CUR+NMF for Learning Spectral Features from Large Data Matrix

Hyekyoung Lee and Seungjin Choi

***Abstract*— Nonnegative matrix factorization (NMF) is a popular method for multivariate analysis of nonnegative data. It was successfully applied to learn spectral features from EEG data. However, the size of a data matrix grows, NMF suffers from 'out of memory' problem. In this paper we present a memory-reduced method where we downsize the data matrix using CUR decomposition before NMF is applied. Experimental results with two EEG data sets in BCI competition, confirm the useful behavior of the proposed method.**

I. Introduction

NONNEGATIVE matrix factorization (NMF) is a popular method for learning a faithful representation of nonnegative data [17], [24], which has been widely used in various applications, including face recognition [23], document clustering [26], [31], audio processing [4], [14], [28], medical imaging [21], bioinformatics [3], and so on. NMF was extended along diverse directions, including hierarchical NMF with a multilayer generative model [1], transformation-invariant NMF [10], and NMF with sparseness constraints [12], [25]. Multiway generalization of NMF, referred to as nonnegative tensor factorization, was developed using PARAFAC model [27], [29], PARAFAC2 model [5], and Tucker model [15].

The main application of NMF considered in this paper is brain computer interface (BCI) where the useful feature extraction from EEG plays an important role for successful EEG classification. Brain computer interface (BCI) is a system that is designed to translate a subject's intention or mind into a control signal for a device such as a computer, a wheelchair, or a neuroprosthesis [30]. BCI provides a new communication channel between human brain and computer and adds a new dimension to human computer interface (HCI). It was motivated by the hope of creating new communication channels for disabled persons, but recently draws attention in multimedia communication, too [9].

The most popular sensory signal used for BCI is electroencephalogram (EEG) which is the multivariate time series data where electrical potentials induced by brain activities are recorded in a scalp. Exemplary spectral characteristics of EEG involving motor, might be μ rhythm (8-12 Hz) and β rhythm (18-25 Hz) which decrease during movement or in preparation for movement (event-related desynchronization, ERD) and increase after movement and in relaxation (event-related synchronization, ERS) [30]. ERD and ERS could be used as relevant features for the task of motor imagery EEG classification. However those phenomena might happen in a different frequency band for some subjects, for instance, in 16-20 Hz, not in 8-12 Hz [16].

Recently, NMF was shown to be useful in determining discriminative basis vectors which well reflect meaningful spectral characteristics without the cross-validation in motor imagery EEG task [19]. Nonnegative tensor factorization was also applied to learn spectral features for continuous EEG classification [20]. However, when a data matrix becomes large, NMF undergoes high time complexity as well as extremely large space complexity, leading to 'out of memory' problem. In this paper we pay our attention to solve the out-of-memory problem in the case where NMF is applied to a large data matrix. We present memory-reduced methods where we use CUR decomposition [6]–[8] to downsize the data matrix before NMF is applied. Two methods, CUR-NMF and C-NMF, are illustrated and tested for EEG classification task using data sets in BCI competition.

Hyekyoung Lee and Seungjin Choi are with the Department of Computer Science, Pohang University of Science and Technology, Korea (email: {leehk,seungjin}@postech.ac.kr).

II. Background

A. NMF

Denote the data matrix by $\boldsymbol{X} = [\boldsymbol{x}_1 \cdots \boldsymbol{x}_l] = [X_{ij}] \in \mathbb{R}^{m \times l}$ which contains m-dimensional data vectors $\boldsymbol{x}_t \in \mathbb{R}^m$ for $t = 1, \ldots, l$. NMF assumes the nonnegative data matrix ($X_{ij} \geq 0$ for $i = 1, \ldots, m$ and $j = 1, \ldots, l$) and seeks a decomposition that is of the form

$$\boldsymbol{X} \approx \boldsymbol{A}\boldsymbol{S} = \widehat{\boldsymbol{X}}, \tag{1}$$

with matrices $\boldsymbol{A}$ and $\boldsymbol{S}$ restricted to have only nonnegative elements, where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ contains basis vectors in its columns and $\boldsymbol{S} \in \mathbb{R}^{n \times l}$ is the associated encoding variable matrix.

