Sparse PCA and CCA Revisited

Abd-Krim Seghouane, Navid Shokouhi and Inge Koch

Department of Electrical and Electronic Engineering
The University of Melbourne, Australia
School of Mathematical Statistics, University of Western Australia

December 6, 2019
Introduction

- Let $\mathbf{X}$ denote an $n \times p$ data matrix of rank $q \leq \min(n, p)$.
- $n$ is the number of data samples, $p$ the number of variables and for all $i = 1, \ldots, n$, $\text{cov}(\mathbf{x}_i) = \mathbf{\Sigma}$

$$
\mathbf{\Sigma} = \sum_{i=1}^{q} \lambda_i \mathbf{v}_i \mathbf{v}_i^\top
$$

- Replacing the original variables with linear combinations of the original variables $\mathbf{Xv}_k$, $k = 1, \ldots, q$ known as the PCs

$$
\mathbf{v}_k = \arg \max_{\mathbf{v}} \text{var} (\mathbf{Xv})
$$

subject to $\mathbf{v}_k^\top \mathbf{v}_k = 1$

and $\mathbf{v}_k^\top \mathbf{v}_j = 0$ for $j \neq k$

- $\mathbf{v}_k = (v_{k1}, \ldots, v_{kp})^\top$ is the $p \times 1$ loading vector and the projection of the data $\mathbf{Xv}_k$ is the $k^{th}$ principal component.
Introduction

- In many real world applications, sparse loading vectors have physical interpretation.

**Figure 1**: Decomposition of face data using sparse loading vectors (Jenatton et al. 2010)
Existing methods

- Add sparsity constraint to the optimization problem

\[
\mathbf{v}_k = \arg \max_{\mathbf{v}} \var (\mathbf{Xv})
\]

subject to

\[
\mathbf{v}_k^\top \mathbf{v}_k = 1
\]

\[
\mathbf{v}_k^\top \mathbf{v}_j = 0 \text{ for } j \neq k
\]

\[
\sum_{j=1}^{p} |v_{ij}| \leq t
\]

Existing methods

- Transforming the original PCA problem into

\[
\left(\hat{\alpha}, \hat{\beta}\right) = \arg \min_{\alpha, \beta} \sum_{i=1}^{n} \| x_i - \alpha \beta^\top x_i \|^2 + \lambda \| \beta \|^2
\]
subject to \( \| \alpha \|^2 = 1 \)

- \( \hat{\beta} \propto v_1 \)

- \( A \) and \( B \) are \( p \times d \)

\[
\left(\hat{A}, \hat{B}\right) = \arg \min_{A, B} \sum_{i=1}^{n} \| x_i - AB^\top x_i \|^2 + \lambda \sum_{j=1}^{d} \| \beta_j \|^2 + \sum_{j=1}^{d} \lambda_j \| \beta_j \|_1
\]
subject to \( A^\top A = I_d \)

- where \( | \beta |_1 = \sum_{i=1}^{p} | \beta_i | \)

Existing methods

- Penalized matrix decomposition

\[ \min_{\mathbf{u}, \mathbf{b}} \| \mathbf{X} - \mathbf{u} \mathbf{b}^\top \|_F^2 + \sum_{i=1}^{p} \alpha_i | b_i | \]

subject to \( \| \mathbf{u} \|_2 = 1 \).

- \( \mathbf{b} = \lambda \mathbf{v} \)


Existing methods

▶ Equivalently, PMD is:

$$\hat{v} = \arg \max_v v^\top X^\top X v$$

subject to

$$\|v\|_2 \leq 1,$$

$$|v|_1 \leq c$$

▶ $\|v\|_2 \leq 1$ is a convex relaxation of $\|v\|^2 = 1$.

Introductory Example: data generation model

Consider a dataset of $p$-dimensional vectors $\mathbf{x} \in \mathbb{R}^p$, where $p$ is the number of variables:

$$\mathbf{x} = \mathbf{\Gamma} \mathbf{W} \mathbf{z} + \epsilon. \quad (1)$$

- $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$.
- $\mathbf{W}$ contains $q$ orthonormal basis vectors in $\mathbb{R}^p$.
- $\epsilon$ represents random noise with $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_p)$.

