SUPPLEMENT TO "JOINT AND INDIVIDUAL VARIATION EXPLAINED (JIVE) FOR INTEGRATED ANALYSIS OF MULTIPLE DATA TYPES"

By Eric F. Lock, Katherine A. Hoadley, J.S. Marron and Andrew B. Nobel

This supplementary article provides additional details and further validation of the JIVE method. In Section 1 we elaborate on the existence and uniqueness of the decomposition. In Section 2 we describe a permutation approach to rank selection. In Section 3 we give psuedocode for the JIVE algorithm. In Section 4 we discuss computing time and efficiency. In Section 5 we discuss invariance properties of the decomposition. In Section 6 we give results from the application of JIVE to many diverse simulated datasets.

1. Notes on Orthogonality. Enforcing orthogonality between joint and individual structure in the JIVE decomposition does not constrain the solution, and is a sufficient statement of condition for uniqueness of each component under mild assumptions. Here we formally present and prove this result.

Theorem 1.1. *Let*

$$T = \begin{bmatrix} T_1 \\ \vdots \\ T_k \end{bmatrix} = \overbrace{\begin{bmatrix} J_1 \\ \vdots \\ J_k \end{bmatrix}}^{J} + \overbrace{\begin{bmatrix} A_1 \\ \vdots \\ A_k \end{bmatrix}}^{A},$$

where $rank(T_i) = rank(J) + rank(A_i)\ \forall i$. *Then,*

1. *There exists* $J^\perp$, $A^\perp$ *such that*

$$T = \overbrace{\begin{bmatrix} J_1^\perp \\ J_2^\perp \\ \vdots \\ J_k^\perp \end{bmatrix}}^{J^\perp} + \overbrace{\begin{bmatrix} A_1^\perp \\ A_2^\perp \\ \vdots \\ A_k^\perp \end{bmatrix}}^{A^\perp},$$

$rank(J^\perp) = rank(J)$, $rank(A_i^\perp) = rank(A_i)\ \forall i$, *and* $J^\perp(A^\perp)' = 0_{p\times p}$.

2. *If* $\mathrm{row}(A_1) \cap \ldots \cap \mathrm{row}(A_k) = \emptyset$*, then* $J^\perp$ *and* $A^\perp$ *in (1) are uniquely defined.*

Proof.

1. Let P_J define the projection matrix onto the row space of J, $\mathrm{row}(J)$. For $i = 1, \ldots, k$, define

$$J_i^\perp = J_i + A_i P_J \quad \text{and} \quad A_i^\perp = A_i(I - P_J).$$

Then, $J_i^\perp + A_i^\perp = J_i + A_i P_J + A_i - A_i P_J = J_i + A_i \ \forall i$, and hence

$$T = \begin{bmatrix} J_1^\perp \\ \vdots \\ J_k^\perp \end{bmatrix} + \begin{bmatrix} A_1^\perp \\ \vdots \\ A_k^\perp \end{bmatrix}.$$

Furthermore,

$$J^\perp A^{\perp'} = J^\perp (I - P_J) A^{\perp'} = (J^\perp - J^\perp) A^{\perp'} = 0_{p \times p}.$$

Note $\mathrm{row}(J^\perp) \subseteq \mathrm{row}(J)$, so $\mathrm{rank}(J^\perp) \leq \mathrm{rank}(J)$.
Also $\mathrm{rank}(A_i^\perp) \leq \min\{\mathrm{rank}(A_i), \mathrm{rank}(I - P_J)\} \leq \mathrm{rank}(A_i)$.
Hence, as

$$\mathrm{rank}(J) + \mathrm{rank}(A_i) = \mathrm{rank}(T_i) = \mathrm{rank}(J^\perp) + \mathrm{rank}(A_i^\perp),$$

$\mathrm{rank}(J^\perp) = \mathrm{rank}(J)$ and $\mathrm{rank}(A_i^\perp) = \mathrm{rank}(A_i) \ \forall i$.

