Partitioned Factor Analysis for Interference Suppression and Source Extraction

Srikantan S. Nagarajan[1], Hagai T. Attias[2], Kensuke Sekihara[3], and Kenneth E. Hild II[1]

[1] Dept. of Radiology, University of California at San Francisco, CA 94122, USA
Srikantan.Nagarajan@radiology.ucsf.edu, k.hild@ieee.org
[2] Golden Metallic Inc., San Francisco, CA 94147, USA
htattias@goldenmetallic.com
[3] Dept. of Electronic Systems and Engineering,
Tokyo Metropolitan Institute of Technology, 191-0065, Japan
ksekiha@cc.tmit.ac.jp

Abstract. It is common for data to be contaminated with artifacts, interference, and noise. Several methods including independent components analysis (ICA) and principal components analysis (PCA) have been used to suppress these undesired signals and/or to extract the underlying (desired) source waveforms. For some data it is known, or can be extracted post hoc, how to partition the data into periods of source activity and source inactivity. Two examples include cardiac data and data collected using the stimulus-evoked paradigm. However, neither ICA nor PCA are able to take full advantage of the knowledge of the partition. Here we introduce an interference suppression method, partitioned factor analysis (PFA), that takes into account the data partition.

1 Introduction

Raw data are corrupted by artifacts, interference, and sensor noise. When the power of these undesired signals is large one of several signal processing techniques may be applied to either reduce the level of interference or to extract the underlying source waveforms directly (we use "denoising" and "interference suppression" interchangeably). Linear denoising methods, including independent components analysis (ICA), attempt to find the source subspace and produce denoised signals by projecting intermediate lower-dimensional data back into the space of the observations. Denoising is useful for spatio-temporal visualization and for source localization [1]. Furthermore, the intermediate data produced by denoising methods can be used as the input to an ICA algorithm when the desire is to extract the source waveforms. While ICA can be applied directly to the observations to perform denoising and source extraction simultaneously, many ICA algorithms are too computationally intensive to be used in this manner when there are many sensors. An alternative is to reduce the dimensionality with a (non-ICA) denoising method and then use ICA.

J. Rosca et al. (Eds.): ICA 2006, LNCS 3889, pp. 189–197, 2006.

Principal component analysis (PCA) is the most widely used method of denoising. PCA is an ideal choice when either the power of the sources is large relative to the power of all undesired signals or when the undesired signals are spatially uncorrelated and isotropic in sensor space [2]. Likewise, ICA is ideal for denoising when the sources of interest are statistically independent of all undesired signals, there are at least as many sensors as sources (desired and undesired), and the subspace that contains the sources of interest can be robustly determined automatically or with the aid of a human expert. However, these requisite conditions may not always be met. In addition, both PCA and ICA are unable to take full advantage of additional information that is available for some data, i.e., knowledge of when the sources are active and when they are inactive. Data for which the timing of the source activity is known or can be estimated include, e.g., cardiac data collected using a magnetocardiogram (MCG) and stimulus-evoked data collected using a magnetoencephalogram (MEG).

Here we introduce a denoising method, partitioned factor analysis (PFA), that is based on what is thought to be a more realistic set of assumptions than PCA or ICA and that is able to incorporate knowledge of the data partition.

2 Partitioned Factor Analysis

Generative model. The proposed method, PFA, is based on the following generative model,

$$\boldsymbol{y}_n = \begin{cases} \boldsymbol{B}\boldsymbol{u}_n + \boldsymbol{v}_n & n = 1, \ldots, N_0 - 1 \\ \boldsymbol{A}\boldsymbol{x}_n + \boldsymbol{B}\boldsymbol{u}_n + \boldsymbol{v}_n & n = N_0, \ldots, N \end{cases}, \tag{1}$$

