EUSIPCO 2013 1569734783

CANONICAL POLYADIC DECOMPOSITION FOR UNSUPERVISED LINEAR FEATURE EXTRACTION FROM PROTEIN PROFILES

A. Jukić, I. Kopriva

Division of Laser and Atomic R&D
Ruđer Bošković Institute
Zagreb, Croatia

A. Cichocki

Laboratory for Advanced Brain Signal Processing
RIKEN Brain Science Institute
Saitama, Japan

ABSTRACT

We propose a method for unsupervised linear feature extraction through tensor decomposition. The linear feature extraction can be formulated as a canonical polyadic decomposition (CPD) of a third-order tensor when transformation matrix is constrained to be equal to the Khatri-Rao product of two matrices. Therefore, standard algorithms for computing CPD decomposition can be used for feature extraction. The proposed method is validated on publicly available low-resolution mass spectra of cancerous and non-cancerous samples. Obtained results indicate that this approach could be of practical importance in analysis of protein expression profiles.

Index Terms— Feature extraction, tensor decomposition, cancer prediction.

1. INTRODUCTION

One of the most important problems in machine learning and data analysis is feature extraction or dimension reduction. This is essential problem in various areas, such as text mining, combinatorial chemistry, and computational biology [1]. A well chosen dimension reduction preprocessing can have a significant impact on the computational cost and interpretation of the originally high-dimensional samples. In classification problems feature extraction is crucial, with often greater impact on the overall performance than the type of the classifier used [2]. Suitable feature extraction prior to learning can improve generalization of the trained model, by reducing overfitting, even for regularized techniques such as support vector machines (SVM).

Here we will consider problem of linear feature extraction (LFE), although feature extraction can be performed through a nonlinear transformation of the data. The goal of LFE is to find a linear transformation of the original samples that results in a low-dimensional representation, while retaining all information for predicting class labels. This can be done in supervised or unsupervised manner, the difference being in utilization of labels provided in the training set. Classical approach for unsupervised LFE is principal component analysis (PCA), while linear discriminant analysis (LDA) is used for supervised LFE. In this paper we propose a method for unsupervised linear feature extraction, that can be performed by decomposing a three-way tensor constructed using samples in the training set.

Modern applications often result in multi-way data that is naturally represented in tensor form [3]. This type of data is commonly encountered in psychometry, neuroscience, chemometrics, computer vision and analysis of social networks. Consequently, numerous algorithms for tensors were developed, with applications in signal processing, data analysis and machine learning [3, 4]. Also, several algorithms for feature extraction for tensor objects were proposed recently, with applications in EEG analysis and image classification [5, 6, 7].

Contribution of this paper is application of canonical polyadic decomposition for linear feature extraction. More specifically, we propose to perform unsupervised LFE with structural constraint on the transformation matrix. Consequently, LFE is formulated as tensor decomposition of a three-way tensor. The proposed approach is tested on binary classification of protein expression profiles acquired by the low-resolution surface-enhanced laser desorption ionization time-of-flight (SELDI-TOF) mass spectrometry, and compared to our previous approach that used a more general Tucker3 model [8]. Obtained results are also compared with recently developed methods for feature extraction in bioinformatics [9, 10]. Experiments show that the proposed method could provide an interesting alternative to existing LFE approaches for protein expression levels, but also possibly for other data with large number of variables such as gene expression profiles.

2. METHODS

In the following scalars will be denoted by italic letters (e.g., x), vectors by bold lowercase letters (e.g., $\mathbf{x}$), matrices by bold capital letters (e.g., $\mathbf{X}$) and tensors by bold underlined capital letters (e.g., $\underline{\mathbf{X}}$). We will mainly consider three-way tensors, with notation as in [3, 4].