2. Assume

$$T = \overbrace{\begin{bmatrix} \tilde{J}_1 \\ \tilde{J}_2 \\ \vdots \\ \tilde{J}_k \end{bmatrix}}^{\tilde{J}} + \overbrace{\begin{bmatrix} \tilde{A}_1 \\ \tilde{A}_2 \\ \vdots \\ \tilde{A}_k \end{bmatrix}}^{\tilde{A}},$$

where $\mathrm{rank}(\tilde{J}) = \mathrm{rank}(J)$, $\mathrm{rank}(\tilde{A}_i) = \mathrm{rank}(A_i) \ \forall i$, and $\tilde{J}(\tilde{A})' = 0_{p \times p}$. Then,

$$\mathrm{rank}(\tilde{J}_i) = \mathrm{rank}(T_i) - \mathrm{rank}(\tilde{A}_i) = \mathrm{rank}(\tilde{J}).$$

So, as $\mathrm{row}(\tilde{J}_i) \subseteq \mathrm{row}(\tilde{J}) \ \forall i$,

$$\mathrm{row}(\tilde{J}_1) = \ldots = \mathrm{row}(\tilde{J}_k) = \mathrm{row}(\tilde{J}).$$

Note that

$$\tilde{J}_i = \tilde{J}_i P_{J_i^\perp} + \tilde{J}_i P_{A_i^\perp} \quad \forall i.$$

We will show $\tilde{J}_i P_{A_i^\perp} = 0$.
First, take $c' A_i^\perp \in \text{row}(\tilde{J}_i P_{A_i^\perp})$. Then, for any $j = 1, ..., k$,

$$\text{row}(c' A_i^\perp) \subseteq \text{row}(\tilde{J}) \subseteq \text{row}(T_j) = \text{row}(J^\perp) \cup \text{row}(A_j^\perp),$$

hence there exists a_j, b_j such that

$$\begin{aligned} a_j' J^\perp + b_j' A_j^\perp &= c' A_i^\perp \\ \rightarrow a_j' J^\perp &= c' A_i^\perp - b_j' A_j^\perp \\ \rightarrow a_j' J^\perp &= 0, \end{aligned}$$

since $\text{row}(J^\perp) \perp \{\text{row}(A_i^\perp) \cup \text{row}(A_j^\perp)\}$. So,

$$b_j' A_j^\perp = c' A_i^\perp \quad \forall j.$$

Note $\text{row}(A_i^\perp) \subseteq \text{row}(A_i)\,\forall\, i$, so by assumption $\text{row}(A_1^\perp) \cap \ldots \cap \text{row}(A_k^\perp) = \emptyset$ and hence $\text{row}(c' A_i^\perp) = 0$. So $\tilde{J}_i P_{A_i^\perp} = 0$, and hence $\tilde{J}_i = \tilde{J}_i P_{J^\perp}$, $\forall i$. So we conclude $\text{row}(\tilde{J}) \subseteq \text{row}(J^\perp)$, and by an analogous argument $\text{row}(J^\perp) \subseteq \text{row}(\tilde{J})$. It follows that $J_i^\perp = \tilde{J}_i\ \forall i$ since

$$T_i P_{J^\perp} = (J_i^\perp + A_i^\perp) P_{J^\perp} = J_i^\perp$$

and

$$T_i P_{J^\perp} = (\tilde{T}_i + \tilde{A}_i) P_{\tilde{J}} = \tilde{J}_i.$$

Furthermore, $A_i^\perp = T_i - J_i^\perp = T_i - \tilde{J}_i = \tilde{A}_i\ \forall i.$ □

2. Rank Selection.

The choice of ranks is important to accurately quantify the amount of joint variation and individual structure among data types. Furthermore, over- or underestimation of what is joint can negatively affect estimation of what is individual, and vice-versa. Rank selection is integral to the estimation of joint and individual structure in JIVE.

Here, we introduce a permutation testing approach to rank selection. This extends the method described in Peres-Neto, Jackson and Somers (2005), which determines the number of significant principal components for a single matrix. Different permutation methods are used to estimate joint and individual structure. To test for joint structure, we permute columns within each data type (across all rows), which maintains the multivariate distibution of each data type while effectively removing between-data-type associations. To test for individual structure, we permute columns within each row of a data type, which maintains the distribution of each variable while effectively removing between-variable associations.

Our initial estimate for the rank of joint structure is determined by comparing the singular values of the original concatenated matrix with the singular values after permuting the columns within each data type. The rank r is chosen such that for $j \leq r$, the j'th singular value in the original data is greater than the $100(1-\alpha)$ percentile of the distribution of the j'th singular value under permutation. The parameter α is a significance threshold that is applied to each of the singular values.