where the $(M_y \times 1)$ vector $\boldsymbol{v}_n$ is used to represent all signals that are spatially uncorrelated in sensor space and that exist in the active and inactive periods, the $(M_u \times 1)$ vector $\boldsymbol{u}_n$ represents all signals that are spatially correlated in sensor space and that exist throughout, and the $(M_x \times 1)$ vector $\boldsymbol{x}_n$ represents all signals that exist only during the active period. We refer to $\boldsymbol{x}_n$ as the factors (which are an arbitrary linear combination of the sources of interest, $\boldsymbol{x}_n = \boldsymbol{W}\boldsymbol{s}_n$), $\boldsymbol{u}_n$ as the interference, and $\boldsymbol{v}_n$ as the noise. The inclusion of the interference signals allow us to model signals of no interest that, unlike the model commonly used for sensor noise, are spatially correlated in sensor space, e.g., respiration, muscle artifacts, eye blinks, and ongoing neural activity. Also, there is a common assumption that the spatial auto-correlation matrix of the post-stimulus equals the auto-correlation of the pre-stimulus plus the auto-correlation of the evoked response. This structure is directly reflected by the model of (1), which provides a more complete representation of the data than what is inherently assumed in PCA and the vast majority of ICA algorithms (which combine $\boldsymbol{x}_n$ and $\boldsymbol{u}_n$ into a single vector and assume that $\boldsymbol{v}_n$ is zero).

The proper choice for the active and inactive periods is problem-dependent. For example, if the goal is to recover a cardiac signal the active period should

correspond to all portions of the data that are near a QRS complex. Likewise, if the goal is to recover an evoked response for data collected using the stimulus-evoked paradigm, the active period corresponds to the post-stimulus period. To simplify the notation the data are assumed to ordered so that the first N_0-1 samples of the $(M_y \times N)$ matrix of observations $\boldsymbol{y}$ (where $\boldsymbol{y}$ corresponds to the collection of $\boldsymbol{y}_n \forall n$) correspond to the concatenation of all inactive periods and the remaining samples correspond to the concatenation of all active periods.

PFA probabilistic graphical model. Each factor $x_{m,n}$ and each interference $u_{m,n}$ is modeled as having a Gaussian probability density function (pdf) with zero mean and unit precision, where the precision is defined as the inverse variance and $x_{m,n}$ and $u_{m,n}$ are the m^{th} element of vectors $\boldsymbol{x}_n$ and $\boldsymbol{u}_n$, respectively. Likewise, the noise at sensor m is modeled as having a Gaussian pdf with zero mean and precision λ_m. We model the factors, interferences, and noises as mutually statistically independent,

$$p(\boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{0},\boldsymbol{I}), \quad p(\boldsymbol{u}_n) = \mathcal{N}(\boldsymbol{u}_n|\boldsymbol{0},\boldsymbol{I}), \quad p(\boldsymbol{v}_n) = \mathcal{N}(\boldsymbol{v}_n|\boldsymbol{0},\boldsymbol{\Lambda}) \ , \tag{2}$$

where $\boldsymbol{0}$ is a column vector of zeros, $\boldsymbol{I}$ is an identity matrix, and $\boldsymbol{\Lambda}$ is a diagonal matrix. By inspection the data likelihood is

$$p(\boldsymbol{y}_n|\boldsymbol{x}_n,\boldsymbol{u}_n,\boldsymbol{A},\boldsymbol{B}) = \begin{cases} \mathcal{N}(\boldsymbol{y}_n|\boldsymbol{B}\boldsymbol{u}_n,\boldsymbol{\Lambda}) & n=1,\ldots,N_0-1 \\ \mathcal{N}(\boldsymbol{y}_n|\boldsymbol{A}\boldsymbol{x}_n+\boldsymbol{B}\boldsymbol{u}_n,\boldsymbol{\Lambda}) & n=N_0,\ldots,N \end{cases} \ . \tag{3}$$

We also assume that the signals are temporally i.i.d. so that

$$p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B}) = \prod_{n=1}^{N} p(\boldsymbol{y}_n|\boldsymbol{x}_n,\boldsymbol{u}_n,\boldsymbol{A},\boldsymbol{B}), \ p(\boldsymbol{x}) = \prod_{n=N_0}^{N} p(\boldsymbol{x}_n), \ p(\boldsymbol{u}) = \prod_{n=1}^{N} p(\boldsymbol{u}_n) \ . \tag{4}$$

The elements of the two mixing matrices are assumed to be independent zero-mean Gaussians that have a precision that is proportional to the noise precision of the corresponding sensor,

$$p(\boldsymbol{A}) = \prod_{m=1}^{M_y}\prod_{k=1}^{M_x} \mathcal{N}(A_{m,k}|0,\lambda_m\alpha_k), \quad p(\boldsymbol{B}) = \prod_{m=1}^{M_y}\prod_{k=1}^{M_u} \mathcal{N}(B_{m,k}|0,\lambda_m\beta_k) \ , \tag{5}$$

where the proportionality constants, α_k, β_k, are referred to as hyperparameters. These priors are chosen so that they have the same functional form as the posterior distribution (when this is true the prior is referred to as a conjugate prior).

