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 $i = 1, \ldots, n$, $\mathrm{cov}(\mathbf{x}_i) = \mathbf{\Sigma}$

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

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

$$\begin{aligned} \mathbf{v}_k = \quad & \arg\max_{\mathbf{v}} & & \mathrm{var}\left(\mathbf{X}\mathbf{v}\right) \\ & \text{subject to} & & \mathbf{v}_k^\top \mathbf{v}_k = 1 \\ & \text{and} & & \mathbf{v}_k^\top \mathbf{v}_j = 0 \text{ for } j \neq k \end{aligned}$$

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

Introduction

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

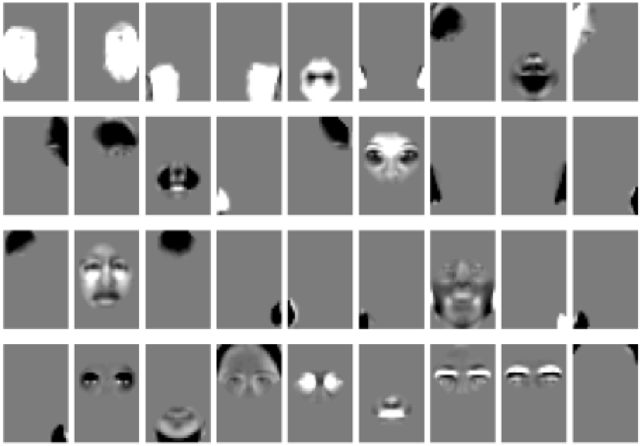


Figure 1: Decomposition of face data using sparse loading vectors (Jenatton et al. 2010)

Existing methods

- Add sparsity constraint to the optimization problem

$$
\begin{aligned}
\mathbf{v}_k = \quad & \arg\max_{\mathbf{v}} & & \operatorname{var}(\mathbf{X}\mathbf{v}) \\
& \text{subject to} & & \mathbf{v}_k^\top \mathbf{v}_k = 1 \\
& & & \mathbf{v}_k^\top \mathbf{v}_j = 0 \text{ for } j \neq k \\
& & & \sum_{j=1}^{p} \mid v_{ij} \mid \leq t
\end{aligned}
$$

I. Joliffe, T. Trendafilov and M. Uddin, "A modified principal component technique based on the Lasso", JCGS, 2003

Existing methods

- Transforming the original PCA problem into

$$\left(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = \arg\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \sum_{i=1}^{n} \| \mathbf{x}_i - \boldsymbol{\alpha}\boldsymbol{\beta}^\top \mathbf{x}_i \|^2 + \lambda \| \boldsymbol{\beta} \|^2$$
$$\text{subject to} \quad \| \boldsymbol{\alpha} \|^2 = 1$$

- $\hat{\boldsymbol{\beta}} \propto \mathbf{v}_1$
- $\mathbf{A}$ and $\mathbf{B}$ are $p \times d$

$$\left(\hat{\mathbf{A}}, \hat{\mathbf{B}}\right) = \arg\min_{\mathbf{A},\mathbf{B}} \sum_{i=1}^{n} \| \mathbf{x}_i - \mathbf{A}\mathbf{B}^\top \mathbf{x}_i \|^2 + \lambda \sum_{j=1}^{d} \| \boldsymbol{\beta}_j \|^2 + \sum_{j=1}^{d} \lambda_j \mid \boldsymbol{\beta}_j \mid_1$$
$$\text{subject to} \quad \mathbf{A}^\top \mathbf{A} = \mathbf{I}_d$$

- where $\mid \boldsymbol{\beta} \mid_1 = \sum_{i=1}^{p} \mid \beta_i \mid$

Zou, H., Hastie, T., Tibshirani, R., 2006. “Sparse principal component analysis”, JCGS.

Existing methods

- Panalized matrix decomposition

$$\min_{\mathbf{u},\mathbf{b}} \left\|\mathbf{X} - \mathbf{u}\mathbf{b}^\top\right\|_F^2 + \sum_{i=1}^{p} \alpha_i \mid b_i \mid$$
$$\text{subject to } \parallel \mathbf{u} \parallel_2 = 1.$$

- $\mathbf{b} = \lambda \mathbf{v}$

Witten, D.M., Tibshirani, R., Hastie, T., 2009. “A penalized matrix decomposition, with applications to sparse PCA and CCA”. Biostatistics.

