AC-PCA adjusts for confounding variation and recovers the anatomical structure of neocortex

Zhixiang Lin, Can Yang, Ying Zhu, John C. Duchi, Yao Fu, Yong Wang, Bai Jiang, Mahdi Zamanighomi, Xuming Xu, Mingfeng Li, Nenad Sestan, Hongyu Zhao* & Wing Hung Wong*

March 8, 2016

Abstract: Dimension reduction methods are commonly applied to high-throughput biological datasets. However, the results can be hindered by confounding factors, either biologically or technically originated. In this study, we propose a Principal Component Analysis-based approach to Adjust for Confounding variation (AC-PCA). We show that AC-PCA can adjust for variations across individual donors present in a human brain exon array dataset. Our approach is able to recover the anatomical structure of neocortex regions, including the frontal-temporal and dorsal-ventral axes, and reveal temporal dynamics of the interregional variation. For gene selection purposes, we extend AC-PCA with sparsity constraints, and propose and implement an efficient algorithm. The top selected genes from this algorithm demonstrate frontal/temporal and dorsal/ventral expression gradients and strong functional conservation.

1 Introduction

Dimension reduction methods, such as Multidimensional Scaling (MDS) and Principal Component Analysis (PCA) are commonly applied in high-throughput biological datasets to visualize data in a low dimensional space, identify dominant patterns and extract relevant features [48, 44, 20, 31, 40, 14]. MDS aims to place each sample in a lower-dimensional space such that the between-sample distances are preserved as much as possible [33]. PCA seeks the linear combinations of the original variables such that the derived variables capture maximal variance [28]. One advantage of PCA is that the principal components (PCs) are more interpretable by checking the loadings of the variables.

Confounding factors, technically or biologically originated, are commonly observed in high throughput biological experiments. Various methods have been proposed to remove the unwanted variation through regression models on known confounding factors [26], factor models and surrogate vector analysis for unobserved confounding factors [36, 19, 35, 45, 57]. However, directly removing the confounding variation using these methods may introduce bias, as a result of
incorrect model assumption of the confounding variation, and it can also remove the desired biological variation. Moreover, limited work has been done in the context of dimension reduction.

To address the limitations of existing methods, we have developed AC-PCA for simultaneous dimension reduction and adjustment for confounding variation. We demonstrate the performance of AC-PCA though the human brain exon array dataset [31]. In the dataset, we found that visualization of the neocortex regions is affected by confounding factors, likely originated from the variations across individual donors (Figure 1). As a result, a) there is no clear pattern among the neocortex regions or/and samples from the same individual donors tend to form clusters, and b) it is challenging to identify neurodevelopmental genes with interregional variation. Applying AC-PCA to the human brain exon array dataset [31], we are able to recover the anatomical structure of neocortex regions, including the frontal to temporal axis and the dorsal to ventral axis. Our results are able to capture the interregional variation in neocortex and reveal the temporal dynamics of the spatial variation. In PCA, the loadings for the variables are typically nonzero. In high dimensional settings, sparsity constraints have been proposed in PCA for better interpretation of the PCs [29, 59, 55, 43] and better statistical properties, such as asymptotic consistency [27, 30, 38]. We have also developed an efficient and fast algorithm to find sparse solutions for AC-PCA. The genes identified by AC-PCA demonstrate smooth frontal to temporal, dorsal to ventral gradient and strong functional conservation.

2 Results

The human brain exon array dataset reported in [31] includes the transcriptomes of 16 brain regions comprising 11 areas of the neocortex, the cerebellar cortex, mediodorsal nucleus of the thalamus, striatum, amygdala and hippocampus. Because the time period system defined in [31] had varying numbers of donors across developmental epochs, we grouped samples from every 6 donors, sorted by age and beginning from period 3. While the last time window had only 5 donors, this reorganization more evenly distributed sample sizes and allowed improved comparisons across time (Table 1).

In the analysis, we used samples from 10 regions in the neocortex. V1C was excluded from the analysis as the distinct nature of this area relative to other neocortical regions tended to compress the other 10 regions into a single cluster. We conducted standard Principal Component Analysis (PCA) for windows 1 and 2, as shown in Figures 1a and 1b. At first glance, neither analysis produced any clear patterns among neocortical regions. However, closer observation of these plots suggested some underlying structure: we performed PCA on just the right hemisphere of donor HSB113 and found that the gross morphological structure of the hemisphere was largely recapitulated (Figure 1c). The gross structure tends to be consistent between hemispheres and across donors within time windows when PCA is performed simultaneously, but the pattern is largely
Table 1: Span of the windows in this study.