Inferring the PFA model from data. All three types of signals, $\boldsymbol{x}_n, \boldsymbol{u}_n, \boldsymbol{v}_n$, are unobserved, as are the $(M_y \times M_x)$ matrix $\boldsymbol{A}$ and the $(M_y \times M_u)$ matrix $\boldsymbol{B}$. Hence, PFA must use only $\boldsymbol{y}$ to infer the quantities of interest, which are $\tilde{\boldsymbol{y}}_n = \boldsymbol{A}\boldsymbol{x}_n$ for denoising and $\boldsymbol{x}_n$ for subsequent source extraction. To infer the model from $\boldsymbol{y}$ we use an extended version of the Expectation-Maximization (EM)

algorithm, which is known as the variational Bayesian EM method (VB-EM) [3]. Whereas standard EM computes the most likely parameter value given the data, i.e., the maximum a posteriori (MAP) estimate, VB-EM computes full posterior distributions. Furthermore, VB-EM provides a natural mechanism for inferring the model order through hyper-parameter optimization [3], whereas standard EM requires ad-hoc methods for model order selection.

Standard EM maximizes the log likelihood, which can be written as

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}) \int p(\boldsymbol{\theta}|\boldsymbol{y}) d\boldsymbol{\theta} = \int p(\boldsymbol{\theta}|\boldsymbol{y}) \log \frac{p(\boldsymbol{\theta},\boldsymbol{y})}{p(\boldsymbol{\theta}|\boldsymbol{y})} d\boldsymbol{\theta} \ , \tag{6}$$

where $\boldsymbol{\theta} = \{\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B}\}$. Since the exact posterior distribution is computationally intractable, we approximate the posterior with a function that factorizes the hidden variables given the data from the parameters given the data,

$$p(\boldsymbol{\theta}|\boldsymbol{y}) \approx q(\boldsymbol{\theta}|\boldsymbol{y}) = q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y}) q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y}) \ . \tag{7}$$

The result is that VB-EM adapts $q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y})$ and $q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y})$ to maximize an approximation of the log likelihood, which can be written as

$$\mathcal{F} = \int q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y}) q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y}) \log \frac{p(\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B},\boldsymbol{y})}{q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y}) q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y})} d\boldsymbol{x} d\boldsymbol{u} d\boldsymbol{A} d\boldsymbol{B} \ . \tag{8}$$

It can be shown that maximizing $\mathcal{F}$ w.r.t. $q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y})$, $q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y})$ is equivalent to minimizing the Kullback-Leibler divergence [4] between $p(\boldsymbol{\theta}|\boldsymbol{y})$ and $q(\boldsymbol{\theta}|\boldsymbol{y})$. Like standard EM the VB-EM optimization method is an iterative algorithm where each iteration is composed of an E-step and an M-step.

E-step. Maximization of $\mathcal{F}$ with respect to the posterior over hidden variables is accomplished by setting the derivative of $\mathcal{F}$ to zero and solving for $q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y})$ while keeping $q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y})$ fixed. This produces

$$q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y}) = \frac{1}{z_1} \exp[\int q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y}) \log p(\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B},\boldsymbol{y}) d\boldsymbol{A} d\boldsymbol{B}] \ , \tag{9}$$

where the joint pdf, due to the previous assumptions, simplifies to

$$p(\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B},\boldsymbol{y}) = p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B}) p(\boldsymbol{x}) p(\boldsymbol{u}) p(\boldsymbol{A}) p(\boldsymbol{B}) \ , \tag{10}$$

and where z_1 is the normalizing constant (normalization of this posterior and the posterior over parameters is enforced by adding two Lagrange multipliers to $\mathcal{F}$). The quantities in (10) are given by (2)-(5). The posterior over hidden variables factorizes over time so that