Various error measures for the factorization with nonnegativity constraints, can be considered [18]. In this paper we only consider Euclidean distance which is given by

$$\| \boldsymbol{X} - \boldsymbol{A}\boldsymbol{S} \|^2 = \sum_{i,j} (X_{ij} - [\boldsymbol{A}\boldsymbol{S}]_{ij})^2.$$

NMF involves the following optimization problem:

$$\underset{\boldsymbol{A},\boldsymbol{S}}{\arg\min} \| \boldsymbol{X} - \boldsymbol{A}\boldsymbol{S} \|^2 \tag{2}$$

$$\text{s.t. } A_{ij}, S_{ij} \geq 0 \ \forall i, j. \tag{3}$$

The multiplicative updating rules [18] for iteratively determining a local minimum of (2), are given by

$$S_{ij} \leftarrow S_{ij} \left(\frac{[\boldsymbol{A}^\top \boldsymbol{X}]_{ij}}{[\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{S}]_{ij}} \right), \tag{4}$$

$$A_{ij} \leftarrow A_{ij} \left(\frac{[\boldsymbol{X}\boldsymbol{S}^\top]_{ij}}{[\boldsymbol{A}\boldsymbol{S}\boldsymbol{S}^\top]_{ij}} \right). \tag{5}$$

B. CUR

CUR decomposition assumes that the original data matrix can be expressed by a smaller number of actual column and row vectors that are appropriately selected from the data matrix itself. The CUR decomposition is of the form:

$$\boldsymbol{X} \approx \widetilde{\boldsymbol{X}} = \boldsymbol{CUR}, \tag{6}$$

where $\boldsymbol{C} \in \mathbb{R}^{m \times c}$ consists of $c \ll l$ columns of $\boldsymbol{X}$, $\boldsymbol{R} \in \mathbb{R}^{r \times l}$ consists of $r \ll m$ rows of $\boldsymbol{X}$, and $\boldsymbol{U} \in \mathbb{R}^{c \times r}$ is an appropriately-defined low-dimensional encoding matrix [8]. It was shown in [8] that the decomposition (6) holds approximately, if the rows and columns are chosen uniformly and/or non-uniformly with probabilities that depend on Euclidean norms of rows and columns,

$$p_j = \sum_i X_{ij}^2 / \sum_{i,k} X_{ik}^2, \tag{7}$$

$$q_i = \sum_j X_{ij}^2 / \sum_{k,j} X_{kj}^2. \tag{8}$$

CUR decomposition requires $\mathcal{O}(l+m)$ space for $\boldsymbol{CUR}$, whereas $\boldsymbol{X}$ is stored in $\mathcal{O}(lm)$ space. If the data matrix $\boldsymbol{X}$ is large and sparse, but is well-approximated by a low-rank matrix, then $\boldsymbol{C}$ and $\boldsymbol{R}$ are sparse, whereas the matrices consisting of the top left and right singular vectors will not be sparse in general. The algorithm for CUR decomposition is described in Table I and more details can be found in [6]–[8].

TABLE I
CUR DECOMPOSITION

Input $\boldsymbol{X} \in \mathbb{R}^{m \times l}$, positive integers c and r, probability distributions $\{p_j\}_{j=1}^{l}$ and $\{q_i\}_{i=1}^{m}$.
Output $\boldsymbol{C} \in \mathbb{R}^{m \times c}, \boldsymbol{U} \in \mathbb{R}^{c \times r}, \boldsymbol{R} \in \mathbb{R}^{r \times l}$.

1. Select c columns in c i.i.d. trials according to $\{p_j\}_{j=1}^{l}$.
   - Set $\boldsymbol{C}_{:,t} = \boldsymbol{X}_{:,j_t} / \sqrt{c p_{j_t}}$, where $\boldsymbol{C}_{:,t}$ is the t-th column vector of $\boldsymbol{C}$ and the j_t-th column is chosen in the t-th independent trial.
2. Select r rows in r i.i.d. trials according to $\{q_i\}_{i=1}^{m}$.
   - Set $\boldsymbol{R}_{t,:} = \boldsymbol{X}_{i_t,:} / \sqrt{r q_{i_t}}$, where $\boldsymbol{R}_{t,:}$ is the t-th row vector of $\boldsymbol{R}$ and the i_t-th row is chosen in the t-th independent trial.
   - Set $\boldsymbol{W}_{t,:} = \boldsymbol{C}_{i_t,:} / \sqrt{r q_{i_t}}$,
3. Let the $c \times r$ matrix $\boldsymbol{U} = \boldsymbol{W}^{+}$.