<table>
<thead>
<tr>
<th>Window</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12PCW ≤ Age &lt; 16PCW</td>
</tr>
<tr>
<td>2</td>
<td>16PCW ≤ Age &lt; 21PCW</td>
</tr>
<tr>
<td>3</td>
<td>21PCW ≤ Age &lt; 35PCW</td>
</tr>
<tr>
<td>4</td>
<td>35PCW ≤ Age &lt; 10M</td>
</tr>
<tr>
<td>5</td>
<td>10M ≤ Age &lt; 8Y</td>
</tr>
<tr>
<td>6</td>
<td>8Y ≤ Age &lt; 19Y</td>
</tr>
<tr>
<td>7</td>
<td>19Y ≤ Age &lt; 30Y</td>
</tr>
<tr>
<td>8</td>
<td>30Y ≤ Age &lt; 42Y</td>
</tr>
<tr>
<td>9</td>
<td>42Y ≤ Age</td>
</tr>
</tbody>
</table>

M: postnatal months
PCW: post-conception weeks
Y: postnatal years.

distorted, likely due to the small sample size and noisy background (Supplementary Materials). If we first regress the gene expression levels on the individual labels and then perform PCA on the residuals, the result is better but still not satisfactory (Supplementary Materials).

In contrast, when we applied AC-PCA to see the effectiveness of our approach in adjusting confounding effects from individual donors, we were able to recover the anatomical structure of neocortex (Figure 1f; Supplementary Figures 1a, 1b, 2a and 2b in [31]). Every region tended to form a smaller cluster, when \( \lambda \) is larger than the best tuning value. Variation of the principal components shrinks and the overall pattern remains consistent (Supplementary Materials). We also compare AC-PCA with ComBat [26] and SVA [36], where regular PCA is implemented after removing the confounder effects (Supplementary Materials). In windows 1 to 3, AC-PCA performs slightly better. In windows 4 to 9, AC-PCA outperforms ComBat and SVA.

Next, we explored the temporal dynamics of the principal components (PC) (Figure 2). The pattern is similar from windows 1 to 5, with PC1 representing the frontal to temporal gradient, which follows the contour of developing cortex [40], and PC2 representing the dorsal to ventral gradient. Starting from window 6, these two components reversed order. In windows 6 to 9, MFC shows increasing distance from the other regions, and primary areas (M1C, S1C, A1C) are separated from other regions in PC1, implying the prominence of transcriptional changes during specification of regional functions.

We also calculated the interregional variation explained by the PCs (Figure 3). In the first three windows, PC1 explains about 20% of the interregional variation. The variation explained by the first two PCs decreases close to birth (window 4) and then increases in later time windows, similar to the “hourglass” pattern previously reported based on cross-species comparison and differential expression [41, 37]. Interestingly, if we compare the proportion of variation explained by the first two PCs, a smooth pattern is revealed: compared with
Figure 1: Visualization of the brain exon-array data [31], windows 1 and 2. Each color represents a donor. Samples from the right hemisphere are labeled as italic. For both windows, the best tuning parameter $\lambda^*$ was 3. In (f), MFC is not visible on the lateral surface, and it belongs to slice L2. The left and right hemispheres from the same donor were treated as different individuals when implementing AC-PCA.
Figure 2: Temporal dynamics of the principal components. Median was taken across individuals. \( \lambda \) was fixed to 2 for all windows when implementing Formulation (1).
Figure 3: Interregional variation explained by the principal components. Interregional variation is calculated to be the sum over the variance across regions among the individuals.

PC1, the variation explained by PC2 first increases, in window 5, it nearly equals to that of PC1, then it decreases and finally slightly increases.

We then implemented AC-PCA with sparsity constrains to select genes associated with the PCs. The number of genes with non-zero loadings are shown in Figure 4, along with the interregional variation explained in the regular PCs. Interestingly, the trends tend to be consistent: when the regular PC explains more variation, more genes are selected in the corresponding sparse PC; compared with PC1, fewer genes tend to be selected in PC2. The numbers of selected genes are quite dynamic. At both early developmental periods prior to birth and later postnatal periods, more genes tend to be selected in both PCs. For PC1 in windows 1, 2, 3 and 9, more than 4,000 genes are selected, which indicates global trends in neocortex that affect large numbers of genes.
Figure 4: Number of genes selected in the sparse PCs and the interregional variation explained by the regular PCs.
Figure 5: Overlap of the top 200 genes in PC1 and PC2.

To produce more stringent and comparable gene lists, we chose the sparsity parameter such that 200 genes are selected in each window. The overlap of gene lists across windows is moderate (Figure 5) and, as expected, the overlap with the first window decreases over time. The overlap between adjacent windows tend to be larger in later time windows, indicating that interregional differences become stable. Interestingly, the overlap between windows 2 and 3 is also large. Results of pathway enrichment analysis using DAVID bioinformatics resources [25, 49] are available in the Supplementary Tables.

