P_{\mathbf{i}_{1\ldots t}}(\mathbf{x}_i)\|^2 + \sqrt{\frac{1}{2(m-t)} \left[t \ln\left(\frac{em}{t}\right) + \ln\left(\frac{2m}{\delta}\right) \right]} \right]$$

Bound comparisons: real world data set

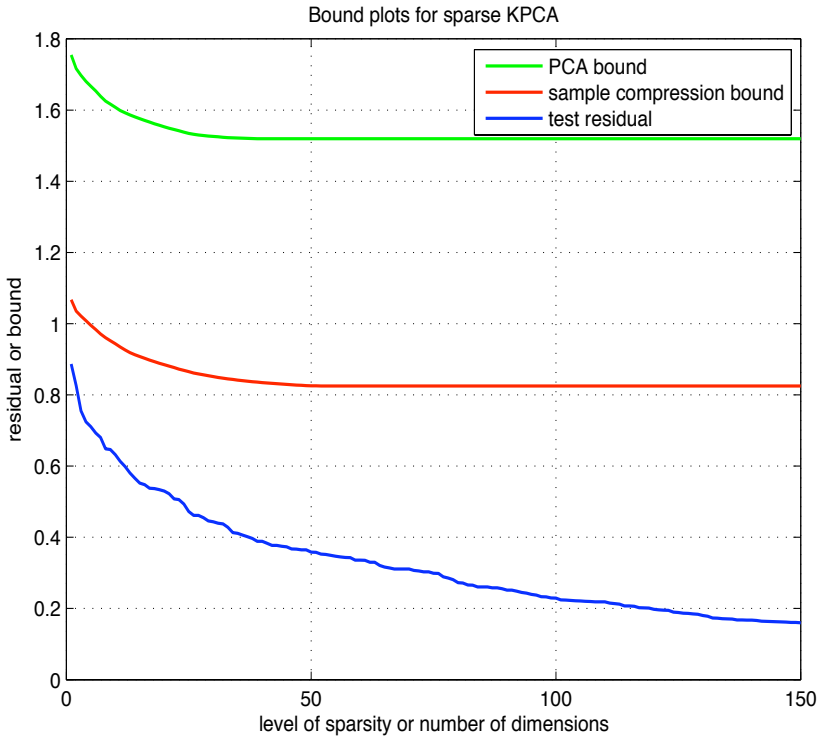

Figure: Boston housing data: bound plots for MPKPCA comparing the sample compression bound proposed and the PCA bound of Shawe-Taylor et al (2005). We plot the residual loss and bound values against the level of sparsity (sample compression bound) and the number of dimensions used (PCA bound).