$$q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y}) = \prod_{n=1}^{N_0-1} q(\boldsymbol{u}_n|\boldsymbol{y}_n) \prod_{n=N_0}^{N} q(\boldsymbol{x}_n,\boldsymbol{u}_n|\boldsymbol{y}_n) \ , \tag{11}$$

where

$$
\begin{aligned}
q(\boldsymbol{u}_n|\boldsymbol{y}_n) &= \mathcal{N}(\boldsymbol{u}_n|\bar{\boldsymbol{u}}_n, \boldsymbol{\Phi}^{-1}) \\
q(\boldsymbol{x}_n,\boldsymbol{u}_n|\boldsymbol{y}_n) &= \mathcal{N}(\begin{bmatrix}\boldsymbol{x}_n \\ \boldsymbol{u}_n\end{bmatrix} | \begin{bmatrix}\bar{\boldsymbol{x}}_n \\ \bar{\boldsymbol{u}}_n\end{bmatrix}, \boldsymbol{\Gamma}^{-1}) \\
\bar{\boldsymbol{x}}_n &= (\boldsymbol{\Gamma}_{xx}\bar{\boldsymbol{A}}^T + \boldsymbol{\Gamma}_{xu}\bar{\boldsymbol{B}}^T)\boldsymbol{\Lambda}\boldsymbol{y}_n \\
\bar{\boldsymbol{u}}_n &= \boldsymbol{\Phi}\bar{\boldsymbol{B}}^T\boldsymbol{\lambda}\boldsymbol{y}_n \qquad\qquad n \in \{1,\ldots,N_0-1\} \\
\bar{\boldsymbol{u}}_n &= (\boldsymbol{\Gamma}_{xu}^T\bar{\boldsymbol{A}}^T + \boldsymbol{\Gamma}_{uu}\bar{\boldsymbol{B}}^T)\boldsymbol{\Lambda}\boldsymbol{y}_n \quad n \in \{N_0,\ldots,N\} \\
\boldsymbol{\Phi} &= (\bar{\boldsymbol{B}}^T\boldsymbol{\Lambda}\bar{\boldsymbol{B}} + \boldsymbol{I} + M_y\boldsymbol{\Psi}_{BB})^{-1} \\
\boldsymbol{\Gamma} &= (\begin{bmatrix}\bar{\boldsymbol{A}} \\ \bar{\boldsymbol{B}}\end{bmatrix}\boldsymbol{\Lambda}\begin{bmatrix}\bar{\boldsymbol{A}} & \bar{\boldsymbol{B}}\end{bmatrix} + \boldsymbol{I} + M_y\boldsymbol{\Psi})^{-1} = \begin{bmatrix}\boldsymbol{\Gamma}_{xx} & \boldsymbol{\Gamma}_{xu} \\ \boldsymbol{\Gamma}_{xu}^T & \boldsymbol{\Gamma}_{uu}\end{bmatrix} \ ,
\end{aligned}
\tag{12}
$$

and where $\bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}, \boldsymbol{\lambda}, \boldsymbol{\Psi}$ are computed in the M-step.

M-step. Similarly, maximization of $\mathcal{F}$ with respect to the posterior over parameters is accomplished by setting the derivative of $\mathcal{F}$ to zero and solving for $q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y})$ while keeping $q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y})$ fixed. This produces

$$
q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y}) = \frac{1}{z_2}\exp[\int q(\boldsymbol{x},\boldsymbol{u}|\boldsymbol{y})\log p(\boldsymbol{x},\boldsymbol{u},\boldsymbol{A},\boldsymbol{B},\boldsymbol{y})d\boldsymbol{x}d\boldsymbol{u}] \ , \tag{13}
$$

where z_2 is the normalizing constant.