In windows 1 and 3, genes with the largest loadings demonstrate interesting spatial patterns (Figure 6). For PC1, the top genes follow the frontal to temporal gradient; while for PC2, they tend to follow the dorsal to ventral gradient. A brief overview of the functions of these genes are listed in Table 2.

Finally, we demonstrate the functional conservation of the 200 genes selected in PC1 and PC2 (Figure 7). These genes tend to have low dN/dS scores for human vs. macaque comparison, even lower than the complete list of all essential genes. In the human vs. mouse comparison, we observed a similar trend (Supplementary Materials). Parallel to the cross-species conservation, we also observed that these genes tend to have low heterozygosity scores, a measure of functional conservation in human (Figure 7).

3 Simulation

We first tested the performance of AC-PCA on simulated datasets with individual variation. We considered $n = 5$, $b = 10$ and $p = 400$.

**Simulation setting 1**: for the $i$th individual, the $b \times p$ matrix $X_i = Wh + \alpha Bs_i + \epsilon_i$. $Wh$ represents the shared variation across individuals. $\alpha Bs_i$ corresponds to individual variation and $\epsilon_i$ is noise. $W = (w_1 \ w_2)$ is a $b \times 2$ matrix, representing the
Figure 6: Expression levels for the genes with top loadings in windows 1 and 3. The top 3 genes are shown and each point represents the median over the individuals.
Table 2: Top genes in the PCs and their functions.

<table>
<thead>
<tr>
<th>Window 1, PC1</th>
<th>Function and Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>NECAB1</td>
<td>calcium ion binding [50, 18]</td>
</tr>
<tr>
<td>SEMA3E</td>
<td>semaphorin, control vascular morphogenesis, serve as axon guidance ligands [22, 47, 10]</td>
</tr>
<tr>
<td>PCDH17</td>
<td>calcium ion binding, establishment and function of specific cell-cell connections in the brain [56, 23]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Window 1, PC2</th>
<th>Function and Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>NTS</td>
<td>dopamine signaling [32]</td>
</tr>
<tr>
<td>SLIT2</td>
<td>axonal guidance, midline guidance in the forebrain [6, 5, 42]</td>
</tr>
<tr>
<td>ADAMTS3</td>
<td>extracellular matrix proteases, cleaves the propeptides of type II collagen [52, 17]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Window 3, PC1</th>
<th>Function and Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>POPDC3</td>
<td>important in heart development [8, 4]</td>
</tr>
<tr>
<td>GREM1</td>
<td>BMP Antagonist, may play a role in regulating organogenesis, body patterning, and tissue differentiation [24]</td>
</tr>
<tr>
<td>CPNE8</td>
<td>calcium-dependent phospholipid binding [53, 12]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Window 3, PC2</th>
<th>Function and Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRP</td>
<td>gastrin-releasing peptide, regulates the gastrointestinal and central nervous systems [39, 9, 51]</td>
</tr>
<tr>
<td>MCHR2</td>
<td>receptor for Melanin-concentrating hormone (MCH), important in feeding behaviors and energy metabolism [3, 54]</td>
</tr>
<tr>
<td>IGFBP5</td>
<td>insulin-like growth factor (IGF) binding protein, essential for growth and development [1, 46]</td>
</tr>
</tbody>
</table>
Figure 7: Conservation and heterozygosity scores for the genes in PC1 and PC2. “All” represents the total 17,568 genes in the exon array experiment. “All-5000” represents the 5,000 genes used in the analysis. “Essential” represents a list of 337 genes that are functionally conserved and essential, obtained from the Database of Essential Genes (DEG) version 5.0 [58]. “Window” is abbreviated as “W”.
latent structure of the shared variation. For visualization purpose, we assumed that it is smooth and has rank 2. Let \( \mu = (1, \cdots, b)' \) and \( w_1 \) is the normalized \( \mu \), with mean 0 and variance 1. \( w_2 \sim \mathcal{N}(0, 0.25 \cdot \Sigma) \), where \( \Sigma_{ij} = \exp(-\frac{(w_{1i} - w_{1j})^2}{4}) \).

\( h \) is a \( 2 \times p \) matrix and the rows in \( h \) are generated from \( \mathcal{N}(0, I_p) \), where \( I_p \) is the \( p \times p \) identity matrix. \( B \) is a \( b \times 1 \) matrix with all 1s. \( s_i \) is generated from \( \mathcal{N}(0, I_p) \). \( \alpha \) is a scalar indicating the strength of confounding variation, we set \( \alpha = 2.5 \). The rows in \( \epsilon_i \) are generated from \( \mathcal{N}(0, 0.25 \cdot I_p) \).