2.1. Preliminaries

Tensor is a multi-way generalization of matrix and vector, and order of tensor is equal to the number of its indices, with each index defining a way or mode of tensor. For example, $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is an order-N or N-way tensor, with elements $x_{i_1 i_2 \ldots i_N}$. A fiber of a tensor is a vector obtained by fixing all indices except one, i.e., generalization of matrix column and row. Mode-n fibers are obtained by fixing all indices but i_n. Mode-n unfolding of a tensor $\underline{\mathbf{X}}$ is obtained by arranging all mode-n fibers as columns of a matrix, and is denoted as $\mathbf{X}_{(n)}$. Ordering of columns is not important as long as it is consistent in all computations. Matricization is transformation of a vector $\mathbf{x}^{(k)} \in \mathbb{R}^{I_1 \cdot I_2}$ to a matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{I_1 \times I_2}$, denoted as $\mathbf{X}^{(k)} = mat\left(\mathbf{x}^{(k)}\right)$. This is inverse to the vec operation, that turns matrix into a vector by stacking the columns into one long vector. The mat operation divides vector $\mathbf{x}^{(k)}$ into I_2 parts, and stacks them as columns in matrix $\mathbf{X}^{(k)}$:

$$\mathbf{X}^{(k)}(:, i_2) = \mathbf{x}^{(k)}\left((i_2 - 1)I_1 + 1 : i_2 I_1\right), \tag{1}$$

for $i_2 \in \{1, \ldots, I_2\}$. Mode-$n$ product of tensor $\underline{\mathbf{X}}$ and matrix $\mathbf{A}$ is new tensor denoted as $\underline{\mathbf{X}} \times_n \mathbf{A}$. It is defined as

$$\underline{\mathbf{Y}} = \underline{\mathbf{X}} \times_n \mathbf{A} \iff \mathbf{Y}_{(n)} = \mathbf{A}\mathbf{X}_{(n)}. \tag{2}$$

The mode-n multiplication is commutative when applied in distinct modes, while repeated multiplication in the same mode can be expressed as

$$\underline{\mathbf{X}} \times_n \mathbf{A} \times_n \mathbf{B} = \underline{\mathbf{X}} \times_n (\mathbf{B}\mathbf{A}). \tag{3}$$

Three-way rank-1 tensor can be written as the outer product of three vectors $\mathbf{a} \in \mathbb{R}^{I_1}$, $\mathbf{b} \in \mathbb{R}^{I_2}$, $\mathbf{c} \in \mathbb{R}^{I_3}$, as

$$\underline{\mathbf{X}} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \in \mathbb{R}^{I_1 \times I_2 \times I_3}, \tag{4}$$

where $\circ$ denotes outer product. Tensor rank of a general tensor $\underline{\mathbf{X}}$ is defined as follows

$$r(\underline{\mathbf{X}}) = \min\left\{ R : \underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \right\}. \tag{5}$$

As opposed to matrix rank, tensor rank is hard to compute [4]. Norm of tensor is defined as $\|\underline{\mathbf{X}}\| = \sqrt{\left(\sum_{i,j,k} x_{ijk}^2\right)}$, and it is compatible with the Frobenius norm in matrix case.

One of the most common tensor decomposition is the canonical polyadic decomposition (CPD) [11], where the original tensor is decomposed into a sum of rank-1 tensors. For tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ the CPD model can be written as

$$\underline{\mathbf{X}} = \sum_{r=1}^{r(\underline{\mathbf{X}})} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r, \tag{6}$$

with $\mathbf{a}_r \in \mathbb{R}^{I_1}$, $\mathbf{b}_r \in \mathbb{R}^{I_2}$, $\mathbf{c}_r \in \mathbb{R}^{I_3}$ for $r \in \{1, \ldots, R\}$. The CP decomposition (6) can be expressed as

$$\underline{\mathbf{X}} = \underline{\mathbf{I}} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}, \tag{7}$$

with factor matrices $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_R]$, $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_R]$, and $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_R]$, and diagonal core tensor $\underline{\mathbf{I}} \in \mathbb{R}^{R \times R \times R}$ with ones on the diagonal. When the core in (7) is not constrained to be cubic and diagonal we obtain Tucker3 model. The model (7) can also be written as

$$\mathbf{X}_k = \mathbf{A}\,\mathrm{diag}\left(\mathbf{C}(k, :)\right)\mathbf{B}^T, \tag{8}$$

where $\mathbf{X}_k = \underline{\mathbf{X}}(:, :, k)$ is the k-th frontal slice of the tensor $\underline{\mathbf{X}}$, and $\mathrm{diag}\left(\mathbf{C}(k, :)\right)$ is a diagonal matrix with the k-th row of matrix $\mathbf{C}$ on its diagonal. Expression (8) can be written as