III. PROPOSED METHOD

The high computational cost in the original NMF is in computations $\boldsymbol{A}^\top \boldsymbol{X}$ and $\boldsymbol{X}\boldsymbol{S}^\top$ in (4) and (5), respectively. Replacing $\boldsymbol{X}$ by $\boldsymbol{CUR}$ in NMF, reduces the computational cost down to $\mathcal{O}(lrn + cnr + mnc) = \mathcal{O}(l+m)$ (since $c, r, n \ll m, l$ and $n < c, r$)from $\mathcal{O}(lm)$. This is referred to as *CUR-NMF* where $\boldsymbol{CUR} \approx \boldsymbol{AS}$ which is summarized below.

$$S_{ij} \leftarrow S_{ij} \left(\frac{[\boldsymbol{A}^\top \boldsymbol{CUR}]_{ij}}{[\boldsymbol{A}^\top \boldsymbol{AS}]_{ij}} \right), \tag{9}$$

$$A_{ij} \leftarrow A_{ij} \left(\frac{[\boldsymbol{C}(\boldsymbol{U}(\boldsymbol{RS}^\top))]_{ij}}{[\boldsymbol{ASS}^\top]_{ij}} \right). \tag{10}$$

The matrix $\boldsymbol{C}$ in CUR decomposition consists of column vectors selected from the data matrix $\boldsymbol{X}$. We can use $\boldsymbol{C}$ only as an input data matrix instead of the original data matrix $\boldsymbol{C}$ in order to compute $\boldsymbol{A}$ and $\boldsymbol{S}$ using NMF. This is quite similar to data selection that was used in [19]. This is referred to as *C-NMF* which is summarized below.

$$S_{ij} \leftarrow S_{ij} \left(\frac{[\boldsymbol{A}^\top \boldsymbol{C}]_{ij}}{[\boldsymbol{A}^\top \boldsymbol{AS}]_{ij}} \right), \tag{11}$$

$$A_{ij} \leftarrow A_{ij} \left(\frac{[\boldsymbol{C}\boldsymbol{S}^\top]_{ij}}{[\boldsymbol{ASS}^\top]_{ij}} \right). \tag{12}$$

Discrete probabilities (7) and (8) used to select columns and rows were computed according to Euclidean norms in the original CUR decomposition. In this paper we also consider a sparseness measure instead of Euclidean norm in computing those discrete probabilities. We use the sparseness measure proposed by Hoyer [12]

$$\xi(\boldsymbol{x}) = \frac{\sqrt{m} - (\sum |x_i|) / \sqrt{\sum x_i^2}}{\sqrt{m} - 1}, \tag{13}$$

where x_i is the ith element of the m-dimensional vector $\boldsymbol{x}$. Discrete probabilities $\{p_j\}$ and $\{q_i\}$ are computed by

$$p_j = \xi(\boldsymbol{X}_{:,j}) / \sum_k \xi(\boldsymbol{X}_{:,k}), \tag{14}$$

$$q_i = \xi(\boldsymbol{X}_{i,:}) / \sum_k \xi(\boldsymbol{X}_{k,:}). \tag{15}$$

Note that this works for feature extraction, not for a low rank approximation of a matrix.

IV. NUMERICAL EXPERIMENTS

We investigate the performance of our CUR-NMF (9) and (10) and C-NMF (11) and (12), in the task of learning discriminative spectral features from EEG classification, compared to the original NMF (4) and (5). For our empirical study, we used two data sets: one is the dataset V in BCI competition III which was provided by the IDIAP Research Institute [13], and the other is the dataset IIIa in BCI competition III which was provided by the Laboratory of Brain-Computer Interfaces (BCI-Lab), Graz University of Technology [2].