We introduce sparsity via the $p \times p$ diagonal matrix $\mathbf{\Gamma}$, which zeros out some entries of $\mathbf{Wz}$.

$$\mathbf{\Gamma} = \text{diag}(1, 1, 1, 1, 0, 0, \ldots, 0)$$
Table 1: 5 samples of generated data $\mathbf{x}$ for $p = 10$

<table>
<thead>
<tr>
<th>data(first 5 samples)</th>
<th>0.129</th>
<th>-0.479</th>
<th>-0.807</th>
<th>-0.347</th>
<th>0.220</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.749</td>
<td>0.221</td>
<td>0.496</td>
<td>-0.618</td>
<td>-0.871</td>
</tr>
<tr>
<td></td>
<td>-0.112</td>
<td>-0.733</td>
<td>-0.717</td>
<td>0.3898</td>
<td>-0.898</td>
</tr>
<tr>
<td></td>
<td>-0.053</td>
<td>0.146</td>
<td>0.059</td>
<td>0.0036</td>
<td>-0.023</td>
</tr>
<tr>
<td></td>
<td>-0.245</td>
<td>0.091</td>
<td>-0.090</td>
<td>-0.109</td>
<td>-0.030</td>
</tr>
<tr>
<td></td>
<td>0.016</td>
<td>0.151</td>
<td>0.037</td>
<td>0.0375</td>
<td>0.1639</td>
</tr>
<tr>
<td></td>
<td>-0.083</td>
<td>-0.045</td>
<td>-0.126</td>
<td>0.0170</td>
<td>-0.189</td>
</tr>
<tr>
<td></td>
<td>0.008</td>
<td>-0.114</td>
<td>0.022</td>
<td>0.1139</td>
<td>0.1318</td>
</tr>
<tr>
<td></td>
<td>0.131</td>
<td>-0.071</td>
<td>-0.041</td>
<td>0.1657</td>
<td>0.0532</td>
</tr>
<tr>
<td></td>
<td>-0.014</td>
<td>-0.033</td>
<td>-0.102</td>
<td>0.0338</td>
<td>0.0601</td>
</tr>
</tbody>
</table>
Introductory Example

- PCA loading vectors are non-sparse.
- Information regarding the important directions in the feature space is lost.
- PCA has **low interpretability**.

Table 2: Decomposition using PCA

<table>
<thead>
<tr>
<th>loading vectors</th>
<th>$\mathbf{v}_1$</th>
<th>$\mathbf{v}_2$</th>
<th>$\mathbf{v}_3$</th>
<th>$\mathbf{v}_4$</th>
<th>$\mathbf{v}_5$</th>
<th>$\mathbf{v}_6$</th>
<th>$\mathbf{v}_7$</th>
<th>$\mathbf{v}_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>-0.343</td>
<td>-0.030</td>
<td>-0.364</td>
<td>-0.860</td>
<td>0.073</td>
<td>-0.037</td>
<td>-0.029</td>
<td>0.030</td>
</tr>
<tr>
<td></td>
<td>0.239</td>
<td>0.301</td>
<td>0.798</td>
<td>-0.451</td>
<td>-0.057</td>
<td>0.004</td>
<td>-0.044</td>
<td>-0.075</td>
</tr>
<tr>
<td></td>
<td>-0.906</td>
<td>0.112</td>
<td>0.350</td>
<td>0.210</td>
<td>0.001</td>
<td>-0.015</td>
<td>0.002</td>
<td>0.021</td>
</tr>
<tr>
<td></td>
<td>-0.021</td>
<td>-0.943</td>
<td>0.304</td>
<td>-0.092</td>
<td>-0.088</td>
<td>-0.019</td>
<td>-0.036</td>
<td>0.010</td>
</tr>
<tr>
<td></td>
<td>-0.017</td>
<td>0.024</td>
<td>-0.056</td>
<td>0.030</td>
<td>-0.145</td>
<td>0.140</td>
<td>-0.756</td>
<td>-0.204</td>
</tr>
<tr>
<td></td>
<td>0.026</td>
<td>0.066</td>
<td>-0.003</td>
<td>0.022</td>
<td>-0.367</td>
<td>-0.614</td>
<td>-0.279</td>
<td>0.435</td>
</tr>
<tr>
<td></td>
<td>-0.008</td>
<td>-0.013</td>
<td>0.027</td>
<td>0.025</td>
<td>0.319</td>
<td>0.240</td>
<td>-0.153</td>
<td>-0.368</td>
</tr>
<tr>
<td></td>
<td>-0.007</td>
<td>0.016</td>
<td>-0.005</td>
<td>-0.021</td>
<td>-0.215</td>
<td>-0.175</td>
<td>0.561</td>
<td>-0.237</td>
</tr>
<tr>
<td></td>
<td>0.015</td>
<td>-0.036</td>
<td>0.015</td>
<td>0.042</td>
<td>0.479</td>
<td>-0.715</td>
<td>-0.095</td>
<td>-0.450</td>
</tr>
<tr>
<td></td>
<td>-0.057</td>
<td>0.017</td>
<td>-0.108</td>
<td>-0.013</td>
<td>-0.671</td>
<td>-0.037</td>
<td>0.001</td>
<td>-0.606</td>
</tr>
</tbody>
</table>
Introductory Example