Shen, H., Huang, J., 2008. “Sparse principal component analysis via regularized low rank matrix approximation”, JMA.

Existing methods

- Equivalently, PMD is:

$$\begin{aligned} \hat{\mathbf{v}} = \quad & \arg\max_{\mathbf{v}} \quad && \mathbf{v}^\top \mathbf{X}^\top \mathbf{X} \mathbf{v} \\ & \text{subject to} && \| \mathbf{v} \|^2 \leq 1, \\ & && | \mathbf{v} |_1 \leq c \end{aligned}$$

- $\| \mathbf{v} \|_2 \leq 1$ is a convex relaxation of $\| \mathbf{v} \|^2 = 1$.

Witten, D.M., Tibshirani, R., Hastie, T., 2009. "A penalized matrix decomposition, with applications to sparse PCA and CCA". Biostatistics.

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:

$$\mathbf{x} = \mathbf{\Gamma W z} + \boldsymbol{\epsilon}. \tag{1}$$

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

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

$$\mathbf{\Gamma} = diag(1, 1, 1, 1, 0, 0, \ldots, 0)$$

Table 1: 5 samples of generated data $\mathbf{x}$ for $p = 10$

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

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

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

Introductory Example

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

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

loading vectors	$\mathbf{v}_1$
SPCA	-0.340 0.236 -0.910 -0.004 0 0 0 0 0 0

Introductory Example

- As we extract more PCs ($\mathbf{v}_2, \ldots, \mathbf{v}_8$), **the sparsity pattern varies**.

Table 4: Decomposition using SPCA

loading vectors	$\mathbf{v}_1$	$\mathbf{v}_2$	$\mathbf{v}_3$	$\mathbf{v}_4$	$\mathbf{v}_5$	$\mathbf{v}_6$	$\mathbf{v}_7$	$\mathbf{v}_8$
SPCA	-0.340	0.032	-0.204	0.901	0.072	0	0.016	-0.030
	0.236	-0.302	0.892	0.358	-0.040	0	0.043	0.075
	-0.910	-0.100	0.284	-0.239	0.010	-0.010	0	-0.017
	-0.004	0.945	0.288	0.048	-0.054	-0.066	0.027	0
	0	-0.003	0	0	-0.028	0	0.767	0
	0	-0.061	0	0	0	-0.900	0.117	-0.337
	0	0	0	0	0.032	0.231	0.080	0.293
	0	0	0	0	0	-0.063	-0.622	0
	0	0.028	0	0	0.805	-0.215	0.024	0.485
	0	-0.017	0	0	-0.583	-0.285	0	0.748

Preserving Sparsity in PCA

- Some studies explicitly estimate $\mathbf{\Gamma}$.
- Results in an intricate Expectation Maximization approach.
- Requires estimating a large number of parameters.
- computationally expensive.

Mattei, P.A., Bouveyron, C. and Latouche, P., 2016. "Globally sparse probabilistic PCA". AISTATS.

Jenatton, R., Obozinski, G. and Bach, F., 2010. "Structured sparse PCA". AISTATS.

Proposed Method

- while enforcing sparsity on the entries of $\mathbf{b}$,
- adaptively penalize loadings to preserve sparsity pattern.

We propose:

$$\begin{aligned} \arg\min_{\mathbf{U},\mathbf{B}} \quad & \left\| \mathbf{X} - \mathbf{U}\mathbf{B}^\top \right\|_F^2 + \sum_{i=1}^{p} \alpha_i \parallel \mathbf{b}_i \parallel_2 \\ \text{subject to} \quad & \mathbf{U}^\top \mathbf{U} = \mathbf{I}_q \end{aligned}$$

where

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

Proposed Method

Numerical Solution:

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

$$\mathbf{b}_i = \left(1 - \frac{\alpha_i\sqrt{q}}{2 \parallel \mathbf{U}^\top \mathbf{x}_i \parallel_2}\right)_+ \mathbf{U}^\top \mathbf{x}_i$$

  where the operator $(.)_+$ is set to 0 when $\frac{2}{\alpha_i\sqrt{q}} \parallel \mathbf{U}^\top \mathbf{x}_i \parallel_2 < 1$.
- 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:

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

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

For $q = 1$ PCs:

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

$$\text{s.t.} \quad \| \mathbf{u} \|_2 = 1.$$

- SPCA/PMD are a special case of the proposed approach.
- Generalization is due to using different α_i per entry of $\mathbf{b}$.