**Simulation setting 2:** for the \( i \)th individual, let \( X_i = \begin{pmatrix} X_{1i} \\ X_{2i} \end{pmatrix} \), where \( X_{1i} \) represents the data matrix for the first 200 variables and \( X_{2i} \) represents that of the other 200 variables. \( X_{1i} = Wh + \epsilon_{1i} \) and \( X_{2i} = \alpha W_i h_i + \epsilon_{2i} \). \( W, h, h_i, \epsilon_{1i} \) and \( \epsilon_{2i} \) are generated similarly as that in setting 1. \( \alpha = 2.5 \). The first column in \( W_i \) is generated from \( \mathcal{N}(0, I_b) \), and the second column is generated from \( \mathcal{N}(0, 0.25 \cdot I_b) \), where \( I_b \) is the \( b \times b \) identity matrix.

Settings 1 and 2 represent two different scenarios: in setting 1, the individual variation is represented by a global trend for all the variables; while in setting 2, we assumed that for some variables, the variation is shared among individuals, and it is not shared for the other variables. The results of data visualization are presented in Figure ??.

Then we tested AC-PCA on simulated datasets with other confounding structure:

**Simulation setting 3:** Let \( N = 10 \) and \( p = 400 \). The \( N \times p \) matrix \( X = Wh + \alpha B s + \epsilon \). \( W, h \) are the same as that in setting 1. We set \( \alpha = 2.5 \). \( B = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \) is a \( N \times 2 \) matrix: the entries in \( b_1 \) have 0.3 probability of being 0, otherwise the entries are set to 1; \( b_2 \) equals \( 1 - b_1 \). \( s \) is a \( 2 \times p \) matrix, where the rows are generated from \( \mathcal{N}(0, I_p) \). \( \epsilon \) is an \( N \times p \) matrix, where the rows are generated from \( \mathcal{N}(0, 0.25 \cdot I_p) \).

**Simulation setting 4:** Let \( N = 10 \) and \( p = 400 \). The \( N \times p \) matrix \( X = Wh + \alpha \tilde{w}_1 s + \epsilon \). \( W, h \) are the same as that in setting 1. We set \( \alpha = 2.5 \). \( \tilde{w}_1 \) is a permutation of \( w_1 \). \( s \) is generated from \( \mathcal{N}(0, I_p) \). \( \epsilon \) is an \( N \times p \) matrix, where the rows are generated from \( \mathcal{N}(0, 0.25 \cdot I_p) \).

Setting 3 represents an experiment with two “batches”, contributing globally to the data. Setting 4 represents an experiment with a continuous confounding factor (e.g. age). We implemented AC-PCA with linear kernels, and set \( Y = B \) and \( \tilde{w}_1 \) for settings 3 and 4, correspondingly. The visualization results are shown in Figure 8.

Finally, we tested AC-PCA for variable selection when the true loading is sparse. For simplicity, we assumed that the latent factor is of rank 1.

**Simulation setting 5:** for the \( i \)th individual, let \( X_i = \begin{pmatrix} X_{1i} \\ X_{2i} \end{pmatrix} \), where \( X_{1i} \) represents the data matrix for the first 200 variables and \( X_{2i} \) represents that of the other 200 variables. \( X_{1i} = wh + \epsilon_{1i} \) and \( X_{2i} = \alpha W_i h_i + \epsilon_{2i} \). \( w \) is the same as \( w_1 \) in setting 1. Some entries in \( h \) are set to 0 to reflect sparsity, the other entries are generated from \( \mathcal{N}(0, I) \). \( W_i \) and \( h_i \) are the same as that in setting 2. The rows in \( \epsilon_{1i} \) and \( \epsilon_{2i} \) are generated from \( \mathcal{N}(0, \sigma^2 \cdot I) \), where \( \sigma \) is a scalar.
Figure 8: Visualization of simulated data with confounding variation. The parameter $\lambda$ was tuned as described in the previous section.
Table 3: Estimation of sparsity and sensitivity

<table>
<thead>
<tr>
<th></th>
<th>True non-zeros=100</th>
<th>True non-zeros=40</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\alpha = 1.5$</td>
<td>$\alpha = 2$</td>
</tr>
<tr>
<td>Estimated non-zeros $\sigma = 0.2$</td>
<td>113.7(34.2)</td>
<td>88.7(20.9)</td>
</tr>
<tr>
<td>Estimated non-zeros $\sigma = 0.5$</td>
<td>120.6(42.5)</td>
<td>91.6(31.1)</td>
</tr>
<tr>
<td>Sensitivity $\sigma = 0.2$</td>
<td>0.85(0.06)</td>
<td>0.83(0.08)</td>
</tr>
<tr>
<td>Sensitivity $\sigma = 0.5$</td>
<td>0.80(0.07)</td>
<td>0.79(0.08)</td>
</tr>
</tbody>
</table>

To calculate sensitivity, $c_2$ was chosen such that the estimated non-zero entries equals the true number. In $\alpha = 2^*$, $wh$ was scaled with 2.5 to let the variation match with the less sparse setting.