- Using SPCA (Zou et al. 2006).
- The first PC gives the correct sparsity pattern.

Table 3: Decomposition using SPCA (PMD is similar).

<table>
<thead>
<tr>
<th>loading vectors</th>
<th>$\mathbf{v}_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPCA</td>
<td>-0.340 0.236 -0.910 -0.004 0 0 0 0 0</td>
</tr>
</tbody>
</table>
As we extract more PCs \((v_2, \ldots, v_8)\), the sparsity pattern varies.

<table>
<thead>
<tr>
<th>SPCA</th>
<th>(v_1)</th>
<th>(v_2)</th>
<th>(v_3)</th>
<th>(v_4)</th>
<th>(v_5)</th>
<th>(v_6)</th>
<th>(v_7)</th>
<th>(v_8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.340</td>
<td>0.032</td>
<td>-0.204</td>
<td>0.901</td>
<td>0.072</td>
<td>0</td>
<td>0.016</td>
<td>-0.030</td>
<td></td>
</tr>
<tr>
<td>0.236</td>
<td>-0.302</td>
<td>0.892</td>
<td>0.358</td>
<td>-0.040</td>
<td>0</td>
<td>0.043</td>
<td>0.075</td>
<td></td>
</tr>
<tr>
<td>-0.910</td>
<td>-0.100</td>
<td>0.284</td>
<td>-0.239</td>
<td>0.010</td>
<td>-0.010</td>
<td>0</td>
<td>-0.017</td>
<td></td>
</tr>
<tr>
<td>-0.004</td>
<td>0.945</td>
<td>0.288</td>
<td>0.048</td>
<td>-0.054</td>
<td>-0.066</td>
<td>0.027</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>-0.003</td>
<td>0</td>
<td>0</td>
<td>-0.028</td>
<td>0</td>
<td>0.767</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>-0.061</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-0.900</td>
<td>0.117</td>
<td>-0.337</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.032</td>
<td>0.231</td>
<td>0.080</td>
<td>0.293</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-0.063</td>
<td>-0.622</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.028</td>
<td>0</td>
<td>0</td>
<td>0.805</td>
<td>-0.215</td>
<td>0.024</td>
<td>0.485</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>-0.017</td>
<td>0</td>
<td>0</td>
<td>-0.583</td>
<td>-0.285</td>
<td>0</td>
<td>0.748</td>
<td></td>
</tr>
</tbody>
</table>
Some studies explicitly estimate $\Gamma$. Results in an intricate Expectation Maximization approach. Requires estimating a large number of parameters. Computationally expensive.


Proposed Method

while enforcing sparsity on the entries of $b,$

adaptively penalize loadings to preserve sparsity pattern.