Table 5: Description of the proposed adaptive sparse PCA algorithm

Algorithm
Given: $\mathbf{X}$, q, α, ϵ.
Tuning parameters selection
Take $\mathbf{B}$ as the first q right singular vectors times the first q singular values of $\mathbf{X}$.
Compute the vector of tuning parameters
$\alpha_i = \frac{\alpha}{\\|\mathbf{b}_i\\|_2}$ for $i = 1, \ldots, p$
While $\\| \mathbf{U}_j - \mathbf{U}_{j-1} \\|_F > \epsilon$
Update: $\mathbf{B}^\top$ per column
For j=1 to p
$\mathbf{b}_i = \left(1 - \frac{\alpha_i \sqrt{q}}{2\\|\mathbf{U}_{j-1}^\top \mathbf{x}_i\\|_2}\right)_+ \mathbf{U}_{j-1}^\top \mathbf{x}_i$
Update: $\mathbf{U}_j = \tilde{\mathbf{U}}\tilde{\mathbf{V}}^\top$ using the SVD of $\mathbf{XB} = \tilde{\mathbf{U}}\mathbf{\Lambda}\tilde{\mathbf{V}}^\top$
Output: $\mathbf{U}$, $\mathbf{B}$

Back to the example:

Table 6: Decomposition using proposed method

loading vectors	$\mathbf{v}_1$	$\mathbf{v}_2$	$\mathbf{v}_3$	$\mathbf{v}_4$	$\mathbf{v}_5$	$\mathbf{v}_6$	$\mathbf{v}_7$	$\mathbf{v}_8$
Proposed	-0.805	0.059	-0.660	1.121	0.029	-0.014	0.008	-0.007
	0.592	-0.627	1.524	0.620	-0.024	0.001	0.013	0.020
	-2.404	-0.252	0.717	-0.310	0.000	-0.006	-0.001	-0.006
	-0.053	2.028	0.598	0.130	-0.039	-0.008	0.011	-0.003
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0

Introduction

CCA finds the **maximum correlation** between **two random vectors**:

$$
\begin{aligned}
(\mathbf{u}_1, \mathbf{v}_1) = \quad & \arg\max_{\mathbf{u},\mathbf{v}} \quad \mathbf{u}^\top \mathbf{\Sigma}_{\mathbf{xy}} \mathbf{v} \\
& \text{subject to} \quad \mathbf{u}^\top \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{u} = 1, \\
& \qquad\qquad\quad\;\; \mathbf{v}^\top \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{v} = 1
\end{aligned}
$$

where

$\mathbf{\Sigma}_{\mathbf{xy}} = E[\mathbf{x}\mathbf{y}^\top]$

$\mathbf{\Sigma}_{\mathbf{xx}} = E[\mathbf{x}\mathbf{x}^\top]$ and $\mathbf{\Sigma}_{\mathbf{yy}} = E[\mathbf{y}\mathbf{y}^\top]$

$\mathbf{x} \in \mathbb{R}^p$ and $\mathbf{y} \in \mathbb{R}^q$

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 $j = 1, \ldots, i-1$.

$$
\begin{aligned}
(\hat{\mathbf{U}}, \hat{\mathbf{V}}) = \quad & \arg\max_{\mathbf{U},\mathbf{V}} \quad tr(\mathbf{U}^\top \boldsymbol{\Sigma}_{\mathbf{xy}} \mathbf{V}) \\
& \text{subject to} \quad \mathbf{U}^\top \boldsymbol{\Sigma}_{\mathbf{xx}} \mathbf{U} = \mathbf{I}_r, \\
& \qquad\qquad\quad\; \mathbf{V}^\top \boldsymbol{\Sigma}_{\mathbf{yy}} \mathbf{V} = \mathbf{I}_r
\end{aligned}
$$

where

$tr(.)$ 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:

$$\frac{1}{n}\mathbf{X}^\top\mathbf{Y} \to \mathbf{\Sigma_{xy}}$$
$$\frac{1}{n}\mathbf{X}^\top\mathbf{X} \to \mathbf{\Sigma_{xx}}$$
$$\frac{1}{n}\mathbf{Y}^\top\mathbf{Y} \to \mathbf{\Sigma_{yy}}$$

where

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

$$(\hat{\mathbf{U}}, \hat{\mathbf{V}}) = \arg\min_{\mathbf{U},\mathbf{V}} \| \mathbf{XU} - \mathbf{YV} \|_F^2 \tag{2}$$

instead of $\max_{\mathbf{U},\mathbf{V}} \; tr(\mathbf{U}^\top\mathbf{X}\mathbf{Y}^\top\mathbf{V})$.

Existing Solution

- Penalized Matrix Decomposition for Sparse CCA:

$$
\begin{aligned}
(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = \quad & \arg\max_{\mathbf{u},\mathbf{v}} \quad && \mathbf{u}^\top \mathbf{X}^\top \mathbf{Y} \mathbf{v} \\
& \text{subject to} && \mathbf{u}^\top \mathbf{X}^\top \mathbf{X} \mathbf{u} \le 1, \mathbf{v}^\top \mathbf{Y}^\top \mathbf{Y} \mathbf{v} \le 1, \\
& && |\ \mathbf{u}\ |_1 \le c_1, |\ \mathbf{v}\ |_1 \le c_2,
\end{aligned}
$$

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

Witten, D.M., Tibshirani, R. and Hastie, T., 2009. A penalized matrix decomposition, with applications to sparse PCA and CCA. Biostatistics.

Introductory Example: data generation model

- Consider the sparse data generating model:

$$\mathbf{x} = \boldsymbol{\Gamma}_{\mathbf{x}}\mathbf{W}_{\mathbf{x}}\mathbf{z} + \boldsymbol{\epsilon}_{\mathbf{x}}$$
$$\mathbf{y} = \boldsymbol{\Gamma}_{\mathbf{y}}\mathbf{W}_{\mathbf{y}}\mathbf{z} + \boldsymbol{\epsilon}_{\mathbf{y}}$$

where

- $\boldsymbol{\Gamma}_{\mathbf{x}}$ and $\boldsymbol{\Gamma}_{\mathbf{y}}$ are diagonal matrices that enforce sparsity.
- $\mathbf{W}_{\mathbf{x}}$ and $\mathbf{W}_{\mathbf{y}}$ are orthogonal matrices.
- $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_r)$.
- $\boldsymbol{\epsilon}_{\mathbf{x}}$ and $\boldsymbol{\epsilon}_{\mathbf{y}}$ 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 $\mathbf{z}$) is $r = 4$.
- noise variance for both signals is $\sigma^2 = 0.1$.
- Sparsity patterns for $\mathbf{x}$ and $\mathbf{y}$ are:

$$\mathbf{\Gamma_x} = diag(1, 1, 1, 1, 0, \ldots, 0)$$
$$\mathbf{\Gamma_y} = diag(0, \ldots, 0, 1, 1, 1, 1, 1)$$

Introductory Example

Using PMD:

Table 7: loading vectors of **X** from PMD

$\mathbf{U}$	0.83 -0.08 0.4 0.38 0.01 0 0 0 0 0

Introductory Example

Remaining loading vector matters have varying sparsity pattern.

Table 8: loading vectors of $\mathbf{X}$ from PMD

$\mathbf{U}$	0.83	0.02	-0.56	0.04
	-0.08	0.33	0.08	0.97
	0.4	-0.66	0.6	0.15
	0.38	0.68	0.56	-0.2
	0.01	0	-0.02	0.01
	0	-0.01	-0.04	0.01
	0	0.02	0	0
	0	-0.01	0	0.02
	0	0	0	0
	0	0	0	0

Introdctory Example

Similarly for **Y**

Table 9: loading vectors of **Y** from PMD

V	0 0 0 0 0 0 -0.1 -0.14 0.68 -0.15 0.69

Introductory Example

Remaining loading vector matters have varying sparsity pattern.