(1) To estimate r, with n_perm permutations and $\alpha \in (0,1)$ (by default, n_perm $= 100$ and $\alpha = 0.05$):
   (a) Let λ_j be the j'th singular value of $X = [X_1' \dots X_k']'$, $i = 1, \dots, \text{rank}(X)$.
   (b) Permute the columns within each X_i, and calculate the singular values of the resulting concatenated matrix. Repeat n_perm times.
   (c) Let λ_i^{perm} be the $100(1-\alpha)$ percentile among the j'th singular values after permutation.
   (d) Choose r to be the largest integer such that $\forall j \leq r, \lambda_j > \lambda_j^{\text{perm}}$.

Similarly, our initial estimate for the rank of the individual structure for a data type is determined by comparing the singular values of the original matrix with the singular values after permuting columns within each row of the data type.

(2) To estimate r_j, with n_perm permutations and significance level α:
   (a) Let λ_j be the j'th singular value of X_i.
   (b) Permute the columns separately within each row of X_j, and calculate the singular values of the permuted matrix. Repeat n_perm times.
   (c) Let λ_j^{perm} be the $100(1-\alpha)$ percentile among the j'th singular values after permutation.
   (d) Choose r_i to be the largest integer such that $\forall j \leq r_i, \lambda_j > \lambda_j^{\text{perm}}$.

JIVE estimates J and $A = [A_1' \dots A_k']'$ are found using the initial ranks for joint and individual structure. Then procedure (1) is used to re-estimate the rank of joint structure r^{new} after removing individual structure $X - A$. Similarly, procedure (2) is used to re-estimate the ranks of individual structure $r_1^{\text{new}}, \dots, r_k^{\text{new}}$ after removing joint structure $X - J$. If $r \neq r^{\text{new}}$ or $r_i \neq r_i^{\text{new}}$ for some $i = 1, \dots, k$, then re-compute J and A under the new ranks and repeat the rank estimation process. Otherwise, choose J and A to be the final estimates of joint and individual structure.

3. Pseudocode: Estimation.

Pseudocode for the iterative estimation procedure is given below.

```
• Initialize $X^{\text{Joint}} = X = [X_1' \ldots X_k']'$
• Loop:
    – Estimate $J = [J_1' \ldots J_k']'$ by a rank $r$ SVD of $X^{\text{Joint}}$ $(J = U\Lambda V')$
    – For $i = 1, \ldots, k$:
        ∗ Set $X_i^{\text{Individual}} = X_i - J_i$
        ∗ Estimate $A_i$ by a rank $r_i$ SVD of $X_i^{\text{Individual}}(I - VV')$
        ∗ Set $X_i^{\text{Joint}} = X_i - A_i$
    – Set $X^{\text{Joint}} = [X_1^{\text{Joint}\prime} \ldots X_k^{\text{Joint}\prime}]'$
```

The use of the projection matrix $I - VV'$ in the estimation of individual structure ensures that the orthogonality constraint is satisfied.

4. Computing Time and Efficiency.

The time needed to compute JIVE estimates can depend on several factors, primarily the dimensions of the data and ranks of joint and individual structure. A JIVE analysis of the illustrative simulation described in the main article takes <1 minute on a 2.67 GHz CPU with 8GB RAM, as do each of the simulated examples in Section 6.2. Analysis of large genomic datasets such as that described in the main article, and the simulation in Section 6.1, take longer (5 minutes and 3 minutes, respectively).

For high-dimensional data, where $p_i > n$, computing time can be improved substantially by reducing the dimensionality of $X_1, \ldots, X_k$ at the outset. We consider a dimension-reducing transformation of the original data: $X_i \to X_i^{\perp}$ where $X_i^{\perp}$ is an $n \times \text{rank}(X_i)$ matrix derived from the SVD of X_i (note $\text{rank}(X_i) \leq n$). That is,

$$X_i^{\perp} = \Lambda_i V_i',$$

where the entries of the diagonal matrix Λ_i are the non-zero singular values of X_i and the columns of V_i are the corresponding right-singular vectors. Covariance and Euclidian distance between columns (samples) of X_i are preserved in $X_i^{\perp}$. Applying the JIVE algorithm to the transformed datasets $X_1^{\perp}, \ldots, X_k^{\perp}$ can be substantially faster for high-dimensional data. For example, model estimation for the GBM data takes 188 minutes before transformation and 5 minutes after transformation.

Estimated joint ($J_i^\perp$) and individual ($A_i^\perp$) structure for $X_i^\perp$ can then be transformed back to the original variable space through the left singular vectors U_i: $J_i = U_i J_i^\perp$ and $A_i = U_i A_i^\perp$. Applying the iterative estimation method to X directly or estimating joint and individual structure for $X_i^\perp$ and mapping back to the original variable space will yield identical results. Hence, high-dimensional data are always transformed via SVD before estimation of joint and individual structure.