Bound comparisons: artificial data set

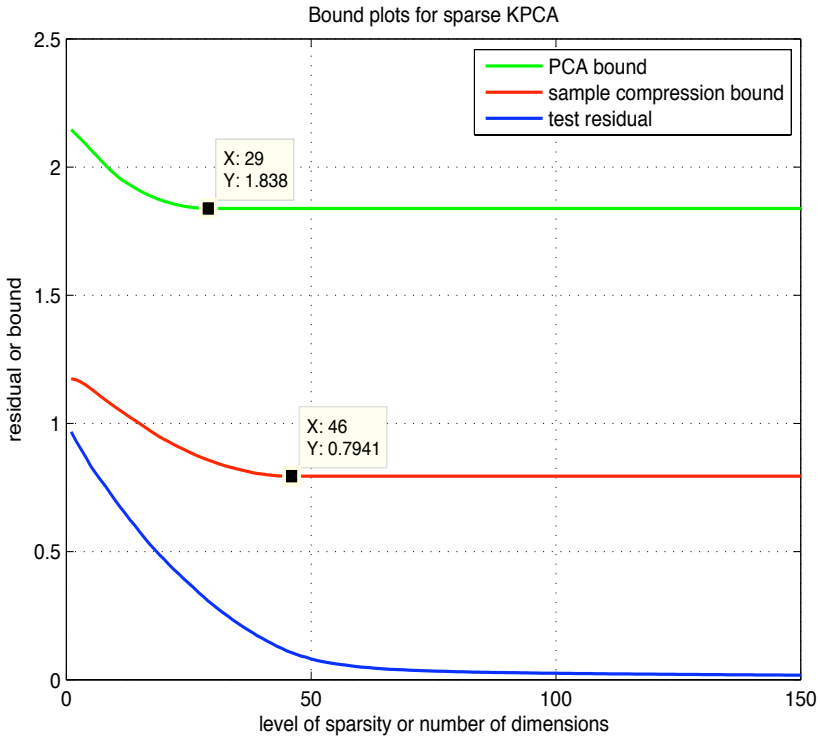

Figure: A plot of the residual loss and bound values against the level of sparsity (sample compression bound) and the number of dimensions used (PCA bound). A Toy experiment with 1000 training examples (and 450 dimensions) drawn randomly from a Gaussian distribution with zero mean and unit variance.

Sparse kernel least squares regression

Let $\mathbf{i} = (i_1, \ldots, i_k)$ be a vector of indices pointing to columns (examples) of the data. By letting $\mathbf{w} = \mathbf{X}[:, \mathbf{i}]\tilde{\alpha}$ and substituting into least squares regression problem we have

$$\max_{\mathbf{i}, \tilde{\alpha}} \|\mathbf{y} - \mathbf{X}^\top \mathbf{X}[:, \mathbf{i}]\tilde{\alpha}\|_2^2$$

$$\max_{\mathbf{i}, \tilde{\alpha}} \|\mathbf{y} - \mathbf{K}[:, \mathbf{i}]\tilde{\alpha}\|_2^2$$

This problem is sparse in the dual.

Fix i,

$$\begin{aligned}
\|\mathbf{y} - \mathbf{K}[:,i]\tilde{\alpha}_i\|^2 &= \left\|\mathbf{y} - \mathbf{K}[:,i]\frac{\mathbf{K}[:,i]^\top \mathbf{y}}{\|\mathbf{K}[:,i]\|^2}\right\|^2 \\
&= \|\mathbf{y}\|^2 - 2\mathbf{K}[:,i]^\top \mathbf{y}\frac{\mathbf{K}[:,i]^\top \mathbf{y}}{\|\mathbf{K}[:,i]\|^2} + \\
&\quad \left(\frac{\mathbf{K}[:,i]^\top \mathbf{y}}{\|\mathbf{K}[:,i]\|^2}\right)^2 \|\mathbf{K}[:,i]\|^2 \\
&= \|\mathbf{y}\|^2 - \left(\frac{\mathbf{K}[:,i]^\top \mathbf{y}}{\|\mathbf{K}[:,i]\|}\right)^2 .
\end{aligned}$$

Hence maximise the final component of the last line,

$$\arg\max_i \left|\frac{\mathbf{K}[:,i]^\top \mathbf{y}}{\|\mathbf{K}[:,i]\|}\right| \qquad \forall i = 1, \ldots, m,$$

which corresponds to finding the kernel basis vector that is most collinear with the regression output vector $\mathbf{y}$.

Kernel matching pursuit

Vincent and Bengio (2002) propose an efficient solution that also updates the dual weight vector on-the-fly called KMP with prefitting. We simplify their algorithm here slightly and present the following two-step algorithm.

- Maximise:

$$i^* = \arg\max_i \left| \frac{\mathbf{K}[:, i]^\top \mathbf{y}}{\|\mathbf{K}[:, i]\|} \right| \qquad i = 1, \ldots, m,$$

- Deflate (orthogonalise):

$$\mathbf{K} = \left(\mathbf{I} - \frac{\mathbf{K}[:, i^*]\mathbf{K}[:, i^*]^\top}{\mathbf{K}[:, i^*]^\top \mathbf{K}[:, i^*]} \right) \mathbf{K}.$$

Generalisation error bound for KMP

- KMP is not a compression scheme as the "rectangular" kernel matrix ($m \times k$) requires all of the information from the m data points.
- Important to notice that the VC-dimension for the set of k-dimensional linear threshold functions is simply the cardinality k.
- Using this fact we can upper bound KMP in the feature space using a compression scheme argument.