It follows from (13) that the posterior over parameters factorizes over the rows of the two mixing matrices. Hence,

$$
q(\boldsymbol{A},\boldsymbol{B}|\boldsymbol{y}) = \prod_{m=1}^{M_y}\mathcal{N}(\begin{bmatrix}\boldsymbol{A}_m^T \\ \boldsymbol{B}_m^T\end{bmatrix} | \begin{bmatrix}\bar{\boldsymbol{A}}_m^T \\ \bar{\boldsymbol{B}}_m^T\end{bmatrix}, \lambda_m\boldsymbol{\Psi}^{-1}) \ , \tag{14}
$$

where $\boldsymbol{A}_m$ is the m^{th} row of $\boldsymbol{A}$ and

$$
\begin{aligned}
\bar{\boldsymbol{A}} &= \left(\sum_{n=N_0}^{N}\boldsymbol{y}_n\bar{\boldsymbol{x}}_n^T\right)\boldsymbol{\Psi}, \qquad \bar{\boldsymbol{B}} = \left(\sum_{n=1}^{N}\boldsymbol{y}_n\bar{\boldsymbol{u}}_n^T\right)\boldsymbol{\Psi} \\
\boldsymbol{\Psi} &= \begin{bmatrix}\boldsymbol{R}_{xx}+\alpha & \boldsymbol{R}_{xu} \\ \boldsymbol{R}_{xu}^T & \boldsymbol{R}_{uu}+\beta\end{bmatrix}^{-1} = \begin{bmatrix}\boldsymbol{\Psi}_{AA} & \boldsymbol{\Psi}_{AB} \\ \boldsymbol{\Psi}_{AB}^T & \boldsymbol{\Psi}_{BB}\end{bmatrix} \\
\boldsymbol{R}_{\boldsymbol{xx}} &= \sum_{n=N_0}^{N}\left(\bar{\boldsymbol{x}}_n\bar{\boldsymbol{x}}_n^T + \boldsymbol{\Gamma}_{\boldsymbol{xx}}\right), \qquad \boldsymbol{R}_{\boldsymbol{xu}} = \sum_{n=N_0}^{N}\left(\bar{\boldsymbol{x}}_n\bar{\boldsymbol{u}}_n^T + \boldsymbol{\Gamma}_{\boldsymbol{xu}}\right) \\
\boldsymbol{R}_{\boldsymbol{uu}} &= \sum_{n=1}^{N_0-1}\left(\bar{\boldsymbol{u}}_n\bar{\boldsymbol{u}}_n^T + \boldsymbol{\Phi}\right) + \sum_{n=N_0}^{N}\left(\bar{\boldsymbol{u}}_n\bar{\boldsymbol{u}}_n^T + \boldsymbol{\Gamma}_{\boldsymbol{uu}}\right) \ ,
\end{aligned}
\tag{15}
$$

and $\boldsymbol{\alpha}, \boldsymbol{\beta}$ are diagonal matrices that contain α_k, β_k, respectively.

The solutions for the noise precision matrix and the hyperparameters are found by computing the derivative of $\mathcal{F}$ and equating the result with zero,

$$\begin{aligned}\boldsymbol{\alpha}^{-1} &= \mathrm{diag}\left(\frac{1}{M_y}\bar{\boldsymbol{A}}^T\boldsymbol{\Lambda}\bar{\boldsymbol{A}} + \boldsymbol{\Psi}_{AA}\right)\\ \boldsymbol{\beta}^{-1} &= \mathrm{diag}\left(\frac{1}{M_y}\bar{\boldsymbol{B}}^T\boldsymbol{\Lambda}\bar{\boldsymbol{B}} + \boldsymbol{\Psi}_{BB}\right)\\ \boldsymbol{\Lambda}^{-1} &= \frac{1}{N}\mathrm{diag}\left(\boldsymbol{R}_{yy} - \bar{\boldsymbol{A}}\boldsymbol{R}_{yx}^T - \bar{\boldsymbol{B}}\boldsymbol{R}_{yu}^T\right)\ ,\end{aligned} \tag{16}$$

where

$$\boldsymbol{R_{yy}} = \sum_{n=1}^{N}\boldsymbol{y}_n\boldsymbol{y}_n^T, \quad \boldsymbol{R_{yx}} = \sum_{n=N_0}^{N}\boldsymbol{y}_n\bar{\boldsymbol{x}}_n^T, \quad \boldsymbol{R_{yu}} = \sum_{n=1}^{N}\boldsymbol{y}_n\bar{\boldsymbol{u}}_n^T\ . \tag{17}$$