Results of simulation setting 5 are shown in Table 3. Larger noise ($\sigma$) leads to lower sensitivity, larger standard error for the estimated non-zeroes, but does not affect the mean much. Smaller confounding variation ($\alpha$) leads to overestimate of the non-zeroes, but does not affect the sensitivity much.

4 Discussion

Dimension reduction methods are commonly applied to visualize datapoints in a lower dimensional space and identify dominant patterns in the data. Confounding variation may affect the performance of these methods, and hence the visualization and interpretation of the results (Figure 1).

In this study, we have proposed AC-PCA for simultaneous dimension reduction and adjustment for confounding variation, and demonstrated its good performance through the analysis of human brain developmental exon array dataset [31] and simulated data as well. We showed that AC-PCA is able to recover the anatomical structure of the neocortex regions. Because of the structural complexity of primate neocortex, physical distance may not be a good measure for the true similarity between regions. Our results showed that AC-PCA is able to reconstruct the regional map in neocortex based on transcriptome data alone. We also demonstrated that AC-PCA is able to selected genes with frontal to temporal gradient and dorsal to ventral gradient, which may be important for cortical patterning.

One feature of AC-PCA is its simplicity. There is no need to specify any analytical forms for the confounding variation. Instead, we extended the objective function of regular PCA with penalty on the dependence between the principal components and the confounding factors. AC-PCA is designed to capture the desired biological variation, even when the confounding factors are unobserved, as long as the labels for the primary variable of interest are known.

The application of AC-PCA is not limited to transcriptome datasets. Dimension reduction methods have been applied to other types of genomics data for various purposes, such as feature extraction for methylation prediction [15], classifying yeast mutants using metabolic footprinting [2], and classifying im-
mune cells using DNA methylome [34], etc. AC-PCA is applicable to these datasets to capture the desired variation, adjust for potential confounders, and select the relevant features. AC-PCA can serve as an exploratory tool and be combined with other methods. For example, the extracted features can be implemented in regression models for more rigorous statistical inference. The R package with user’s guide and application examples is available on https://github.com/linzx06/AC-PCA. The Matlab source code is available upon request.

5 Methods

5.1 AC-PCA adjusting for variations of individual donors

Let $X_i$ represent the $b \times p$ matrix for the gene expression levels of individual $i$, where $b$ is the number of brain regions and $p$ is the number of genes. By stacking the rows of $X_1, \ldots, X_n$, $X$ represents the $(n \times b) \times p$ matrix for the gene expression levels of $n$ individuals. We propose the following objective function to adjust for individual variation:

$$\maximize_{v \in \mathbb{R}^p} v^T X^T X v - \frac{2}{n-1} \lambda \sum_{i=1}^{n-1} \sum_{j=i+1}^n v^T (X_j - X_i)^T (X_j - X_i) v$$

subject to $\|v\|_2^2 \leq 1$. (1)

In (1), the term $v^T X^T X v$ is the objective function for standard PCA, and the regularization term $-\sum_{i=1}^{n-1} \sum_{j=i+1}^n v^T (X_j - X_i)^T (X_j - X_i) v$ encourages the coordinates of the brain regions across individuals to be similar. The factor $\frac{2}{n-1}$ makes the regularization term in formulation (1) scale linearly with the number of individuals. The tuning parameter $\lambda > 0$ controls the strength of regularization. When $\lambda = +\infty$, we are forcing the coordinates of the same brain region across individuals to be the same after projection. Only the labels for brain regions (i.e. labels for the primary variables) are required when implementing formulation (1). We can apply it even if the individual labels of donors (i.e. the confounding variables) are unknown. The connection of formulation (1) with Canonical Correlation Analysis (CCA) is shown in the appendix.