Table 10: loading vectors of **Y** from PMD

V	0	-0.02	0	0
	0	0	0	0
	0	0	0	0
	0	0.01	0	0
	0	0	-0.01	-0.03
	0	0.01	0	0
	-0.1	0.41	0.37	0.81
	-0.14	-0.62	0.76	0
	0.68	0.01	0	0.32
	-0.15	0.67	0.42	-0.38
	0.69	0.05	0.32	-0.33

Proposed Method

Our proposed method uses adaptive weights for the sparsity penalty:

$$\min_{\mathbf{U},\mathbf{V}} \parallel \mathbf{XU} - \mathbf{YV} \parallel_F^2 + \sum_{i=1}^{p} \alpha_i \parallel \mathbf{u}^i \parallel_2 + \sum_{j=1}^{q} \beta_j \parallel \mathbf{v}^j \parallel_2 \qquad (3)$$

where

$\mathbf{u}^i$ represents the i^{th} **row** of $\mathbf{U}$,

$\mathbf{v}^j$ represents the j^{th} **row** of $\mathbf{V}$.

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

$$\mathbf{v}^j = \frac{1}{\mathbf{y}_j^\top \mathbf{y}_j} \left(1 - \frac{1}{\| \mathbf{y}_j^\top \mathbf{E}_j \|_2} \right)_+ \mathbf{y}_j^\top \mathbf{E}_j$$

where $\mathbf{y}_j$ is the j^{th} column of $\mathbf{Y}$, $\mathbf{E}_j = \mathbf{X}\mathbf{U} - \sum_{i=1, i \neq q}^{q} \mathbf{y}_i \mathbf{v}^i$ and $()_+ = \max(0, x)$

Proposed Method

Table 11: loading vectors of $\mathbf{X}$ - Proposed method

U	0.028	-0.009	0.019	0.009
	0.014	-0.016	0.009	0.002
	-0.026	0.029	-0.015	0.009
	-0.026	0.011	-0.017	0.006
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0

Proposed Method

Table 12: loading vectors of $\mathbf{Y}$ - Proposed method

$\mathbf{V}$	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0.018	-0.007	0.012	0.008
	0.028	-0.006	0.019	0.006
	-0.024	0.022	-0.015	0.003
	-0.016	0.017	-0.01	-0.002
	-0.025	0.009	-0.017	-0.008

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.