We propose:

$$\text{arg min}_{U,B} \left\| X - UB^\top \right\|_F^2 + \sum_{i=1}^{p} \alpha_i \| b_i \|_2$$

subject to $U^\top U = I_q$

where

- $b_i$ is the $i^{th}$ column of the $q \times p$ matrix $B^\top.$
- Sometimes $\alpha_i \leftarrow \alpha_i \sqrt{q}$ is used to rescale the penalty with respect to the dimensionality of $b_i.$
Proposed Method

Numerical Solution:

- Keeping $\mathbf{U}$ fixed, optimizing $\mathbf{B}$:

$$b_i = \left( 1 - \frac{\alpha_i \sqrt{q}}{2 \| \mathbf{U}^\top \mathbf{x}_i \|_2} \right)_+ \mathbf{U}^\top \mathbf{x}_i$$

where the operator $(.)_+$ is set to 0 when

$$\frac{2}{\alpha_i \sqrt{q}} \| \mathbf{U}^\top \mathbf{x}_i \|_2 < 1.$$

- Given $\mathbf{B}$, optimization with respect to $\mathbf{U}$ is:

$$\min_{\mathbf{U}} \left\| \mathbf{X} - \mathbf{U} \mathbf{B}^\top \right\|_F^2 \text{ subject to } \mathbf{U}^\top \mathbf{U} = \mathbf{I}_q$$

This is an orthogonal Procrustes problem, with the solution:

$$\mathbf{U} = \mathbf{U} \tilde{\mathbf{V}}^\top,$$

where

- $\mathbf{U}$ and $\tilde{\mathbf{V}}$ are obtained from the SVD of $\mathbf{X} \mathbf{B} = \mathbf{U} \Lambda \tilde{\mathbf{V}}$
- $\mathbf{U}$ is $n \times q$
- and $\tilde{\mathbf{V}}$ is $p \times q$. 
For $q = 1$ PCs:

$$\min_{u, b} \left\| \mathbf{X} - \mathbf{u} \mathbf{b}^\top \right\|_F^2 + \sum_{i=1}^{p} \alpha_i \| b_i \|
$$

s.t. $\| u \|_2 = 1$.

- SPCA/PMD are a special case of the proposed approach.
- Generalization is due to using different $\alpha_i$ per entry of $\mathbf{b}$. 
Table 5: Description of the proposed adaptive sparse PCA algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Given:</strong> $X$, $q$, $\alpha$, $\epsilon$.</td>
</tr>
<tr>
<td><strong>Tuning parameters selection</strong></td>
</tr>
<tr>
<td>Take $B$ as the first $q$ right singular vectors times the first $q$ singular values of $X$.</td>
</tr>
<tr>
<td>Compute the vector of tuning parameters $\alpha_i = \frac{\alpha}{|b_i|_2}$ for $i = 1, \ldots, p$.</td>
</tr>
<tr>
<td><strong>While</strong> $|U_j - U_{j-1}|_F &gt; \epsilon$</td>
</tr>
<tr>
<td><strong>Update:</strong> $B^\top$ per column</td>
</tr>
<tr>
<td>For $j=1$ to $p$</td>
</tr>
<tr>
<td>$b_i = \left(1 - \frac{\alpha_i \sqrt{q}}{2|U_{j-1}^\top x_i|<em>2}\right) U</em>{j-1}^\top x_i$</td>
</tr>
<tr>
<td><strong>Update:</strong> $U_j = \tilde{U}\tilde{V}^\top$ using the SVD of $XB = \tilde{U}\Lambda\tilde{V}^\top$</td>
</tr>
<tr>
<td><strong>Output:</strong> $U$, $B$</td>
</tr>
</tbody>
</table>
Back to the example:

**Table 6: Decomposition using proposed method**

<table>
<thead>
<tr>
<th>loading vectors</th>
<th>( \mathbf{v}_1 )</th>
<th>( \mathbf{v}_2 )</th>
<th>( \mathbf{v}_3 )</th>
<th>( \mathbf{v}_4 )</th>
<th>( \mathbf{v}_5 )</th>
<th>( \mathbf{v}_6 )</th>
<th>( \mathbf{v}_7 )</th>
<th>( \mathbf{v}_8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>-0.805</td>
<td>0.059</td>
<td>-0.660</td>
<td>1.121</td>
<td>0.029</td>
<td>-0.014</td>
<td>0.008</td>
<td>-0.007</td>
</tr>
<tr>
<td></td>
<td>0.592</td>
<td>-0.627</td>
<td>1.524</td>
<td>0.620</td>
<td>-0.024</td>
<td>0.001</td>
<td>0.013</td>
<td>0.020</td>
</tr>
<tr>
<td></td>
<td>-2.404</td>
<td>-0.252</td>
<td>0.717</td>
<td>-0.310</td>
<td>0.000</td>
<td>-0.006</td>
<td>-0.001</td>
<td>-0.006</td>
</tr>
<tr>
<td></td>
<td>-0.053</td>
<td>2.028</td>
<td>0.598</td>
<td>0.130</td>
<td>-0.039</td>
<td>-0.008</td>
<td>0.011</td>
<td>-0.003</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
CCA finds the **maximum correlation** between **two random vectors**:

\[
(u_1, v_1) = \arg \max_{u, v} \quad u^\top \Sigma_{xy} v
\]

subject to

\[
u^\top \Sigma_{xx} u = 1, \quad v^\top \Sigma_{yy} v = 1
\]

where

\[
\Sigma_{xy} = E[xy^\top]
\]

\[
\Sigma_{xx} = E[xx^\top] \quad \text{and} \quad \Sigma_{yy} = E[yy^\top]
\]

\[x \in \mathbb{R}^p \quad \text{and} \quad y \in \mathbb{R}^q\]
Introduction

$r \leq \min(p, q)$ loading vectors can be extracted:

- for $i = 1, \ldots, r$ the vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ are estimated.
- $\mathbf{u}_i^\top \mathbf{u}_j = 0$ for $j = 1, \ldots, i - 1$.

$$(\hat{\mathbf{U}}, \hat{\mathbf{V}}) = \arg \max_{\mathbf{U}, \mathbf{V}} \text{tr}(\mathbf{U}^\top \Sigma_{xy} \mathbf{V})$$
subject to
$$\mathbf{U}^\top \Sigma_{xx} \mathbf{U} = \mathbf{I}_r,$$
$$\mathbf{V}^\top \Sigma_{yy} \mathbf{V} = \mathbf{I}_r$$

where

$\text{tr}(.)$ is the trace operator.

$\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_r]$
$\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_r]$
Introduction

In practice:

\[
\frac{1}{n} X^\top Y \rightarrow \Sigma_{xy} \\
\frac{1}{n} X^\top X \rightarrow \Sigma_{xx} \\
\frac{1}{n} Y^\top Y \rightarrow \Sigma_{yy}
\]

where

- \( n \) is the number of samples.
- \( X \) and \( Y \) are \( n \times p \) and \( n \times q \) matrices.
- number of samples should be the same for \( X \) and \( Y \)
- we can alternatively solve the equivalent problem:

\[
(\hat{U}, \hat{V}) = \arg \min_{U, V} \| XU - YV \|_F^2
\]

instead of \( \max_{U, V} \) \( \text{tr}(U^\top XY^\top V) \).
Penalized Matrix Decomposition for Sparse CCA:

\[ (\hat{u}, \hat{v}) = \arg \max_{u, v} \quad u^\top X^\top Y v \]

subject to \[ u^\top X^\top X u \leq 1, \quad v^\top Y^\top Y v \leq 1, \]
\[ |u|_1 \leq c_1, \quad |v|_1 \leq c_2, \]

In (Witten et al. 2009), \( X^\top X \) and \( Y^\top Y \) are assumed to be diagonal.