5.2 AC-PCA in a general form

Let $X$ be the $N \times p$ data matrix and $Y$ be the $N \times l$ matrix for $l$ confounding factors. Denote $y_i$, the $i$th row in $Y$. We propose the following objective function to adjust for more general confounding variation:

$$\maximize_{v \in \mathbb{R}^p} v^T X^T X v - \lambda v^T X^T K X v$$

subject to $\|v\|_2^2 \leq 1$, (2)

where $K$ is the $N \times N$ kernel matrix, and $K_{ij} = k(y_i, y_j)$. It can be shown that $v^T X^T K X v$ is the same as the empirical Hilbert-Schmidt independence
criterion[21, 7] for $Xv$ and $Y$, when linear kernel is applied on $Xv$ (Appendix). In the objective function, we are penalizing the dependence between $Xv$ (i.e. extracted feature) and the confounding factors. Formulation (1) is a special case of (2), where linear kernel (i.e. $YY^T$) is applied on $Y$, and $Y$ has the following structure: in each column of $Y$, there are only two non-zero entries, $\sqrt{2/(n-1)}$ and $-\sqrt{2/(n-1))}$, corresponding to a pair of samples from the same brain region but different individuals.

Denote $Z = X^T X - \lambda X^T K X$. Problem (2) can be rewritten as:

$$\max_{v \in \mathbb{R}^p} v^T Z v \quad \text{subject to } \|v\|_2^2 \leq 1.$$ (3)

Therefore it can be solved directly by implementing eigendecomposition on $Z$.

### 5.3 AC-PCA with sparse loading

Denote $H = X^T K X$. It can be shown that solving (2) is equivalent to solving:

$$\max_{v \in \mathbb{R}^p} v^T X^T X v \quad \text{subject to } v^T H v \leq c_1, \|v\|_1 \leq c_2, \|v\|_2^2 \leq 1,$$ (4)

where $c_1$ is a constant depending on $\lambda$. A sparse solution for $v$ can be achieved by adding $\ell_1$ constraint:

$$\max_{v \in \mathbb{R}^p} v^T X^T X v \quad \text{subject to } v^T H v \leq c_1, \|v\|_1 \leq c_2, \|v\|_2^2 \leq 1.$$ (5)

Following [55], this is equivalent to:

$$\min_{u, v \in \mathbb{R}^p} -u^T X v \quad \text{subject to } v^T H v \leq c_1, \|v\|_1 \leq c_2, \|v\|_2^2 \leq 1, \|u\|_2^2 \leq 1.$$ (6)

Problem (6) is biconvex in $u$ and $v$, and it can be solved by iteratively updating $u$ and $v$. At the $k$th iteration, the update for $u$ is simply $\frac{Xv(k-1)}{\|Xv(k-1)\|_2^2}$. To update $v$, we need to solve:

$$\min_{v \in \mathbb{R}^p} -u_{(k)}^T X v \quad \text{subject to } v^T H v \leq c_1, \|v\|_1 \leq c_2, \|v\|_2^2 \leq 1.$$ (7)

Because of the quadratic constraint on $v$, it is hard to solve (7) directly. We propose to use the bisection method, solving the following feasibility problem iteratively:

$$\text{find } v \quad \text{subject to } -u_k^T X v \leq t, \|v\|_1 \leq c_2, \|v\|_2^2 \leq 1,$$ (8)

where $t$ is an upper-bound for $-u_k^T X v$ and is updated in each iteration. Note that $H$ is positive semidefinite and (8) can be solved by alternating projection on the convex sets. Details for the projection are included in the Appendix. In summary, the algorithm to update $v$ is as follows:

**Algorithm 1: Bisection method for solving problem (7) and updating $v$**

1. Initialize $t_{up} = 0$ and $t_{low} = -\|X^T u_k\|_2$
2. Iterate until convergence:
   (a) \( t^* = (t_{up} + t_{low})/2 \)
   (b) \( t_{up} \leftarrow t^* \) if (8) is feasible for \( t^* \)
   (c) \( t_{low} \leftarrow t^* \) if (8) is not feasible for \( t^* \)
3. Let \( t = t_{up} \) and find \( v \) by solving (8)

The algorithm to solve (6) is as follows:

Algorithm 2: Finding the sparse principal component

1. Initialize \( v \) to be the solution of (2)
2. Iterate until convergence:
   (a) \( u \leftarrow \frac{Xv_{(k-1)}}{||Xv_{(k-1)}||^2} \)
   (b) Update \( v \) by implementing algorithm 1

5.4 Multiple principal components

In (2), obtaining multiple principal components is straightforward, as they are just the eigenvectors of \( Z \); for the sparse solution, (6) can only obtain the first sparse principal component. To obtain the other sparse principal components, we can update \( X \) sequentially with \( X_{(i+1)} = X_{(i)}(I - \hat{v}_i \hat{v}_i^T) \) for \( i = 1, \cdots \), and implement (6) on \( X_{(i+1)} \), where \( X_{(1)} = X \) and \( \hat{v}_i \) is the \( i \)th principal component.