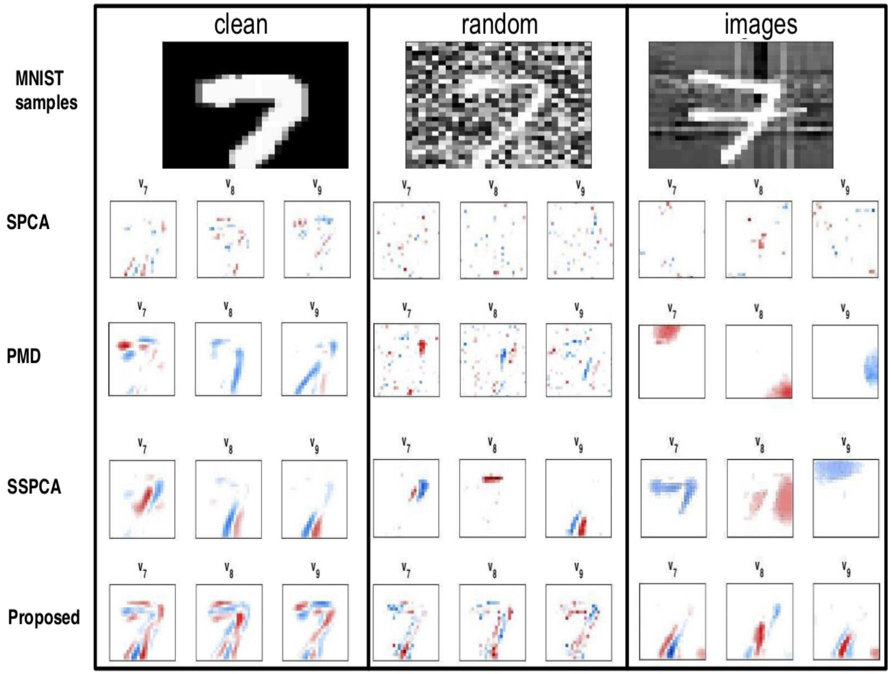
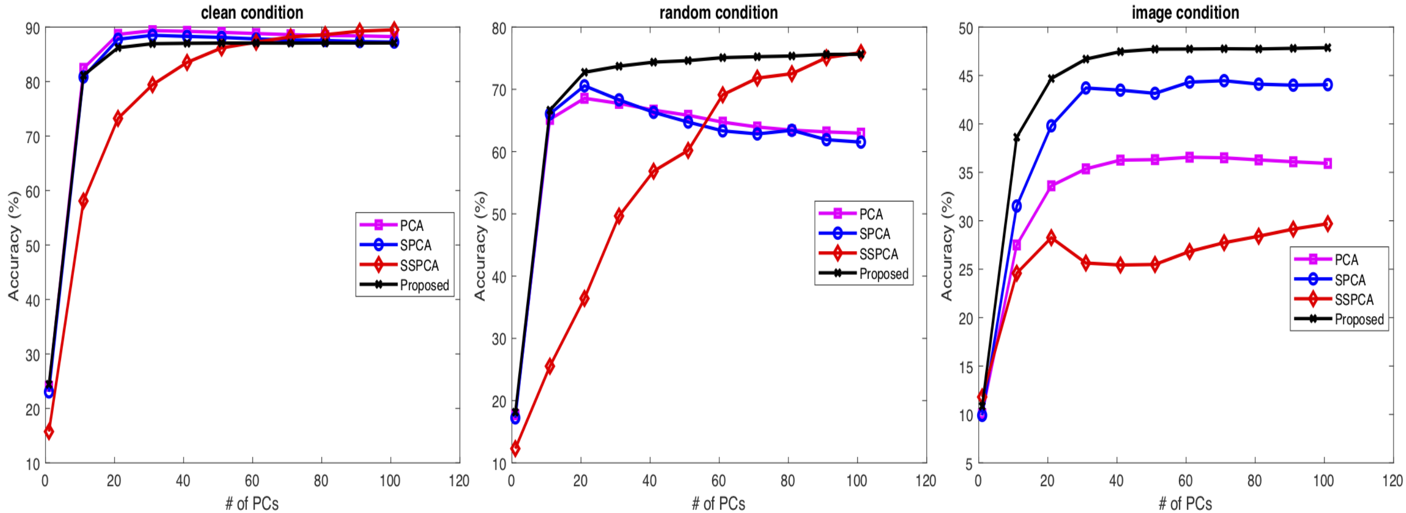


Figure 2: SPCA, PMD, SSPCA, and Proposed. backgrounds: clean (left), uniform (middle), and non-uniform (right) noise.

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).

$$\mathbf{X}_{voxels \times timepoints} = \mathbf{Spatial} \times \mathbf{Temporal}$$

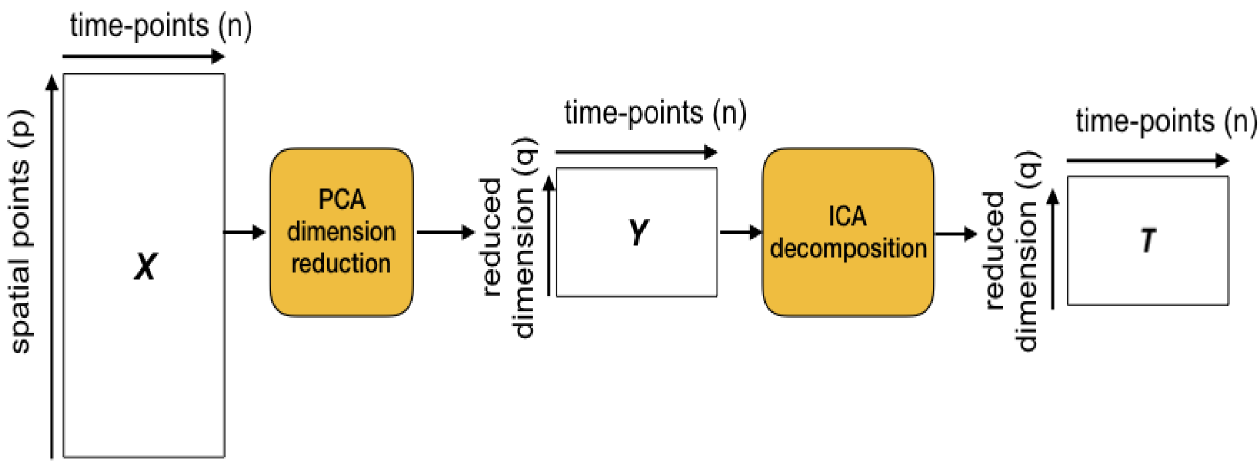
- One of the issues of using ICA for fMRI data is high-dimensionality, which
  - Negatively impacts reproducability of ICA results.
  - Is computationally expensive.
- fMRI researchers use PCA as a dimension reduction stage.
- However, the issue with PCA is that it distorts the spatial arrangement of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