$$vec(\mathbf{X}_k) = (\mathbf{B} \odot \mathbf{A})(\mathbf{C}(k, :)^T), \tag{9}$$

where $\odot$ is Khatri-Rao product, defined as $\mathbf{B} \odot \mathbf{A} = [\mathbf{b}_1 \otimes \mathbf{a}_1, \ldots, \mathbf{b}_R \otimes \mathbf{a}_R]$, i.e., column-wise Kronecker product. An important property of the CPD is that it is essentially unique (up to permutation and scaling of factors) under mild conditions [12]. Sufficient condition for uniqueness is

$$k(\mathbf{A}) + k(\mathbf{B}) + k(\mathbf{C}) \geq 2r(\underline{\mathbf{X}}) + 2, \tag{10}$$

where $k(\mathbf{A})$ denotes Kruskal rank of a matrix, that is equal to the maximal number k such that every k columns of $\mathbf{A}$ are linearly independent.

In practical applications, it is often convenient to approximate given tensor with model of lower rank, $R < r(\underline{\mathbf{X}})$. This low rank approximation is usually found through minimization of the residual

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \|\underline{\mathbf{X}} - \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r\|, \tag{11}$$

that boils down to least squares problem in case of previously defined norm, although other norms can be used.

2.2. CPD for linear feature extraction

Let $S = \left\{(\mathbf{x}^{(k)}, c^{(k)}), k \in \{1, \ldots, K\}\right\}$ be a given set of K training samples $\mathbf{x}^{(k)} \in \mathbb{R}^I$ paired with their labels $c^{(k)} \in \{1 \ldots C\}$, where C is number of classes. Commonly, the LFE is formulated as optimization of some criterion that measures *goodness* of the extracted features. The optimization procedure is usually performed over the set of column-orthogonal matrices $\mathbf{W} \in \mathbb{R}^{I \times J}$, $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, with $J < I$. The extracted features are obtained by projecting samples onto column-space of the transformation matrix as

$$\mathbf{f}^{(k)} = \mathbf{W}^T\mathbf{x}^{(k)}. \tag{12}$$

Then optimal linear transformation (in some sense) can be found by solving

$$\mathbf{W}^* = \underset{\mathbf{W}^T\mathbf{W}=\mathbf{I}}{\arg\max}\, D\left(\mathbf{W}|S\right), \tag{13}$$

where D is the selected criterion calculated over the training set S. In supervised approaches, D is a function of both features and class labels, while in unsupervised methods D depends only on the extracted features. In LDA, D is calculated

as the ratio of between class variance and within class variance, while in PCA D quantifies the proportion of the variance preserved in the features. Recently, several methods for LFE based on maximization of information-theoretic criteria were presented [13, 14], achieving high performance but at the cost of very high computational complexity.

The LFE can be formulated in similar way as

$$\mathbf{x}^{(k)} \approx \mathbf{W}\mathbf{f}^{(k)}, \tag{14}$$

where columns of $\mathbf{W}$ can be seen as a dictionary used to approximate the input samples, e.g., in least squares sense. Also, column-orthogonality constraint for the matrix $\mathbf{W}$ can be relaxed, requiring $\mathbf{W}$ only to have a full (column) rank. The dictionary $\mathbf{W}$ is learned from the given training data samples, in supervised or unsupervised manner [15].

In several applications (e.g., high resolution mass spectrometry) the input samples have a very large number of entries, meaning that their dimension I is huge. In this situation LFE (12) can become computationally difficult, due to size of the matrix $\mathbf{W}$. However, this problem can be alleviated by introducing structure into the transformation matrix. Here we propose to constrain $\mathbf{W}$ to be equal to Khatri-Rao product of two smaller matrices, i.e.,

$$\mathbf{W} = \mathbf{V} \odot \mathbf{U}, \tag{15}$$

with $\mathbf{U} \in \mathbb{R}^{I_1 \times J}$, $\mathbf{V} \in \mathbb{R}^{I_2 \times J}$, $I = I_1 \cdot I_2$, and $J \ll I$. This assumption enables formulation of feature extraction as a CPD of a three-way tensor. While this enables use of the CPD, the features are now constrained to lie in a subspace spanned by matrix in form $\mathbf{V} \odot \mathbf{U}$. Taking (15) into account, relation between the extracted features $\mathbf{f}^{(k)}$ and the input sample $\mathbf{x}^{(k)}$ can be written as

$$\mathbf{x}^{(k)} \approx (\mathbf{V} \odot \mathbf{U})\, \mathbf{f}^{(k)}. \tag{16}$$