Experiments for EEG classification involves the following procedures:

1) preprocessing (by wavelet transform or short-time Fourier transform)
2) CUR decomposition with discrete probabilities using
   - Euclidean distance defined by (7) and (8),
   - Sparseness defined by (14) and (15)
3) NMF-based feature extraction using
   - NMF: NMF with $\boldsymbol{X}$, (4) and (5)
   - CUR-NMF: NMF with $\boldsymbol{CUR}$, (9) and (10)
   - C-NMF: NMF with $\boldsymbol{C}$, (11) and (12)
4) classification (by time-dependent linear discriminant analysis [19], [22] or Viterbi algorithm [20]).

A. IDIAP dataset

1) Data description: The IDIAP dataset contains EEG data recorded from 3 normal subjects and involves three tasks, including the imagination of repetitive self-paced left/right hand movements and the generation of words beginning with the same random letter. The subject performed a given task for about 15 seconds and then switched randomly to another task at the operator's request. In contrast to the Graz dataset, EEG data is not split in trials, since the subjects are continuously performing any of the mental tasks (i.e., no trial structure).

We use the precomputed features which obtained by the power spectral density (PSD) in the band 8-30 Hz every 62.5 ms, (i.e., 16 times per second) over the last second of data with a frequency resolution of 2 Hz for the eight centro-parietal channels C_3, C_z, C_4, CP_1, CP_2, P_3, P_z, and P_4 after the raw EEG potentials were first spatially filtered by means of a surface Laplacian. As a result, an EEG sample is a 96-dimensional vector (eight channels times 12 frequency components).

2) Preprocessing: The data matrix $\boldsymbol{X}_{train} \in \mathbb{R}^{96\times 10528}$ is constructed by normalizing spectral components $P_i(f,t)$ (precomputed features), i.e.,

$$\mathcal{X}_{(i-1)*8+f,t} = \frac{P_i(f,t)}{\sum_f P_i(f,t)}, \tag{16}$$

for $f \in \{8, 10, \ldots, 28, 30\}$ Hz, $i = 1, 2, \ldots, 8$ (corresponding to 8 different channels, including C_3, C_z, C_4, CP_1, CP_2, P_3, P_z, and P_4), $t = 1, \ldots, 10528$ where 10528 is the number of data points in the training set (note that there is no trial structure in this dataset). In the same way, we make the test data matrix, $\boldsymbol{X}_{test} \in \mathbb{R}^{96\times 3504}$.

3) CUR decomposition: As mentioned above, we decompose $\boldsymbol{X}_{train}$ to $\boldsymbol{CUR}$ using the probability obtained by Euclidean norm and sparseness according to CUR algorithm in Table I. We can compute the approximation error in (6) by

$$\| \boldsymbol{X}_{train} - \boldsymbol{CUR} \|^2, \tag{17}$$

where $\| \cdot \|$ means spectral norm. For each probability, the approximation errors are shown in Fig. 1 and 2. Euclidean norm is better decomposing the matrix than sparseness. When the number of selected rows, r, increases, the approximation error decreases. Whereas, the number of selected columns, c, is little influence on it. This means that same tasks are repetitively observed in the training data and CUR decomposition can find the data points which are not duplicated information. Thus, selected 1000 columns among 10528 columns is enough to represent whole data points.

4) Feature extraction and classification: After computing basis matrix $\boldsymbol{A}$ by NMF, C-NMF, or CUR-NMF, we can obtain the feature matrix $\boldsymbol{S}$ by $[\boldsymbol{A}]^\dagger \boldsymbol{X}$ where $\dagger$ represents the pseudo-inverse.