Bound for KMP

Theorem (Hussain and Shawe-Taylor (2008))

Fix $\alpha \in \mathbb{R}$, $\alpha > 0$. Let $\mathcal{A}$ be the regression algorithm of KMP, m the size of the training set S and k the number of chosen indices in $\mathbf{i}$. Let S be reordered so that the last $m-k$ points are outside of the set $S_{\mathbf{i}}$ and let $t = \sum_{i=m-k}^{m} \mathbb{I}(|f(\mathbf{x}_i) - y_i| > \alpha)$ be the number of errors for those points in $S \setminus S_{\mathbf{i}}$. Then with probability $1-\delta$ over the generation of the training set S the expected loss $\mathcal{E}[\ell(\cdot)]$ of algorithm $\mathcal{A}$ can be bounded by,

$$\mathcal{E}[\ell(\mathcal{A}(S))] \leq \frac{2}{m-k-t}\left[(k+1)\log\left(\frac{4e(m-k-t)}{k+1}\right)\right.$$
$$\left. + k\log\left(\frac{em}{k}\right) + t\log\left(\frac{e(m-k)}{t}\right) + \log\left(\frac{2m^2}{\delta}\right)\right].$$

KMP bound experiments

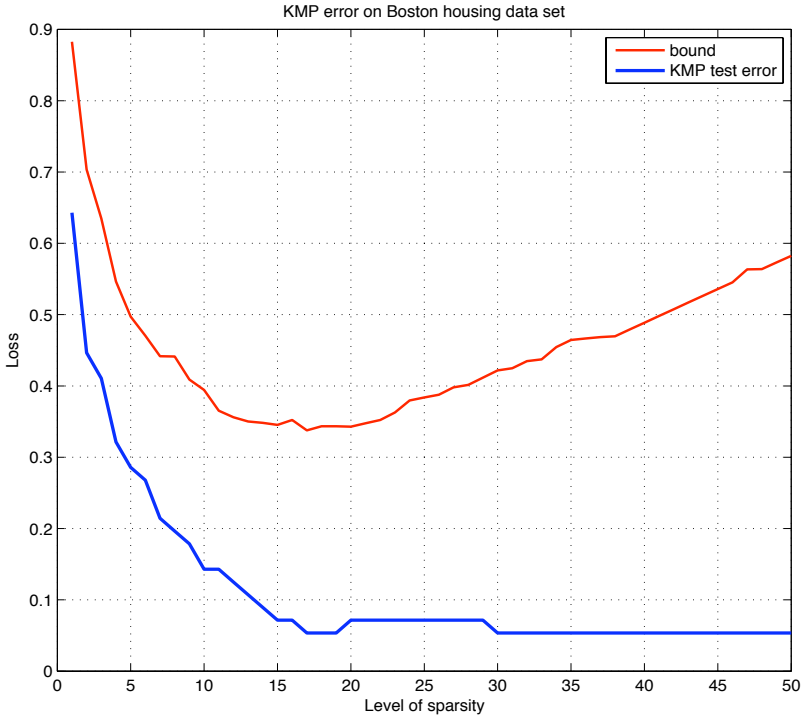

Figure: Boston housing data set: Plot for KMP bound vs KMP test error using Gaussian kernel. Bound scaled by factor of 5.

Extensions: kernel canonical correlation analysis

- Assume two views $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^m$ and $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^m$ of the same data.
- Projections $P_x : \mathbf{x} \mapsto \mathbf{x}^\top \mathbf{w}_x$ and $P_y : \mathbf{y} \mapsto \mathbf{y}^\top \mathbf{w}_y$.
- The idea of canonical correlation analysis (CCA) is to maximise the correlation $corr(P_x(\mathbf{X}), P_y(\mathbf{Y}))$ between the data in their corresponding projection space.