3 Results

Denoising performance is measured using the output signal-to-noise/interference ratio (SNIR),

$$\mathrm{SNIR} = \frac{1}{M}\sum_{m=1}^{M} 10\log_{10}\frac{\sum_{n=1}^{N}(\boldsymbol{Ax})_{m,n}^2}{\sum_{n=1}^{N}((\boldsymbol{Ax})_{m,n} - (\bar{\boldsymbol{A}}\bar{\boldsymbol{x}})_{m,n})^2}\ \ (\mathrm{dB})\ .$$

For real data we replace $\boldsymbol{Ax}_n$ with the sensor signals due to the 5 principal components (representing 97% of the energy) of the average sensor data, where the average is taken over 525 trials. The metric for source extraction performance is the source-to-distortion ratio (SDR),

$$\mathrm{SDR} = \frac{1}{M_s}\sum_{m=1}^{M_s} 10\log_{10}\frac{1}{M_s}\sum_{m'=1}^{M_s}\left(\frac{1}{2 - \frac{2}{N-N_0+1}\left|\sum_{n=N_0}^{N} s_{m,n}\bar{s}_{m',n}\right|}\right)\ \ (\mathrm{dB})\ ,$$

where the distortion for source estimate m includes noise, interference, and all sources other than source m, $\boldsymbol{s}_n = \boldsymbol{W}^{-1}\boldsymbol{x}_n$, $\boldsymbol{W}$ is found using ICA, and both $s_{m,n}$ and $\bar{s}_{m,n}$ (the estimate of source m at time n) are normalized to have unit variance. For simulated data the SNIR and SDR are shown as a function of the input signal-to-interference ratio (SIR) for a fixed value of input signal-to-noise ratio (SNR). The former is defined as the ratio of the power of the factors to the power of the interferences measured in sensor space. The latter is defined in a similar fashion.

Simulated data. For the simulations $N = 1000$ data points/trial, $N_0 = 631$, $M_y = 132$, $M_x = M_s = 2$, $M_u = 1000$, SNR$= 0$ dB, and the number of trials is 10. The results represent the mean over 10 Monte Carlo experiments (per trial) and error

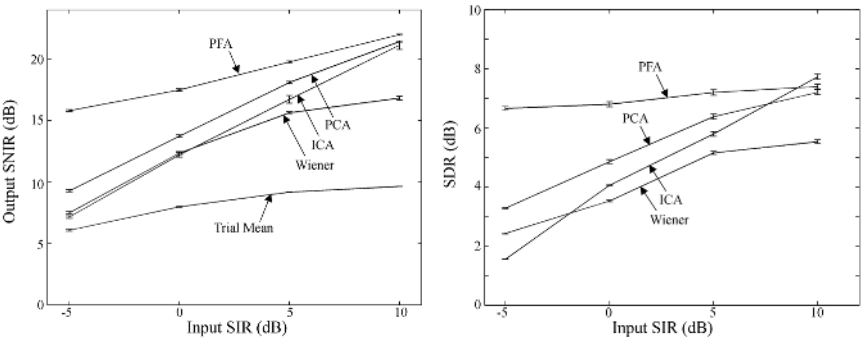

Fig. 1. Left subplot (1a): Output SNIR as a function of input SIR for 10 trials, input SNR = 0 dB, and the ICA method is TDSEP. Right subplot (1b): SDR as a function of input SIR for 10 trials, input SNR = 0 dB, and the ICA method is FastICA.

bars are used to indicate one standard error. The comparison includes the proposed method (PFA), PCA [2], Wiener [5], ICA (TDSEP [6] or FastICA [7]), and the mean over trials. For ICA, the source subspace is determined as the components having the largest ratio of active power to inactive power.

Figure 1a shows the denoising performance as a function of input SIR. All of the methods perform better than the trial mean. PFA performs the best across all values of input SIR. The performances of both PCA and Wiener approach that of PFA as the input SIR increases.

Figure 1b shows the SDR as a function of input SIR. In this figure PFA, PCA, and Wiener are all combined with ICA. Also shown is the result for ICA without dimension reduction. PFA produces the best overall results and is the least sensitive to input SIR. The results for ICA (with no dimension reduction) indicate that for this dataset dimension reduction should be used when the input SIR is low and is not needed if the input SIR is > 10 dB.