5.5 Tuning parameters

5.5.1 Tuning \( \lambda \)

Let \( X \) denote the \( N \times p \) data matrix. \( l = v^T X^T K X v \) can be treated as a loss function to be minimized. We do 10-fold cross-validation to tune \( \lambda \):

(a) From \( X \), we construct 10 data matrices \( X_1, \cdots, X_{10} \), each of which is missing a non-overlapping one-tenth of the rows of \( X \)
(b) For each \( X_i \), \( i = 1, \cdots, 10 \), implement (2) and obtain \( v_i \)
(c) Calculate \( l_{cv} \). In \( X v \), we use \( v_i \) and the missing rows that are left out in \( X_i \), for \( i = 1, \cdots, 10 \)
(d) When \( \lambda \) increases from 0, \( l_{cv} \) usually decreases sharply and then either increases or becomes flat. In practice, we choose \( \lambda \) to be the “elbow” point of \( l_{cv} \).
5.5.2 Tuning $c_1$ and $c_2$

Because $c_1$ and $c_2$ capture different aspects of the data, we propose a two-step approach: first tune $c_1$, and then tune $c_2$ with $c_1$ fixed. This also greatly reduces the computational cost since tuning $c_2$ can be slow. To tune $c_1$:

(a) We follow the previous procedure to tune $\lambda$

(b) The best $\lambda$ is used to calculate $v$, as in (2)

(c) Let $c_1 = v^T X^T K X v$

To tune $c_2$, we follow Algorithm 5 in [55], which is based on matrix completion:

(a) From $X$, we construct 10 data matrices $X_1, \cdots, X_{10}$, each of which is missing a non-overlapping one-tenth of the elements of $X$

(b) For $X_1, \cdots, X_{10}$, fit (6) and obtain $\hat{X}_i = duv^T$, the resulting estimate of $X_i$ and $d = u^T X_i v$

(c) Calculate the mean squared errors of $\hat{X}_i$, for $i = 1, \cdots, 10$, using only the missing entries

(d) Choose $c_2$ that minimizes the sum of mean squared errors

5.6 Data preprocessing

The human brain exon array dataset was downloaded from the Gene Expression Omnibus (GEO) database under the accession number GSE25219. The dataset was generated from 1,340 tissue samples collected from 57 developing and adult post-mortem brain[31]. Details for the quality control (QC) of the dataset are described as in [31]. After the QC procedures, noise reduction was accomplished by removing genes that were not expressed [37], leaving 13,720 genes in the dataset. We next selected the top 5,000 genes sorted by coefficient of variation and centered the expression levels of this consort.

5.7 Conservation and heterozygosity scores

The dN/dS score for cross-species conservation was calculated using Ensembl BioMart [13]. The heterozygosity score was calculated using 1,000 Genomes phase 1 version 3 [11]. Let $f_1, \cdots, f_p$ denote the allele frequencies for the $p$ non-synonymous coding SNPs in a gene, and let $l$ denote the maximum transcript length over all isoforms of that gene. The heterozygosity score was calculated as: $2 \sum_{i=1}^{p} f_i (1 - f_i)/l$. For a gene with low heterozygosity score, the non-synonymous variants in that gene tend to be rare, which indicates the functional importance of that gene.
6 Acknowledgement

Zhixiang Lin and Hongyu Zhao were supported in part by the National Science Foundation grant DMS-1106738 and the National Institutes of Health grants R01 GM59507 and P01 CA154295. Ying Zhu, Xuming Xu, Mingfeng Li and Nenad Sestan were supported by the National Institutes of Health grants P50 MH106934 and U01 MH103339. We also thank Matthew W. State for the partial financial support of Zhixiang Lin. Can Yang was supported by grant No.61501389 from National Science Funding of China, grant No.22302815 from the Hong Kong Research Grant Council, and grant FRG2/14-15/069 from Hong Kong Baptist University the Hong Kong RGC grant HKBU. All computations were performed on the Yale University Biomedical High Performance Computing Center.

References


Appendix A  Connection with Canonical Correlation Analysis (CCA)

Without loss of generality, let $n = 2$, then the objective function in formulation (1) becomes:

$$v^T X^T X v - 2\lambda v^T (X_2 - X_1)^T (X_2 - X_1) v = (1 - 2\lambda) v^T X^T X v + 4\lambda v^T X_2^T X_1 v$$
In Canonical Correlation Analysis (CCA), there are two datasets $X$ and $Y$, and the goal is to maximize the correlation between $X$ and $Y$ after projections. The objective function in CCA is $a^T X^T Y b$, where $a$ and $b$ are two column vectors. Note that $a^T X^T Y b$ and $v^T X_2^T X_1 v$ have similar forms: in $v^T X_2^T X_1 v$, the projection vectors $a$ and $b$ are the same as $v$. Therefore the objective function in (1) represents a balance between regular PCA and CCA. When $\lambda > 0.5$, the weight on PCA is negative.