Hence maximise the following:

$$\begin{aligned} \lambda_{x,y} &= \max_{\mathbf{w}_x, \mathbf{w}_y} \frac{\mathbf{w}_x^\top \mathbf{X}\mathbf{Y}^\top \mathbf{w}_y}{\sqrt{\mathbf{w}_x^\top \mathbf{X}\mathbf{X}^\top \mathbf{w}_x \mathbf{w}_y^\top \mathbf{Y}\mathbf{Y}^\top \mathbf{w}_y}} \\ &= \max_{\mathbf{w}_x, \mathbf{w}_y} \frac{\mathbf{w}_x^\top \mathbf{C}_{xy} \mathbf{w}_y}{\sqrt{\mathbf{w}_x^\top \mathbf{C}_{xx} \mathbf{w}_x \mathbf{w}_y^\top \mathbf{C}_{yy} \mathbf{w}_y}}, \end{aligned} \tag{1}$$

where $\lambda_{x,y}$ are the eigenvalues corresponding to $\mathbf{w}_x$ and $\mathbf{w}_y$.

$$\mathbf{w}_x = \mathbf{X}[:,\mathbf{i}]\tilde{\boldsymbol{\alpha}}_x, \quad (2)$$

$$\mathbf{w}_y = \mathbf{Y}[:,\mathbf{i}]\tilde{\boldsymbol{\alpha}}_y. \quad (3)$$

Substituting these two expressions into the CCA problem of Equation (1) we get:

$$\max_{\mathbf{i},\tilde{\boldsymbol{\alpha}}_x,\tilde{\boldsymbol{\alpha}}_y} \frac{\tilde{\boldsymbol{\alpha}}_x^\top \mathbf{X}[:,\mathbf{i}]^\top \mathbf{X}\mathbf{Y}^\top \mathbf{Y}[:,\mathbf{i}]\tilde{\boldsymbol{\alpha}}_y}{\sqrt{\tilde{\boldsymbol{\alpha}}_x^\top \mathbf{X}[:,\mathbf{i}]^\top \mathbf{X}\mathbf{X}^\top \mathbf{X}[:,\mathbf{i}]\tilde{\boldsymbol{\alpha}}_x \tilde{\boldsymbol{\alpha}}_y^\top \mathbf{Y}[:,\mathbf{i}]^\top \mathbf{Y}\mathbf{Y}^\top \mathbf{Y}[:,\mathbf{i}]\tilde{\boldsymbol{\alpha}}_y}}.$$

Let $\mathbf{K}_x[:,\mathbf{i}]^\top = \mathbf{X}[:,\mathbf{i}]^\top \mathbf{X}$ and $\mathbf{K}_y[:,\mathbf{i}]^\top = \mathbf{Y}[:,\mathbf{i}]^\top \mathbf{Y}$, to get sparse KCCA as:

$$\tilde{\lambda}_{x,y} = \max_{\mathbf{i},\tilde{\boldsymbol{\alpha}}_x,\tilde{\boldsymbol{\alpha}}_y} \frac{\tilde{\boldsymbol{\alpha}}_x^\top \mathbf{K}_x[:,\mathbf{i}]^\top \mathbf{K}_y[:,\mathbf{i}]\tilde{\boldsymbol{\alpha}}_y}{\sqrt{\tilde{\boldsymbol{\alpha}}_x^\top \mathbf{K}_x^2[\mathbf{i},\mathbf{i}]\tilde{\boldsymbol{\alpha}}_x \tilde{\boldsymbol{\alpha}}_y^\top \mathbf{K}_y^2[\mathbf{i},\mathbf{i}]\tilde{\boldsymbol{\alpha}}_y}},$$

where $\tilde{\boldsymbol{\alpha}}_x$ and $\tilde{\boldsymbol{\alpha}}_y$ are sparse dual eigenvectors.

Matching pursuit kernel canonical correlation analysis

Following MPKPCA and KMP we repeat the following two-step procedure k times.

- Maximise:

$$i^* = \arg\max_i \frac{\mathbf{e}_i^\top \mathbf{K}_x^\top \mathbf{K}_y \mathbf{e}_i}{\sqrt{\mathbf{e}_i^\top \mathbf{K}_x^2 \mathbf{e}_i \mathbf{e}_i^\top \mathbf{K}_y^2 \mathbf{e}_i}} \qquad i = 1, \ldots, m,$$

- Deflate (orthogonlise):

$$\begin{aligned} \mathbf{K}_x &= \left(\mathbf{I} - \frac{\mathbf{K}_x[:, i^*] \mathbf{K}_x[:, i^*]^\top}{\mathbf{K}_x[:, i^*]^\top \mathbf{K}_x[:, i^*]} \right) \mathbf{K}_x, \\ \mathbf{K}_y &= \left(\mathbf{I} - \frac{\mathbf{K}_y[:, i^*] \mathbf{K}_y[:, i^*]^\top}{\mathbf{K}_y[:, i^*]^\top \mathbf{K}_y[:, i^*]} \right) \mathbf{K}_y. \end{aligned}$$

Experiments for MPKCCA

Given a paired example $(\mathbf{x}_j, \mathbf{y}_j)$ the difference $\|P_x(\phi(x_j)) - P_y(\phi(y_j))\|$ between the two projections is the reconstruction error of MPKCCA.