Real data. Figure 2a shows the denoising performance as a function of the number of trials for a real MEG dataset, which uses a somatosensory stimulus ($M_y = 274$, M_x is assumed to be 2, M_u is assumed to be 50, $N = 361$, $N_0 = 121$). PFA performs the best and both PCA and Wiener (which performs almost identically to PCA and is not shown here) outperform the trial mean.

Figure 2b shows the sensor signals before and after PFA denoising is applied to real fetal MCG data, which is a mixture of both fetal and maternal cardiac sources ($M_y = 51$, M_x is assumed to be 10, M_u is assumed to be 50, $N = 4000$, $N_0 = 501$). Since the goal is to recover the fetal cardiac factors, the inactive period is chosen such that it contains minimal activity of the fetal heart (two 250-length portions) and the active period is chosen such that it contains both maternal and fetal cardiac activity. Notice that QRS complexes of the mother are effectively suppressed and several fetal QRS complexes that were previously obscured, e.g., at 1.3 sec and 3.1 sec, are now visible.

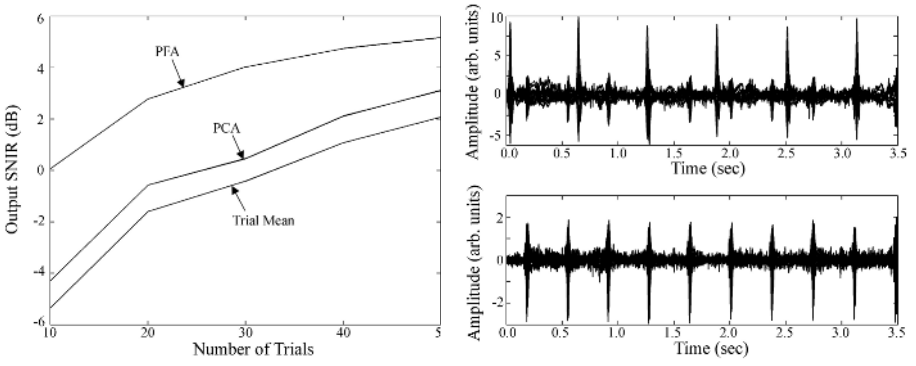

Fig. 2. Left subplot (2a):Output SNIR as a function of number of trials for real MEG data. Right subplot (2b): Observations before (upper subplot) and after (lower subplot) PFA denoising for real fetal MCG data.

4 Conclusions

The PFA graphical model for denoising and dimension reduction is introduced. This model takes into account additional information that is available for several types of data, including cardiac data and data collected using the evoked-response paradigm. The results of simulated and real data indicate that PFA may be a viable alternative to ICA for interference suppression and may, when used as a preprocessor, improve the performance of ICA for source extraction. This appears to be true especially when the power of the noise/interference is large or there are only a few trials available.

Acknowledgments

This work supported by National Institutes of Health grants 1 F32 NS 52048-01 and RO1 DC 4855.

References

1. Ossadtchi, A., Baillet, S., Mosher, J.C., Thyerlei, D., Sutherling, W., Leahy, R.M.: Automated interictal spike detection and source localization in magnetoencephalography using independent components analysis and spatio-temporal clustering. Clinical Neurophysiology **115** (2004) 508–522
2. Jackson, J.E.: A User's Guide to Principal Components (2003)
3. Attias, H.: A variational Bayesian framework for graphical models. Advances in Neural Information Proc. Systems **13** (2000) 209–215
4. Cover, T.M., Thomas, J.A.: Elements of Information Theory (1991)

5. Ungan, P., Basar, E.: Comparison of Wiener filtering and selective averaging of evoked potentials. Electroencephalography and Clinical Neurophysiology **40** (1976) 516–520
6. Ziehe, A., Mller, K.R.: TDSEP - an efficient algorithm for blind separation using time structure. Intl. Conf. on Artificial Neural Networks (1998) 675–680
7. Hyvarinen, A.: Fast and robust fixed-point algorithms for Independent Component Analysis. IEEE Trans. on Neural Networks **10** (1999) 626–634