Appendix B Hilbert-Schmidt independence criterion

The linear kernel of $X_v$ is $L = X v v^T X^T$. Let $K$ be the kernel of $Y$. Let $H = I - N^{-1} ee^T$, where $e$ is a column vector with all 1s. Then $H^T X = HX = X$, as $X$ is centered. The empirical Hilbert-Schmidt independence criterion for $X_v$ and $Y$ is:

$$
\text{Tr}(HLHK) = \text{Tr}(HX v v^T X^T H K)
$$

$$
= \text{Tr}(v^T X^T H K H X v)
$$

$$
= \text{Tr}(v^T X^T K X v)
$$

$$
= v^T X^T K X v
$$

Appendix C Details for the projections

Projection onto $\ell^2$ ball

First do decomposition $K = \Delta \Delta^T$, and let $M = \Delta^T X$. We have $M^T M = X^T \Delta \Delta^T X = X^T K X$.

We need to solve the following optimization problem:

$$
\text{minimize} \quad \|v - v_0\|^2_2 \quad \text{subject to} \quad \|M v\|^2_2 \leq c_1
$$

It is equivalent to the following Lagrangian problem:

$$
L(\lambda, v) = \|v - v_0\|^2_2 + \lambda(\|M v\|^2_2 - c_1), \text{ where } \lambda \geq 0.
$$

For any fixed $\lambda$, $L(\lambda, v)$ is a convex differentiable function of $v$. By taking derivative, it can be shown that $v^\star \equiv v^\star(\lambda) = (I_p + \lambda M^T M)^{-1} v_0$ minimizes the Lagrangian. Then we have

$$
g(\lambda) \equiv \inf_v L(\lambda, v) = L(\lambda, v^\star)
$$

By singular value decomposition, $M = U \Sigma V^T$, where $U^T U = U U^T = I_N$ and $V^T V = V V^T = I_p$. Then we have

$$
v^\star = (I_p + \lambda V \Sigma^T \Sigma V^T)^{-1} v_0 = V(I_p + \lambda \Sigma^T \Sigma)^{-1} V^T v_0 \quad (9)
$$
and
\[
g(\lambda) = \|V(I_p - (I_p + \lambda \Sigma^T \Sigma)^{-1})V^Tv_0\|_2^2 + \lambda\|U\Sigma(I_p + \lambda \Sigma^T \Sigma)^{-1}V^Tv_0\|_2^2 - \lambda c_1 \\
= \|(I_p - (I_p + \lambda \Sigma^T \Sigma)^{-1})V^Tv_0\|_2^2 + \lambda\|\Sigma(I_p + \lambda \Sigma^T \Sigma)^{-1}V^Tv_0\|_2^2 - \lambda c_1 \\
= \sum_{i=1}^{d} \left(\frac{\lambda}{1 + \lambda \sigma_i^2}\right) V^Tv_0^2_i + \lambda\sum_{i=1}^{d} \left(\frac{\sigma_i}{1 + \lambda \sigma_i^2}\right)^2 (V^Tv_0^2_i - \lambda c_1) \\
= \sum_{i=1}^{d} \frac{\lambda}{1 + \lambda \sigma_i^2} (V^Tv_0^2_i - \lambda c_1),
\]

where \(d(\leq N)\) is the number of non-zero singular values of \(M\), \(\sigma_i\) is the \(i\)th non-zero singular value, and \((\cdot)_i\) represents the \(i\)th element of a vector. When \(\lambda \geq 0\), \(g(\lambda)\) is concave and the optimal value \(\lambda^*\) can be found by Newton-Raphson’s Method. With \(\lambda = \lambda^*\), the projection \(v\) can be calculated with (9), where the inversion part is a diagonal matrix. It can be shown that only the first \(d\) columns of \(V\) affect the projection and typically we have \(d \ll p\).

**Projection onto \(\ell_1\) ball**

\[
\begin{align*}
\text{minimize} & \quad \|v - v_0\|_2^2 & & \text{subject to} & \|v\|_1 \leq c_2 \\
\end{align*}
\]

This can be solved efficiently by the algorithm presented in [16].

**Projection onto hyperplane**

\[
\begin{align*}
\text{minimize} & \quad \|v - v_0\|_2^2 & & \text{subject to} & -u_k^T X v \leq t \\
\end{align*}
\]

The solution is:
\[
v = v_0 - \frac{u_k^T X v_0 + t}{\|X^T u_k\|_2^2} X^T u_k
\]