If we use (9) and transform sample $\mathbf{x}^{(k)} \in \mathbb{R}^I$ to matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{I_1 \times I_2}$, as $\mathbf{X}^{(k)} = mat\left(\mathbf{x}^{(k)}\right)$, then (16) can be written as

$$\mathbf{X}^{(k)} \approx \mathbf{U}\, \mathrm{diag}\left(\mathbf{f}^{(k)}\right) \mathbf{V}^T. \tag{17}$$

Using (7) and (8), the joint diagonalization (17) can be expressed as a CPD of a three-way tensor

$$\underline{\mathbf{X}} \approx \underline{\mathbf{I}} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{F}, \tag{18}$$

where $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times K}$ is composed of transformed samples $\mathbf{X}^{(k)}$ as frontal slices, i.e., $\mathbf{X}_k = \mathbf{X}^{(k)}$, $k \in \{1, \ldots, K\}$. In (18), the mode-3 factor matrix $\mathbf{F} \in \mathbb{R}^{K \times J}$ contains features for each of the K samples, with each row of $\mathbf{F}$ being equal to a vector of features, $\mathbf{F}(k,:) = \mathbf{f}^{(k)^T}$.

Matrices $\mathbf{U}$ and $\mathbf{V}$ are learned by CPD of a three-way tensor formed from the training set. However, note that factor matrices in decomposition can be determined only up to permutation and scaling, as also stated in the uniqueness condition (10). Therefore, we scale matrices $\mathbf{U}$ and $\mathbf{V}$ to have unit ℓ_2 norm of columns, so the features in $\mathbf{F}$ carry all the variance. When faced with an unseen test sample $\mathbf{x}_{te} \in \mathbb{R}^I$, features $\mathbf{f}_{te} \in \mathbb{R}^J$ are obtained by solving a linear system, and expressed as

$$\mathbf{f}_{te} = (\mathbf{V} \odot \mathbf{U})^{\dagger}\, \mathbf{x}_{te}, \tag{19}$$

where $\dagger$ denotes Moore-Penrose psudoinverse.

2.3. Some practical issues

One of the practical problems is selection of the algorithm for canonical polyadic decomposition. The most common algorithm for solving (11) is based on alternating least squares (ALS), called CP-ALS. In each step, all factor matrices except one are kept fixed and optimization is performed with respect to a single factor matrix, reducing a nonlinear least squares problem (11) to linear least squares (for more details see [16]). Also, several advanced methods that minimize (11) were proposed [16]. However, CP-ALS remains a workhorse algorithm in a wide range of applications and we used it in our experiments because of its speed and simplicity.

Another practical issue is the number of selected features R, that is equal to rank of the model (18). Several methods exist for estimation of number of components, such as CORCONDIA [17] and DIFFIT [18]. However, they are aimed for situations where ultimate goal is low-rank approximation in terms of fit of the model. In this paper we deal with classification problem, so optimal number of features is selected by finding R that gives best classification performance estimated through cross-validation. In our previous approach [8] we used Tucker3, where core tensor is not constrained to be cubic and diagonal. In that case, number of features is determined as a product of two parameters (dimensions of the core tensor in modes 1 and 2). Important practical advantage when using CPD model is that only one parameter (R) determines number of the extracted features. Also, in context of feature extraction uniqueness of CPD model is an important theoretical advantage over Tucker3.

3. EXPERIMENTS

The proposed method for LFE is validated on prediction of prostate and ovarian cancer from protein expression profiles. We used CP-ALS algorithm implemented in [19]. Obtained features were classified using nonlinear support vector machine with Gaussian kernel (rbfSVM), implemented in [20], with parameters C and γ tuned through cross-validation (CV). Overall performance was estimated by two-fold CV, that was repeated on 20 random partitions of the dataset.