Proposed solution:

- Using sparse PCA can focus only on local brain regions that have valuable information.

fMRI source Separation

Source separation pipeline for fMRI data:



- Spatial maps are reconstructed by calculating the correlation of the resulting q time series with the original data.

fMRI source Separation: Simulated data

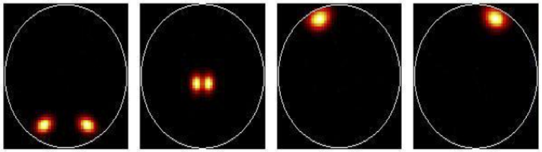
Extracting spatial maps for simulated fMRI-like data:

$$\mathbf{X}_{p\times n} = \mathbf{S}_{p\times q}\mathbf{T}_{q\times n} + \boldsymbol{\xi} \tag{4}$$

where

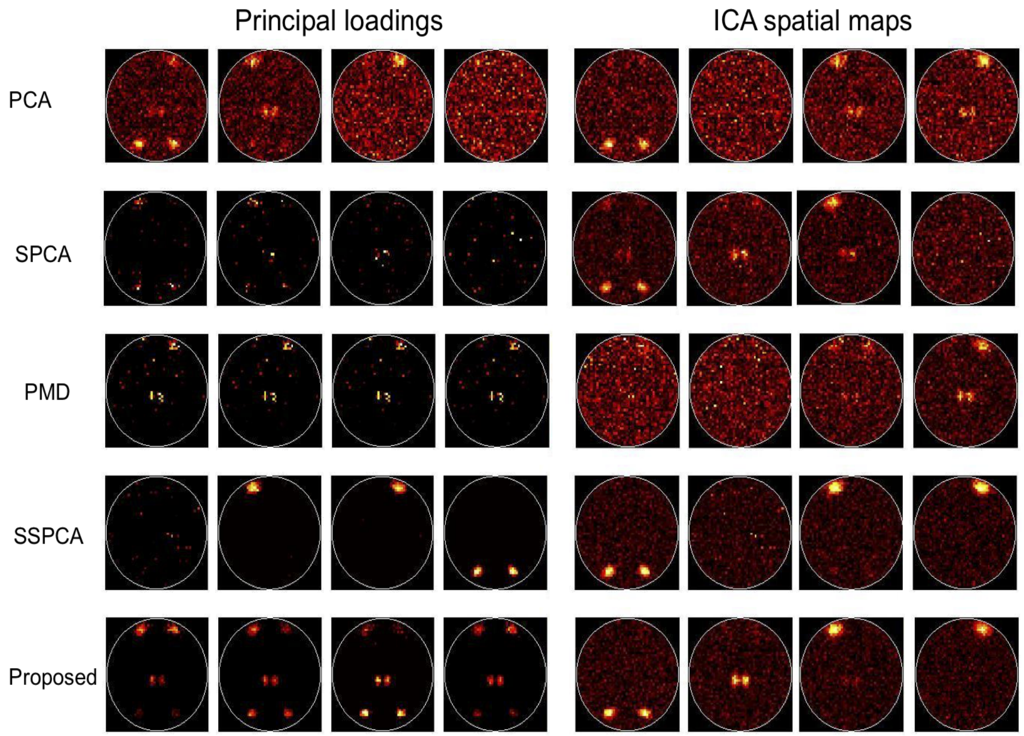
- $\mathbf{S}$: q spatial maps of dimension $p \times 1$.
- $\mathbf{T}$: q time series of dimension $n \times 1$ calculated from a independent Gaussian processes.
- $\boldsymbol{\xi}$: White Gaussian noise with variance σ^2.