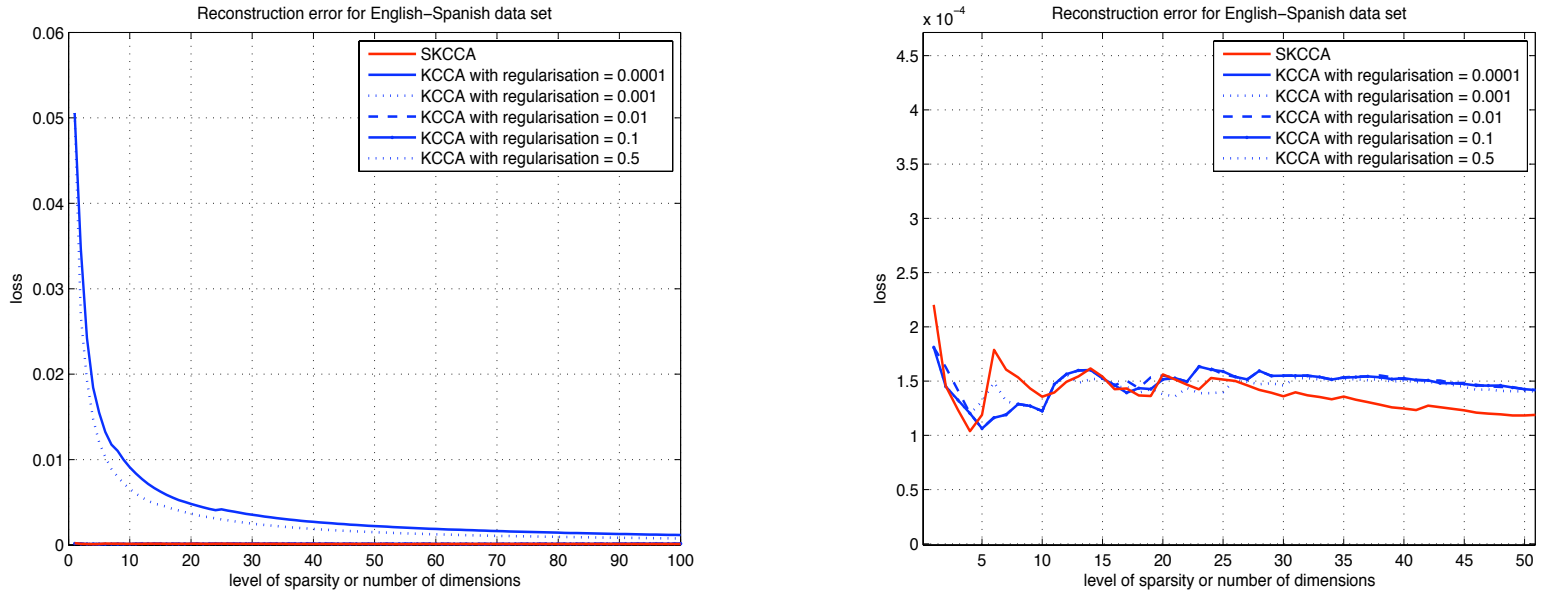

Figure: English-Spanish dat set. Left: plot for all dimensions. Right: a closer look at the lhs plot for the first 50 dimensions or basis vectors.

Bound for MPKPCA

We can apply the KMP bound to the individual dimensions (by viewing the reconstruction loss as a regression loss).