5. Invariance Properties.

The JIVE estimates J and A have several basic invariance properties. The following properties follow directly from analogous results for the SVD.

1. *Scaling.* If $c \in \mathbb{R}$, then $X \to cX$ implies $J \to cJ$ and $A \to cA$.
2. *Right multiplication by an orthogonal matrix.* If Q is $n \times n$ and $Q'Q = QQ' = I$, then $X \to XQ$ implies $J \to JQ$ and $A \to AQ$.
3. *Left multiplication of a single data type by an orthogonal matrix.* If Q_i is $p_i \times p_i$ and $Q_i'Q_i = Q_iQ_i' = I$, then $X_i \to Q_iX_i$ implies $J_i \to J_iQ_i$ and $A_i \to A_iQ_i$.

Note that property (2) entails that the JIVE estimates are invariant under column permutation, so long as the same permutation is applied to each data type. Property (3) entails that the JIVE estimates are invariant under row permutation, so long as the rows are permuted within a single data type. In general, JIVE estimates are not invariant under right multiplication of X_i by an orthogonal matrix or left multiplication of X by an orthogonal matrix.

6. Further Validation of the Method.

Here we further demonstrate the use of JIVE. In Section 6.1, we add an artificial joint signal to real data from three genomic platforms with complex structure and apply JIVE to identify this signal. In Section 6.2, the algorithm is applied to 200 diverse simulated datasets, in order to demonstrate its robustness.

6.1. *Example Based on Genomic Data.*

We now use real genomic data to illustrate the extraction of a known joint signal from data types with realistic individual variation. We begin with copy number variation (CN), gene expression (GE), and microRNA (miRNA) data available for the same set of 234 GBM tumor tissue samples from TCGA. The CN and GE data are both of dimension 14,556 (expression intensity and average copy number variation for the same 14,556 genes); the miRNA data are of dimension 534 (534 miRNA intensities measured). The three datasets are scaled as in Section **??**.

These three genomic datasets are used to simulate data with realistic background variation (see Figure 1). First, samples (columns) are randomly permuted separately within each dataset. This effectively removes joint structure between data types, but maintains the individual structure within each data type. An artificial joint signal is then added to 5% of the rows in each of the three data types. This joint signal is generated by adding a constant equal to the row standard deviation to the first half of the samples, and subtracting the same constant from the other half. This yields two sample clusters that are common to each of the three data types. We would like to identify this common signal among the variation individual to each data type.

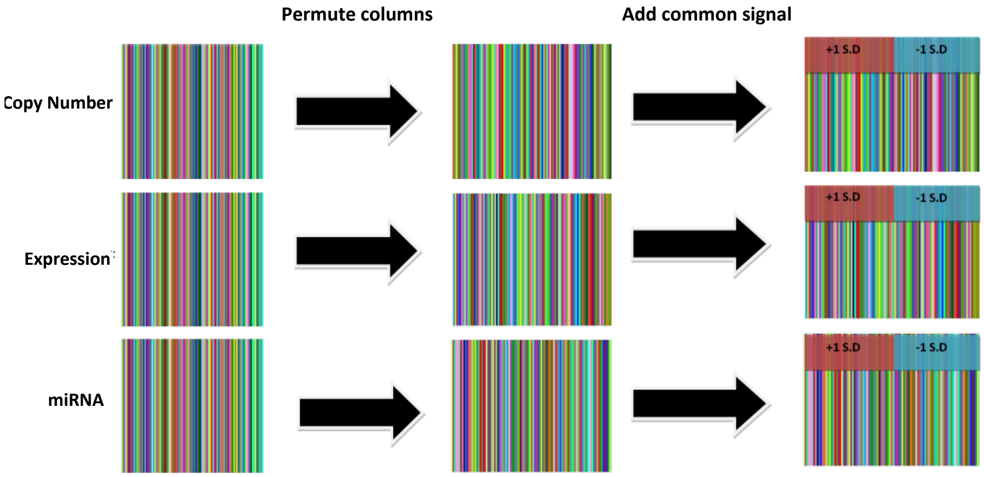


Fig 1. *Simulation based on CN, GE and miRNA data. The 234 columns are permuted within each data type so that samples are not associated. A common signal is then added to* 5% *of the rows in each data type.*

The first Consensus PC of the three datasets is shown in Figure 2. It shows a slight association between the two joint clusters, but does not clearly separate them. In this case, the joint signal does not dominate the variation in the data, and is hard to extract without accounting for individual structure. Application of PLS and CCA is complicated by the fact that these methods are designed for two data types. Moreover, the pairwise application of these methods is also negatively affected by individual structure and over-fitting.