Introductory Example: data generation model

Consider the sparse data generating model:

\[ x = \Gamma_x W_x z + \epsilon_x \]
\[ y = \Gamma_y W_y z + \epsilon_y \]

where

- \( \Gamma_x \) and \( \Gamma_y \) are diagonal matrices that enforce sparsity.
- \( W_x \) and \( W_y \) are orthogonal matrices.
- \( z \sim \mathcal{N}(0, I_r) \).
- \( \epsilon_x \) and \( \epsilon_y \) are random Gaussian noise vectors.
Introductory Example

- $n = 300$ samples of $p = 10$ and $q = 11$ dimensional vectors.
- The number of common components (i.e., dimension of $z$) is $r = 4$.
- Noise variance for both signals is $\sigma^2 = 0.1$.
- Sparsity patterns for $x$ and $y$ are:

\[
\Gamma_x = \text{diag}(1, 1, 1, 1, 0, \ldots, 0) \\
\Gamma_y = \text{diag}(0, \ldots, 0, 1, 1, 1, 1, 1, 1)
\]
Introductory Example

Using PMD:

Table 7: loading vectors of $\mathbf{X}$ from PMD

| $\mathbf{U}$ | 0.83  
|             | -0.08 |
|             | 0.4   |
|             | 0.38  |
|             | 0.01  |
|             | 0     |
|             | 0     |
|             | 0     |
|             | 0     |
Remaining loading vector matters have varying sparsity pattern.

**Table 8: loading vectors of $X$ from PMD**

<table>
<thead>
<tr>
<th></th>
<th>0.83</th>
<th>0.02</th>
<th>-0.56</th>
<th>0.04</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>-0.08</td>
<td>0.33</td>
<td>0.08</td>
<td>0.97</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>-0.66</td>
<td>0.6</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>0.38</td>
<td>0.68</td>
<td>0.56</td>
<td>-0.2</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>0</td>
<td>-0.02</td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>-0.01</td>
<td>-0.04</td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0.02</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>-0.01</td>
<td>0</td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Similarly for $\mathbf{Y}$

**Table 9: loading vectors of $\mathbf{Y}$ from PMD**

<table>
<thead>
<tr>
<th>$\mathbf{V}$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>-0.1</th>
<th>-0.14</th>
<th>0.68</th>
<th>-0.15</th>
<th>0.69</th>
</tr>
</thead>
</table>
Introductory Example

Remaining loading vector matters have varying sparsity pattern.

Table 10: loading vectors of $\mathbf{Y}$ from PMD

<table>
<thead>
<tr>
<th>$\mathbf{V}$</th>
<th>0</th>
<th>-0.02</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>-0.01</td>
<td>-0.03</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>-0.1</td>
<td>0.41</td>
<td>0.37</td>
<td>0.81</td>
<td></td>
</tr>
<tr>
<td>-0.14</td>
<td>-0.62</td>
<td>0.76</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0.68</td>
<td>0.01</td>
<td>0</td>
<td>0.32</td>
<td></td>
</tr>
<tr>
<td>-0.15</td>
<td>0.67</td>
<td>0.42</td>
<td>-0.38</td>
<td></td>
</tr>
<tr>
<td>0.69</td>
<td>0.05</td>
<td>0.32</td>
<td>-0.33</td>
<td></td>
</tr>
</tbody>
</table>
Proposed Method

Our proposed method uses adaptive weights for the sparsity penalty:

$$\min_{U, V} \| XU - YV \|_F^2 + \sum_{i=1}^{p} \alpha_i \| u^i \|_2 + \sum_{j=1}^{q} \beta_j \| v^j \|_2$$  \hspace{1cm} (3)

where

- $u^i$ represents the $i^{th}$ row of $U$,
- $v^j$ represents the $j^{th}$ row of $V$. 