Theorem

Fix $\alpha \in \mathbb{R}$, $\alpha > 0$. Let $\mathcal{A}$ be the MPKCCA algorithm, m the size of the paired training sets $S^{\mathcal{X}\times\mathcal{Y}}$ and k the cardinality of the set $\mathbf{i}$. Let $S^{\mathcal{X}\times\mathcal{Y}}$ be reordered so that the last $m-k$ points are outside of the set $\mathbf{i}$ and define $t = \sum_{i=m-k}^{m} \mathbb{I}(|f_x(\mathbf{x}_i) - f_y(\mathbf{y}_i)| > \alpha)$ to be the number of errors for those points in $S^{\mathcal{X}\times\mathcal{Y}} \setminus S_{\mathbf{i}}^{\mathcal{X}\times\mathcal{Y}}$. Then with probability $1-\delta$ over the generation of the paired training sets $S^{\mathcal{X}\times\mathcal{Y}}$ the expected loss $\mathcal{E}[\ell(\cdot)]$ of algorithm $\mathcal{A}$ can be bounded by,

$$\mathcal{E}[\ell(\mathcal{A}(S))] \leq \frac{2}{m-k-t}\left[(k+1)\log\left(\frac{4e(m-k-t)}{k+1}\right)\right.$$
$$\left. + k\log\left(\frac{em}{k}\right) + t\log\left(\frac{e(m-k)}{t}\right) + \log\left(\frac{2m^2}{\delta}\right)\right].$$

MPKCCA bound experiments

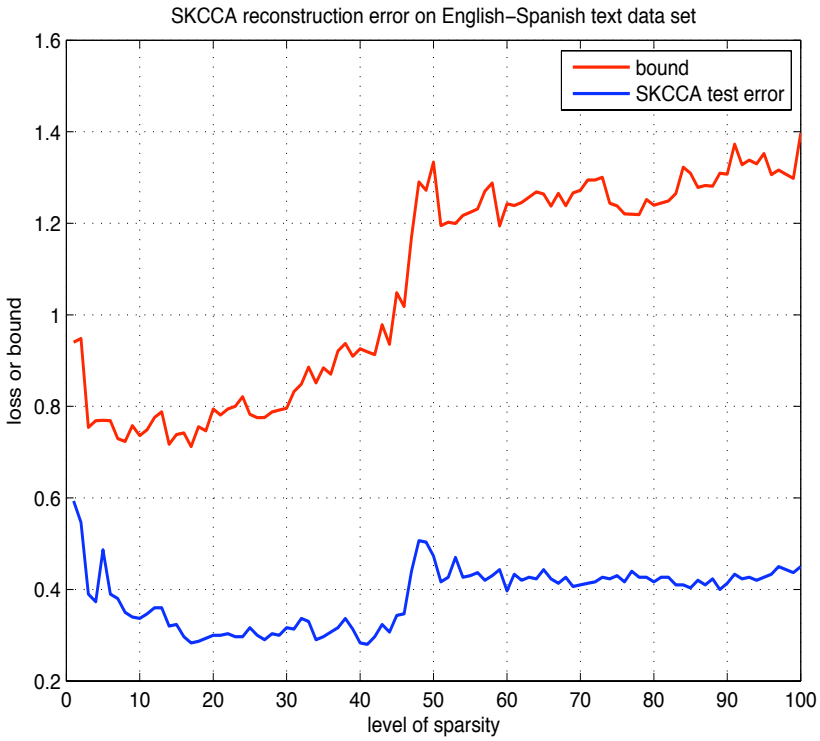

Figure: English-Spanish data set. Bound plot for sparse KCCA using 1-dimension with $\alpha = 0.0095$ and scaled down by a factor of 5.

Take home messages

- A general two-step algorithm of
  - maximisation,
  - deflation,

  to find $\mathbf{i}$
- Matching pursuit is a meta-scheme for learning algorithms that can be applied to various problems such as regression, dimensionality reduction and classification.

Conclusions

- We bounded the reconstruction error and generalisation error of MPKPCA and KMP.
- The MPKPCA bound was a compression scheme.
- The KMP bound was an amalgamation of compression schemes and VC theory.
- KMP bound unaffected by the ambient dimension of the feature space (unlike traditional VC bounds).
- Gave a general 2-step matching pursuit technique, with an application to KCCA.
- Can also be applied to kernel fisher discriminant analysis.
- How to achieve tighter bounds for KMP?
- How to achieve a bound for ALL dimensions of MPKCCA? (without resorting to a union bound)