A JIVE analysis of the data successfully distinguishes the joint and individual structure. The permutation approach described in Section 2 suggests

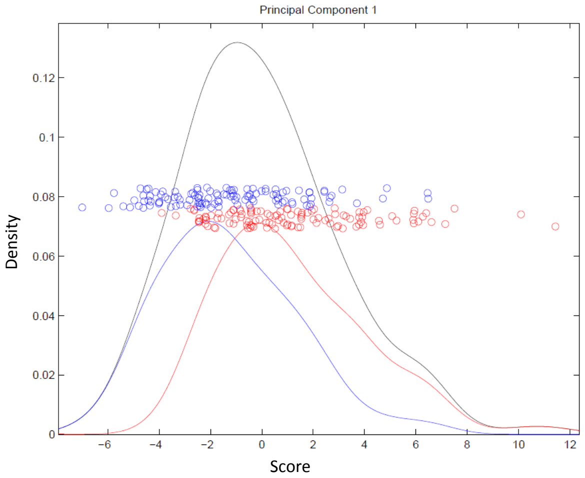


FIG 2. *Scores for the first Consensus PC of the concatenated data, colored by cluster. Kernel density curves are shown for all scores (black) and the two clusters (red and blue). Vertical jitter was added to the (one-dimensional) scores so that each point appears clearly. The two clusters (representing common signal) are not well distinguished.*

estimating rank 1 joint structure, and rank 39, 35, and 13 individual structure for GE, CN and miRNA, respectively. Sample scores and CN, GE and miRNA loadings for the joint component are shown in Figure 3. Joint scores now clearly separate the two artificial clusters, and the loadings help to show which rows contain the joint signal in each of the three data types.

6.2. *Extensive Simulations from Model.* To test the robustness of the iterative estimation method, we apply it to a diverse set of 200 randomly generated models. In our simulations, two data matrices X_1 and X_2 are generated with varying sample size and dimensions. Joint and individual structure are generated from diverse probability distributions, and have varying ranks. These randomly generated models are tested both with and without noise.

We first test models with joint and individual structure, and no noise. The sample size n and dimensions p_1 and p_2 for X_1 and X_2 are drawn at random uniformly from $\{10,11,...,100\}$. The ranks of joint structure, r, and individual structure, r_1 and r_2, are each drawn from $\{0,1,...,4\}$. Low-rank structure is then generated in factorized form:

$$\begin{aligned} X_1 &= U_1 S + W_1 S_1 \\ X_2 &= U_2 S + W_2 S_2. \end{aligned}$$

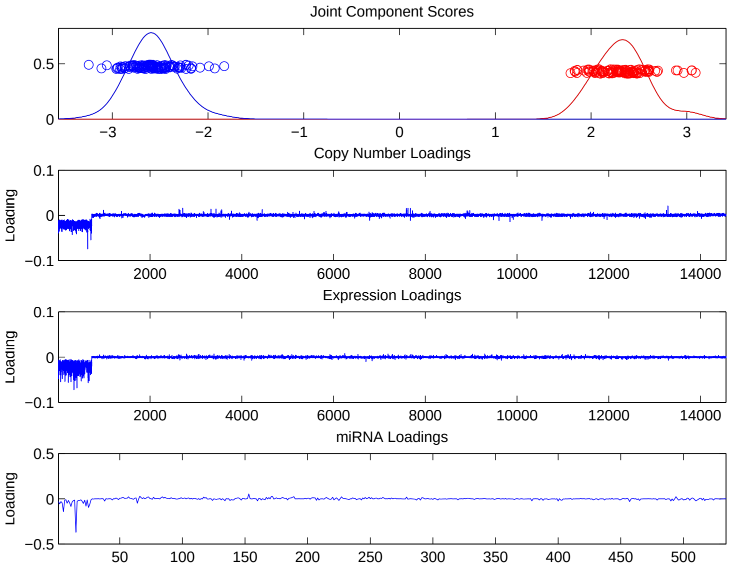


FIG 3. *Joint component scores and loadings. Scores are colored by artificial cluster and show good separation. Loadings for CN, GE, and miRNA are ordered so that the first* 5% *have joint signal added, and these tend to show the strongest contribution (difference from 0). Additional variation in the loadings is a result of the noise and complexity of the data.*