original spatial maps



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:

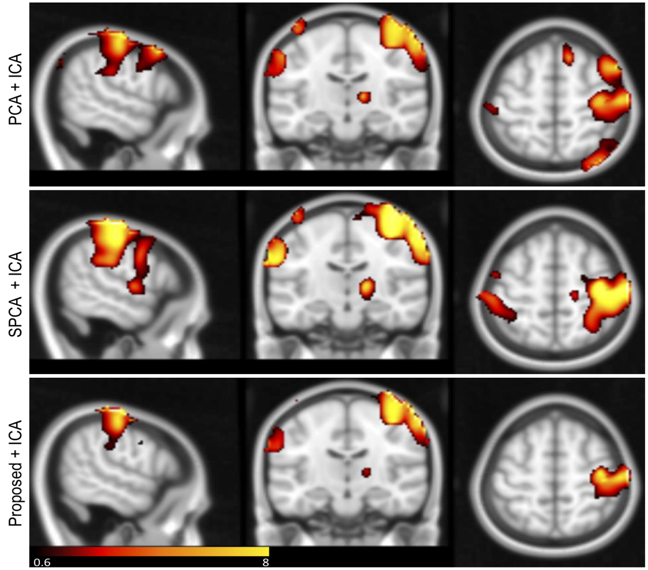


Figure 3: Spatial maps corresponding to motor cortex.

- Less spurious results from using proposed sparse PCA.

fMRI source Separation: Real data

- Corresponding time series.

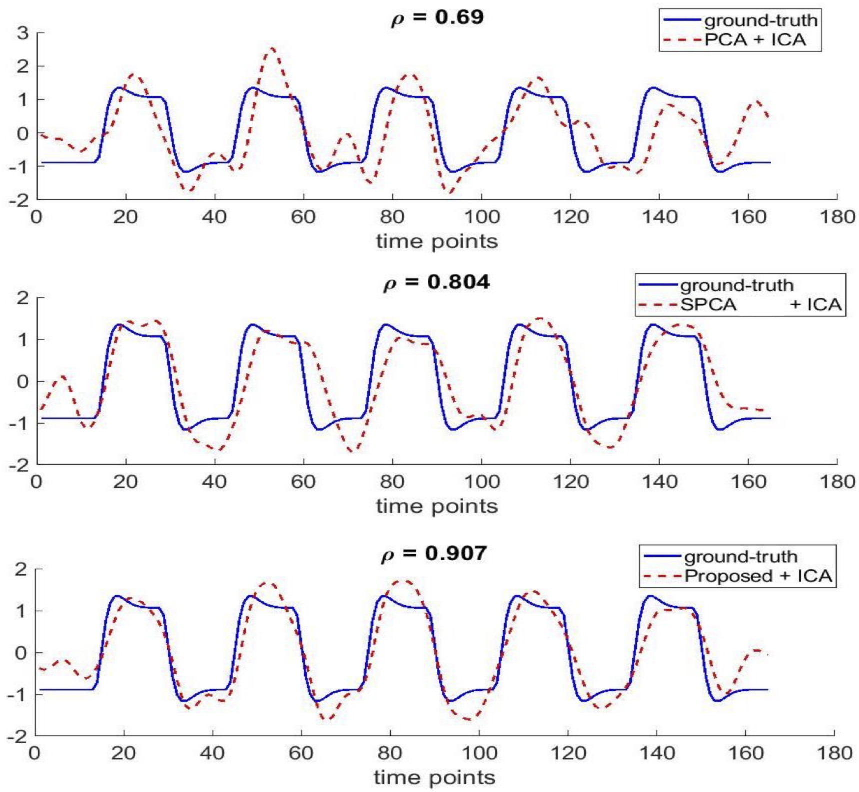


Figure 4: Time-courses for finger-tapping experiment. Blue: ground-truth, Red: estimated time-course using ICA.

References

1. Seghouane, Shokouhi, Koch, “Sparse Principal Component Analysis with Preserved Sparsity Pattern”, IEEE Trans. on Image Processing, 2019.
2. Seghouane, Shokouhi, “Sparsity Preserved Canonical Correlation Analysis”, IEEE Trans. on Pattern Analysis and Machine Intelligence, 2019.

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