For the on-line classification for IDIAP data which consist of uncued EEG signals, we use the Viterbi algorithm [11] that is a dynamic programming algorithm for finding a most

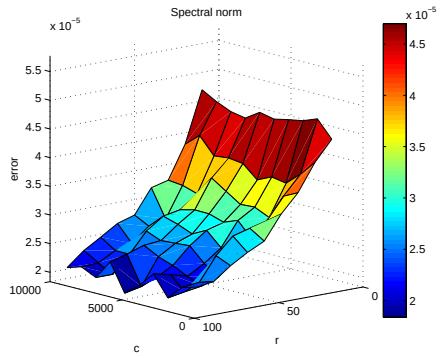

Fig. 1. CUR approximation error for the probability obtained by Euclidean norm. the x-axis means 'c', the number of selected columns and the y-axis means 'r', the number of selected rows.

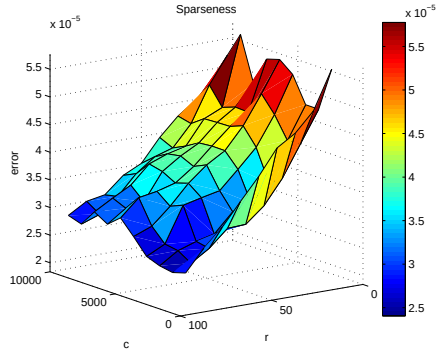

Fig. 2. CUR approximation error for the probability obtained by sparseness

probable sequence of hidden states that explains a sequence of observations. The graphical model involving the Viterbi algorithm is shown in Fig. 3, where hidden states follow the first-order Markov chain and an observed variable at time t depends on only a hidden state at time t.

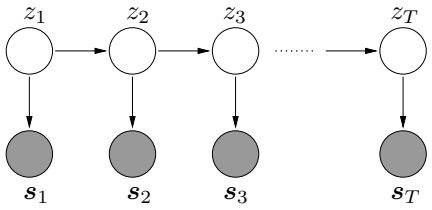

Fig. 3. The graphical model involving the Viterbi algorithm is shown.

The dependency between hidden states at $t-1$ and t is defined by a transition probability $P(z_t|z_{t-1})$ and the dependency between observation at t and hidden state at t is defined by an emission probability $P(\boldsymbol{s}_t|z_t)$. In our case, hidden states correspond to class labels, i.e., $z_t \in \{1, 2, 3\}$ which are related to imagery left/right hand movements and the imagination of word generation. All probabilities can be estimated in the phase of training easily. We can infer the hidden label, class label, given the test sequence using Viterbi algorithm (more details in [20]).

Fig. 4, 5, 6 and 7 show the classification error varying the number of selected columns, c and the number of selected

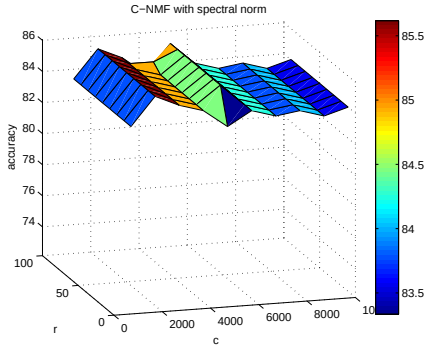

Fig. 4. Classification accuracy of NMF of $\boldsymbol{C}$ selected by the probability using spectral norm

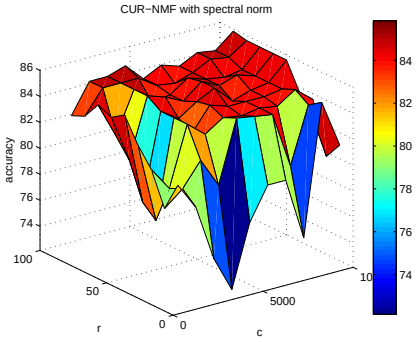

Fig. 6. Classification accuracy of CUR-NMF selected by the probability using spectral norm

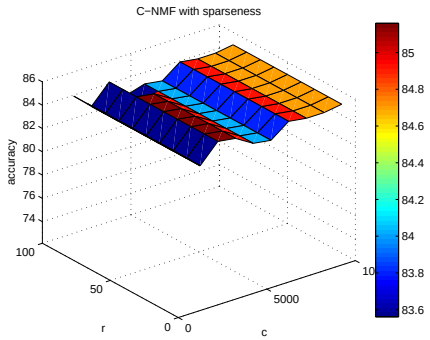

Fig. 5. Classification accuracy of NMF of $\boldsymbol{C}$ selected by the probability using sparseness