3.1. Description of data

We used samples obtained by analysis of proteins in serum of different patients by low-dimensional mass spectroscopy

[21]. Each sample in the dataset is labeled as control (corresponding to a healthy individual) or disease (patient with cancer confirmed through biopsy). Prostate cancer dataset consists of 69 disease samples and 63 control samples, while ovarian cancer dataset consists of 100 disease and 100 control samples. In both cases each sample is represented by intensity levels at $I_0 = 15154$ m/z ratios. Original samples were preprocessed by hand, and the baseline was subtracted [21]. We discarded last 25 elements (corresponding to the highest m/z ratios) in each sample, yielding vector samples $\mathbf{x}^{(k)} \in \mathbb{R}^I$ with $I = 15129$. Each sample is transformed to matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{I_1 \times I_2}$ with $I_1 = I_2 = 123$[1].

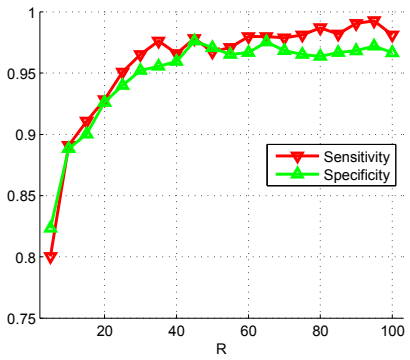
3.2. Results

We performed evaluation of the proposed approach for number of extracted features R in range $5 - 100$, with step size 5. Estimated performance on both datasets in terms of sensitivity and specificity is presented in Figure 1. The best result in classification of prostate cancer samples was obtained using $R = 95$ features, yielding sensitivity of 99.27% and specificity 97.20%. The best result in classification of ovarian cancer samples was obtained using $R = 100$ features, yielding sensitivity of 92.95% and specificity 92.00%. Table 1 contains comparison with results reported previously in the literature. Note that methods presented in [9, 8, 10] were evaluated on the same datasets as the proposed method, using the same validation procedure (two-fold CV). It can be seen that the proposed approach achieves competitive performance. When compared to [8], for the same number of features LFE based on CPD achieves better results.

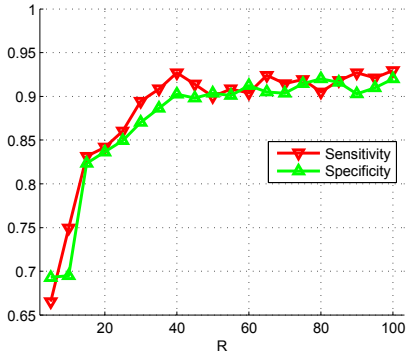
4. CONCLUSIONS

Feature extraction is of critical importance in machine learning, especially in scenarios with small number of samples with large number of variables that are typical for medical applications. Here we propose unsupervised LFE method based on canonical polyadic decomposition of a three-way tensor. The method is validated on publicly available data, with results indicating practical importance for analysis of protein expression profiles. However, we conjecture that the proposed approach would be useful in analysis gene expression levels and other data with large number of variables. An interesting direction of future research would be to develop a supervised method for CPD, that uses labels provided in the training set. Linear transformation obtained in this way would provide more discriminative features, possibly resulting in better classification performance.

[1]The number of discarded elements is negligible with respect to the total number of elements in a sample. However, in this way we could fold each sample to a square matrix.



(a) Prostate cancer



(b) Ovarian cancer

Fig. 1. Classification performance vs. number of features.

Table 1. Reported results for cancer prediction

Prostate cancer	SE [%]	SP [%]
Petricoin *et al.* [22]	94.7	75.9
Xu *et al.* [23]	97.1	96.8
Henneges *et al.* [9]	86	67.8
Kopriva and Filipović [10]	97.6	99
Kopriva *et al.* [8], 100 features	98.2	95.6
Proposed method, 95 features	99.3	97.2
Ovarian cancer	SE [%]	SP [%]
Petricoin *et al.* [24]	100	95
Li *et al.* [25]	98	95
Henneges *et al.* [9]	81.4	71.7
Kopriva and Filipović [10]	96.2	93.6
Kopriva *et al.* [8], 100 features	91.1	87.7
Proposed method, 100 features	93	92

5. ACKNOWLEDGEMENTS

The work of A. Jukić and I. Kopriva has been supported through grant 098-0982903-2558 funded by the Ministry of Science, Education and Sports, Republic of Croatia.