The computation of $U$ and $V$ are obtained by a block coordinate descent method where each of the variables are computed row by row using the closed form solutions.

$$
v^j = \frac{1}{y_j^\top y_j} \left( 1 - \frac{1}{\|y_j^\top E_j\|_2} \right) y_j^\top E_j + (\cdot)_+ = \max(0, \cdot)
$$

where $y_j$ is the $j^{th}$ column of $Y$, $E_j = XU - \sum_{i=1, i \neq q}^q y_i v_i$ and $(\cdot)_+ = \max(0, \cdot)$
Proposed Method

Table 11: loading vectors of $X$ - Proposed method

<table>
<thead>
<tr>
<th></th>
<th>0.028</th>
<th>-0.009</th>
<th>0.019</th>
<th>0.009</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.014</td>
<td>-0.016</td>
<td>0.009</td>
<td>0.002</td>
<td></td>
</tr>
<tr>
<td>-0.026</td>
<td>0.029</td>
<td>-0.015</td>
<td>0.009</td>
<td></td>
</tr>
<tr>
<td>-0.026</td>
<td>0.011</td>
<td>-0.017</td>
<td>0.006</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Proposed Method

Table 12: loading vectors of \( Y \) - Proposed method

<table>
<thead>
<tr>
<th>( \mathbf{V} )</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0.018</th>
<th>-0.007</th>
<th>0.012</th>
<th>0.008</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.028</td>
<td>-0.006</td>
<td>0.019</td>
<td>0.006</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-0.024</td>
<td>0.022</td>
<td>-0.015</td>
<td>0.003</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-0.016</td>
<td>0.017</td>
<td>-0.01</td>
<td>-0.002</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-0.025</td>
<td>0.009</td>
<td>-0.017</td>
<td>-0.008</td>
</tr>
</tbody>
</table>
Proposed Method for Sparse CCA

Applications
Applications

- **Sparse PCA**
  - Hand-written digit recognition.
  - Blind source separation of single-subject fMRI data.
MNIST digit recognition

- MNIST is a popular benchmark for evaluating PCA.
- The goal is to classify a given image of a handwritten digit.
- contaminate background with noise.
- Sparse PCA can be used to select a subset of the pixels.

**Figure 2:** SPCA, PMD, SSPCA, and Proposed. backgrounds: clean (left), uniform (middle), and non-uniform (right) noise.
**MNIST digit recognition**

- Using sparse PCA as a pre-processing step for nearest-neighbor classification of digits.
- As the number PCs increases, the proposed method (black) is not confused into selecting background pixels.

**MNIST classification**
fMRI source Separation

- Single-subject functional connectivity analysis using function Magnetic Resonance Imaging (fMRI) data.
- Uses Independent component Analysis (ICA).

\[ X_{\text{voxels} \times \text{timepoints}} = \text{Spatial} \times \text{Temporal} \]

- One of the issues of using ICA for fMRI data is high-dimensionality, which
  - Negatively impacts reproducability of ICA results.
  - Is computationally expensive.
- fMRI researchers use PCA as a dimension reduction stage.
- However, the issue with PCA is that it distorts the spatial arrangement of brain signals.

Proposed solution:
- Using sparse PCA can focus only on local brain regions that have valuable information.
Spatial maps are reconstructed by calculating the correlation of the resulting $q$ time series with the original data.
fMRI source Separation: Simulated data

Extracting spatial maps for simulated fMRI-like data:

\[ X_{p \times n} = S_{p \times q} T_{q \times n} + \xi \]  \hspace{1cm} (4)

where

- **S**: \( q \) spatial maps of dimension \( p \times 1 \).
- **T**: \( q \) time series of dimension \( n \times 1 \) calculated from a independent Gaussian processes.
- **\( \xi \)**: White Gaussian noise with variance \( \sigma^2 \).
fMRI source Separation: Simulated data

- Left half shows first $q = 4$ loading vectors of PCA.
- ICA (right half) works better when sparsity is preserved across components.
fMRI source Separation: Real data

- For real fMRI data.
- Task-related spatial maps for finger-tapping experiment.
- Subject asked to tap right finger in a block-event paradigm.
- desired activation: Motor cortex.
fMRI source Separation: Real data

- Estimated spatial activations:

![Spatial maps corresponding to motor cortex.](image)

**Figure 3:** Spatial maps corresponding to motor cortex.

- Less spurious results from using proposed sparse PCA.
Figure 4: Time-courses for finger-tapping experiment. Blue: ground-truth, Red: estimated time-course using ICA.
References


Code available at: https://github.com/idnavid/sparse_PCA