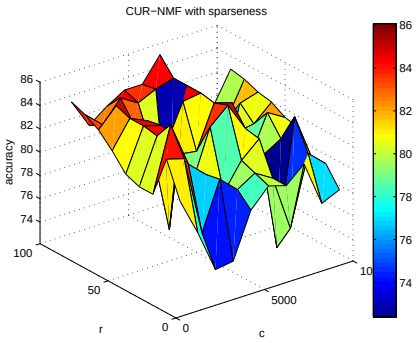

Fig. 7. Classification accuracy of CUR-NMF selected by the probability using sparseness

rows, r. The number of basis, n is 4 and we choose subject 1's data set. For the case of C-NMF in Fig. 4 and 5, it is no wonder varying r is no influence on the performance because we only use the matrix, $\boldsymbol{C} \in \mathbb{R}^{m \times c}$. Varying c is also little influence on the results because $\boldsymbol{C}$ is well-selected matrix, which is already mentioned in CUR approximation error, without reference to the probability measure, Euclidean norm or sparseness. (Actually, spectral norm has slightly better performance than sparseness.) From this, we can conclude that it is enough to analyze 1000 data points, about 10 $\%$ of whole data points.

The classification result of CUR-NMF with Euclidean norm (Fig. 6) is almost consistent when r is more than 30 and c is more than 3000. However, in the case of sparseness (Fig. 7), the performance is random for varying c. The reason can find in Fig. 8, q_i obtained by sparseness is somewhat uniform, (plotted the red dotted line on right figure, where the x-axis indicates the probability q_i and the y-axis indicates the index of column). However, real data are easily distinguished informative data. Thus, if wrong data are selected by good bad luck, the performance will become bad. The bottom figure in 8 is p_j, the blue solid line of Euclidean norm is nearly similar to the red dotted line for sparseness. Increasing c is increasing the performance to some extent.

Total classification accuracy of IDIAP data set for all subject show in II. The first line is the results of NMF. The subject1's data set has the best performance. 'p' and 's' means whether the probability is obtained by Euclidean norm or sparseness. All things considered, the results from sparseness is not inspire confidence in consistent performance. When CUR technique is used, the performance of subject2 and subject3 is better than the one of subject1. This results show that data selection with probability is useful.

TABLE II
CLASSIFICATION ACCURACY OF IDIAP DATA

	sub1	sub2	sub3	avg
NMF	84.93	70.51	55.28	70.24
C-NMF-p	86.30	73.27	58.49	72.69
C-NMF-s	87.21	75.58	60.09	74.29
CUR-NMF-p	86.07	76.73	59.17	73.99
CUR-NMF-s	85.16	72.81	55.96	71.31
BCI comp. winner	79.60	70.31	56.02	68.65

B. Graz dataset

1) Data description: The Graz dataset involves left hand, right hand, foot, tongue imagery movements and consists of 120 labeled trials for training and 120 unlabeled trials for test. Each trial has a duration of 7 seconds, where a visual

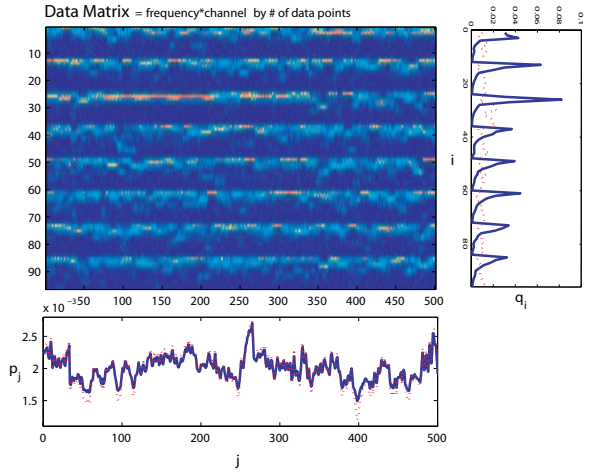

Fig. 8. Plot the probability of columns and rows of given data matrix. Blue solid line and red dotted line are the probability using spectral norm and sparseness, respectively.

cue (arrow) is presented pointing to the left, right, up or down after 3-second preparation period and imagination task of a left hand, right hand, tongue or foot movement is carried out for 4 seconds. It contains EEG acquired from 60 channels (with sampling frequency 250 Hz). Requirements for result comparison is to provide a continuous classification accuracy for each time point of trial during imagination session. This dataset is also recorded from 3 subjects, however, we employ only one dataset, subject 'l1b.'