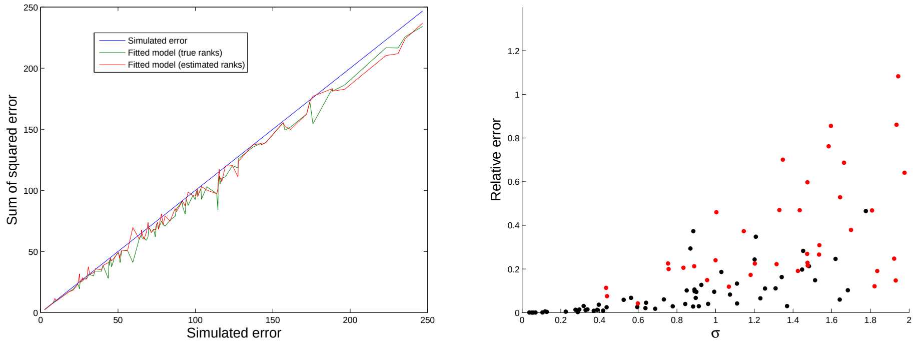
For each realization of $U_1, U_2, S, W_1, W_2, S_1$, and S_2, the entries are generated from a random choice among the distributions $\mathrm{N}(0,1)$, $\mathrm{Uniform}(0,1)$ and $\mathrm{Bernoulli}(\frac{1}{2})$.

We applied the iterative estimation method to X_1 and X_2 for 100 simulated examples, generated as described above. In all cases, the fitted model with the true ranks r, r_1 and r_2 converged to the simulated data X_1 and X_2 ($||X - \hat{X}||^2 < 10^{-12}$). This is evidence that, in the absence of noise, low-rank joint and individual structure can be recovered exactly.

We then include noise in a separete set of 100 independently generated simulations, using the model

$$\begin{aligned} X_1 &= U_1 S + W_1 S_1 + E_1 \\ X_2 &= U_2 S + W_2 S_2 + E_2, \end{aligned}$$

where $U_1, U_2, S, W_1, W_2, S_1, S_2$ are generated as above, and E_1, E_2 are error matrices with independent entries from $N(0, \sigma^2)$. The standard deviation, σ, is randomly determined from a $\mathrm{Uniform}(0,2)$ distribution. Joint and individual structure are estimated given the true ranks r, r_1 and r_2, and also with ranks estimated by the approach described in Section 2. We expect



(a) (b)

Fig 4. *Panel (a) shows the sum of squared errors in the simulated model (blue), the sum of squared residuals in the fitted model with true ranks (green), and the sum of squared residuals in the fitted model under permutation testing (red) in 100 randomly generated simulations, ordered by the amount of simulated error. Panel (b) shows the relative error in estimated joint and individual structure vs. the standard deviation of the noise (σ). Simulations in which the true ranks are not estimated correctly are colored red.*

the sum of squared residuals after estimating joint and individual structure, $||R_1||^2 + ||R_2||^2$, to be close to the sum of squared errors, $||E_1||^2 + ||E_2||^2$.

Figure 4(a) shows the sum of squared errors in the simulated model and the sum of squared residuals from the fitted model, with the true ranks and with estimated ranks, for the 100 examples. When the true ranks are used, the sum of squared residuals for the fitted model is always smaller than the sum of squared errors in the simulation. This is evidence that the iterative approach is successful in minimizing the sum of squared residuals. Figure 4(b) shows a measure of relative error in the JIVE estimates:

$$\text{Relative error} = \frac{||\hat{J} - J||^2 + ||\hat{A} - A||^2}{||J||^2 + ||A||^2}.$$

Simulations in which the true ranks are not estimated correctly (for either r, r_1, or r_2) are identified. The estimated joint and individual components, and the estimated ranks, generally lose accuracy as the noise level increases. Estimated ranks for joint and individual structure agree completely with the true ranks in 62% of simulations. The permutation testing approach tends to underestimate the simulated ranks, as noise can overwhelm the low-rank signal. The rank of joint structure was underestimated in 20% of simulations and overestimated in just 1% of simulations; individual structure was underestimated in 18% of simulations and overestimated in 13% of simulations.

References.

Peres-Neto, P., Jackson, D. and Somers, K. How many principal components? stopping rules for determining the number of non-trivial axes revisited. (2005). *Computational Statistics & Data Analysis* **49** 974-997.