6. REFERENCES

[1] I. Guyon and A. Elisseeff, "An introduction to variable selection and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[2] T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, New York: Springer-Verlag, 2001.

[3] Andrzej A. Cichocki, A. H. Phan, and R. Zdunek, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*, Wiley, Chichester, 2009.

[4] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[5] F. Nie, S. Xiang, Y. Song, and C. Zhang, "Extracting the optimal dimensionality for local tensor discriminant analysis," *Pattern Recognition*, vol. 42, no. 1, pp. 105–114, Jan. 2009.

[6] H. A. Phan and A. Cichocki, "Tensor decompositions for feature extraction and classification of high dimensional datasets," *Nonlinear Theory and Its Applications, IEICE*, vol. 1, no. 1, pp. 37–68, 2010.

[7] F. Cong, A. H. Phan, Q. Zhao, T. Huttunen-Scott, J. Kaartinen, T. Ristaniemi, H. Lyytinen, and A. Cichocki, "Benefits of multi-domain feature of mismatch negativity extracted by non-negative tensor factorization from eeg collected by low-density array," *Int. J. Neural Syst.*, vol. 22, no. 6, 2012.

[8] I. Kopriva, A. Jukić, and A. Cichocki, "Feature extraction for cancer prediction by tensor decomposition of 1d protein expression levels," in *Proceedings of the IASTED Conference on Computational Bioscience CompBio 2011*, July 2011.

[9] Carsten Henneges, Pavel Laskov, Endang Darmawan, Jürgen Backhaus, Bernd Kammerer, and Andreas Zell, "A factorization method for the classification of infrared spectra," *BMC Bioinformatics*, vol. 11, no. 1, pp. 561, 2010.

[10] Ivica Kopriva and Marko Filipović, "A mixture model with a reference-based automatic selection of components for disease classification from protein and/or gene expression levels," *BMC Bioinformatics*, vol. 12, pp. 496, 2011.

[11] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *Journal of Mathematics and Physics*, , no. 7, pp. 164–189, 1927.

[12] J Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra and Its Applications*, vol. 18, pp. 95–138, 1977.

[13] K. Torkkola, "Feature extraction by non-parametric mutual information maximization," *Journal of Machine Learning Research*, vol. 3, pp. 1415–1438, 2003.

[14] J. M. Leiva-Murillo and A. Artès-Rodrìgues, "Maximization of mutual information for supervised linear feature extraction," *IEEE Transactions on Neural Networks*, vol. 18, no. 5, pp. 1433–1441, 2007.

[15] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 791–804, 2012.

[16] G. Tomasi and R. Bro, "A comparison of algorithms for fitting the PARAFAC model," *Computational Statistics & Data Analysis*, vol. 50, no. 7, pp. 1700–1734, 2006.

[17] R. Bro and H. A. L. Kiers, "A new efficient method for determining the number of components in PARAFAC models," *Journal of Chemometrics*, vol. 17, pp. 274–286, 2002.

[18] H. A. L. Kiers and A. der Kinderen, "A fast method for choosing the number of components in tucker3 analysis," *British Journal of Mathematical and Statistical Psychology*, vol. 56, pp. 119–125, 2003.

[19] B. Bader, T. Kolda, et al., "MATLAB tensor toolbox version 2.5," Available online, January 2012.

[20] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[21] Center for Cancer Research, National Cancer Institute, "Clinical Proteomics Program," Available online.

[22] E. F. Petricoin *et al.*, "Serum proteomic patterns for detection of prostate cancer," *Journal of the National Cancer Institute*, vol. 94, no. 20, pp. 1576–1578, 2002.

[23] Q. Xu *et al.*, "Mass spectrometry-based proteomic pattern analysis for prostate cancer detection using neural networks with statistical significance test-basedd feature selection," pp. 837–842, 2009.

[24] E. F. Petricoin *et al.*, "Use of proteomic patterns in serum to identify ovarian cancer," *The Lancet*, , no. 359, pp. 572–577, 2002.

[25] L. Li *et al.*, "Application of the GA/KNN method to SELDI proteomics data," *Bioinformatics*, vol. 20, no. 10, pp. 1638–1640, 2003.