2) Preprocessing: We downsample to 125 Hz and obtain the time-frequency representation of the EEG data, by filtering it with complex Morlet wavelets [19]. We make labeled and unlabeled data matrix, $\boldsymbol{X}_{train} \in \mathbb{R}^{1620\times 60000}$ and $\boldsymbol{X}_{test} \in \mathbb{R}^{1620\times 60000}$, respectively. The number of rows is 60 channels $\times$ 27 frequency components in $\{4, 5, 6, \ldots, 30\}$ Hz. The number of columns is 500 data points in a trial $\times$ 120 number of training data/test data.

3) CUR decomposition: The same methods are applied to Graz dataset, too. One difference is that this dataset is too large to analyze NMF with 2 GByte RAM in MATLAB due to 'out of memory' problem. Thus, we first estimate the probability p_j and q_i, simultaneous with the transformation step to time-frequency representation, then not saving the transformed data in RAM. We make the matrix $\boldsymbol{C}$, $\boldsymbol{U}$ and $\boldsymbol{R}$ with $c = 6000$ and $r = 40$. These matrix saved in smaller space than the transformed data.

4) Feature extraction and classification: We apply C-NMF method because it has shown good performance for IDIAP dataset. For the single-trial online classification for Graz data (with trial structure), we use a Gaussian probabilistic model-based classifier [22] where Gaussian class-conditional probabilities for a single point in time t are integrated temporally by taking the expectation of the class probabilities with respect to the discriminative power at each point in time. and apply the one-against all approach because this method is possible to 2-class classification. The performance measure is kappa value that is evaluation measure for competition. Our result is 0.7532 for the subject 'l1b'. Although this result is lower than best result 0.8000 in competition, we show the possibility doing NMF for large data set.

V. Conclusions

We have presented methods of learning discriminative spectral features from large data matrix involving EEG power spectrum. Incorporating CUR decomposition into NMF led us to downsize the large data matrix such that NMF could be applied to compute discriminative spectral features. Two methods, CUR-NMF and C-NMF, were presented, reducing the space overhead down to $\mathcal{O}(l+m)$ from $\mathcal{O}(lm)$. We have also proposed the sparseness-based discrete probabilities that were used in selecting columns and rows from the data matrix. Experiments on two EEG data sets used in BCI competition have confirmed that the data selection by CUR decomposition was useful in NMF-based spectral feature extraction for EEG classifications.

Acknowledgments: This work was supported by KOSEF Basic Research Program (grant R01-2006-000-11142-0) and National Core Research Center for Systems Bio-Dynamics.

References

[1] J. H. Ahn, S. Choi, and J. H. Oh, "A multiplicative up-propagation algorithhm," in *Proceedings of the International Conference on Machine Learning*, Banff, Canada, 2004, pp. 17–24.

[2] B. Blankertz, K. R. Müller, D. J. Krusierski, G. Schalk, J. R. Wolpaw, A. Schlögl, G. Pfurtscheller, and N. Birbaumer, "The BCI competition III: Validating alternative approaches to actual BCI problems," *IEEE Trans. Neural Systems and Rehabilitation Engineering*, vol. 14, pp. 153–159, 2006.

[3] J. P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proceedings of the National Academy of Sciences, USA*, vol. 101, no. 12, pp. 4164–4169, 2004.

[4] Y. C. Cho and S. Choi, "Nonnegative features of spectro-temporal sounds for classfication," *Pattern Recognition Letters*, vol. 26, no. 9, pp. 1327–1336, 2005.

[5] A. Cichocki, R. Zdunek, S. Choi, R. J. Plemmons, and S. Amari, "Non-negative tensor factorization using alpha and beta divergences," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Honolulu, Hawaii, 2007.

[6] P. Drineas, R. Kannan, and M. W. Mahoney, "Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication," *SIAM Journal on Computing*, vol. 36, no. 1, pp. 132–157, 2006.

[7] ——, "Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix," *SIAM Journal on Computing*, vol. 36, no. 1, pp. 158–183, 2006.

[8] ——, "Fast Monte Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition," *SIAM Journal on Computing*, vol. 36, no. 1, pp. 184–206, 2006.

[9] T. Ebrahimi, J. F. Vesin, and G. Garcia, "Brain-computer interface in multimedia communication," *IEEE Signal Processing Magazine*, vol. 20, no. 1, pp. 14–24, Jan. 2003.

[10] J. Eggert, H. Wersing, and E. Körner, "Transformation-invariant representation and NMF," in *Proceedings of the International Joint Conference on Neural Networks*, 2004.

[11] G. D. Forney, "The Viterbi algorithm," *Proceedings of of the IEEE*, vol. 61, pp. 268–278, 1973.

[12] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[13] J. del R. Millán, "On the need for on-line learning in brain-computer interfaces," in *Proceedings of the International Joint Conference on Neural Networks*, Budapest, Hungary, 2004.

[14] M. Kim and S. Choi, "Monaural music source separation: Nonnegativity, sparseness, and shift-invariance," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation*. Charleston, South Carolina: Springer, 2006, pp. 617–624.

[15] Y. D. Kim and S. Choi, "Nonnegative Tucker decomposition," in *Proceedings of the IEEE CVPR-2007 Workshop on Component Analysis Methods*, Minneapolis, Minnesota, 2007.

[16] T. N. Lal, M. Schroder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Schölkopf, "Support vector channel selection in BCI," Max Planck Institute for Biological Cybernetics, Tech. Rep. 120, 2003.

[17] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[18] ——, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, vol. 13. MIT Press, 2001.

[19] H. Lee, A. Cichocki, and S. Choi, "Nonnegative matrix factorization for motor imagery EEG classification," in *Proceedings of the International Conference on Artificial Neural Networks*. Athens, Greece: Springer, 2006.

[20] H. Lee, Y. D. Kim, A. Cichocki, and S. Choi, "Nonnegative tensor factorization for continuous EEG classification," *International Journal of Neural Systems*, vol. 17, no. 4, pp. 305–317, 2007.

[21] J. S. Lee, D. D. Lee, S. Choi, and D. S. Lee, "Application of non-negative matrix factorization to dynamic positron emission tomography," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation*, San Diego, California, 2001, pp. 629–632.

[22] S. Lemm, C. Schäfer, and G. Curio, "BCI competition 2003-data set III: Probabilistic modeling of sensorimotor μ rhythms for classification of imaginary hand movements," *IEEE Trans. Biomedical Engineering*, vol. 51, no. 6, 2004.

[23] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. S. Cheng, "Learning spatially localized parts-based representation," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii, 2001, pp. 207–212.

[24] P. Paatero and U. Tapper, "Least squares formulation of robust non-negative factor analysis," *Chemometrics Intelligent Laboratory Systems*, vol. 37, pp. 23–35, 1997.

[25] A. Pascual-Montano, J. M. Carazo, K. K. D. Lehmann, and R. D. Pascual-Margui, "Nonsmooth nonnegtive matrix facotorization (nsNMF)," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 403–415, 2006.

[26] F. Shahnaz, M. Berry, P. Pauca, and R. Plemmons, "Document clustering using nonnegative matrix factorization," *Information Processing and Management*, vol. 42, pp. 373–386, 2006.

[27] A. Shashua and T. Hazan, "Non-negative tensor factorization with applications to statistics and computer vision," in *Proceedings of the International Conference on Machine Learning*, Bonn, Germany, 2005.

[28] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, 2003, pp. 177–180.

[29] M. Welling and M. Weber, "Positive tensor factorization," *Pattern Recognition Letters*, vol. 22, pp. 1255–1261, 2001.

[30] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, pp. 767–791, 2002.

[31] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval*, Toronto, Canada, 2003.