Interpretable Low-Rank Document Representations with Label-Dependent Sparsity Patterns

Ivan Ivek

Rudjer Boskovic Institute, Bijenicka 54, 10000 Zagreb, Croatia

Abstract. In context of document classification, where in a corpus of documents their label tags are readily known, an opportunity lies in utilizing label information to learn document representation spaces with better discriminative properties. To this end, in this paper application of a Variational Bayesian Supervised Nonnegative Matrix Factorization (supervised vbNMF) with label-driven sparsity structure of coefficients is proposed for learning of discriminative nonsubtractive latent semantic components occuring in TF-IDF document representations. Constraints are such that the components pursued are made to be frequently occuring in a small set of labels only, making it possible to yield document representations with distinctive label-specific sparse activation patterns. A simple measure of quality of this kind of sparsity structure, dubbed inter-label sparsity, is introduced and experimentally brought into tight connection with classification performance. Representing a great practical convenience, inter-label sparsity is shown to be easily controlled in supervised vbNMF by a single parameter.

Keywords: Document Categorization, Latent Semantic Analysis, Supervised Sparse Nonnegative Matrix Factorization, Variational Bayes

1 Introduction

As an essential step in machine learning applications which both efficiency and quality of learning depend on, dimensionality reduction has become a well covered subject of research [MPH09] which produced archetipal linear methods with low-rank assumptions such as Principal Component Analysis (PCA) [Jol02] and Nonnegative Matrix Factorization (NMF) [LS99], as well as their kernelized and generally non-linear variants, to touch upon some. Originally they have been formulated as entirely unsupervised methods. However, in supervised and semi-supervised learning applications, where labels of learning samples are readily available, it may be appealing to use this information to obtain lower-dimensional representations of data which not only attempt to preserve the original variance in the data, but also promise to deliver representation spaces with better discriminative properties. A well known representative which incorporates this desideratum is Fisher's Linear Discriminant Analysis (FLD) [MK01].

In recent relevant literature there is a pronounced trend of using probabilistic generative models for this purpose. Probabilistic approaches to learning lie on a well developed mathematical apparatus which offers flexible enough modeling of prior knowledge in form of graphical models, supported by well-known meta-algorithms for estimating model parameters. Of this family of algorithms, along with Probabilistic Latent Semantic Analysis (pLSA) [Hof99], a common probabilistically formulated baseline algorithm in text mining is Latent Dirichlet Allocation (LDA) [BNJ12] with its more recent discriminative modifications [BM07][LJSJ08], as well as probabilistic formulations of sparse NMF [Cem09] and their supervised counterparts [Ive14].
Sparse coding is known to result in efficient and robust representations which have proven suitable for applications such as data compression, denoising and missing data imputting [Mal99]. On the other hand, representations obtained discriminatively are suitable for classification purposes. Combining those two properties, the basis of this work is a probabilistically formulated method for sparse additive representations of data using nonnegative latent components which are of sparsity structure additionally driven by data labeling [Ive14]. In context of document classification, the decomposition is suitable for finding interpretable patterns of semantically related terms, with high discriminative potential.

1.1 Document Feature Spaces

Disregarding syntactic and semantic interrelations of words, the simplest and most often used intermediate form for document representation is bag-of-words; after tokenization, purification and stemming, frequency of relevant terms is determined for each document resulting in representations of documents as frequencies of particular terms. Models such as LDA have a natural interpretation when decomposing bag-of-words representations, while other approaches may benefit from TF-IDF weighting [RU11] which heuristically measures the importance of a term for a particular document in a specific corpus of documents. For a term with index τ in ν-th document, as a product of two measures,

$$tfidf_{\nu\tau} = tf_{\nu\tau} * idf_{\tau}, \tag{1}$$

TF-IDF score is proportional to (normalized) frequency of a particular term in a document,

$$tf_{\nu\tau} = \frac{\#_{\nu\tau}}{\max_t(\#_{\nu t})}, \tag{2}$$

but stunted by a measure of how rare this term occurs in the entire corpus,

$$idf_{\tau} = \ln \frac{N}{n_{\tau}}, \tag{3}$$

where the number of occurences of term τ in ν-th document is denoted by $\#_{\nu\tau}$, the number of documents in the corpus by N and the number of documents which contain term τ at least once by n_{τ}.

1.2 NMF as a Tool for Latent Semantic Analysis

Bag-of-words-based approaches to text mining are known to suffer from problems of polysemy and synonymy of terms. These problems can be alleviated by representing documents in spaces of patterns of frequencies of semantically related terms rather than in the original space of term frequencies [DDF+90]. Luckily, algorithms for learning of such representations exist, of which perhaps the best known are pLSA formulations. Also assuming inherent nonnegativity in the data, NMF decompositions can be interpreted the same way as pLSA, revealing patterns of semantically related terms underlying the data. Furthermore, a specific connection worth mentioning is that a NMF formulation based on generalized KL-divergence minimizes the exactly same objective function as the original pLSA formulation does [GG05].
Nonnegativity is a reasonable assumption and a desireable bias when modeling either term frequencies or derived intermediate document representations such as TF-IDF. In general, NMF aims at decompositions in form of $\boldsymbol{X}\approx\boldsymbol{T}\boldsymbol{V}$, where $\boldsymbol{X}$, $\boldsymbol{T}$ and $\boldsymbol{V}$ are all nonnegative matrices. Although the decomposition is nonunique in general, to some extent nonuniqueness may be compensated for by adding additional bias in the model, of which most prominent is sparsity of solution [LS99]. Sparsity is enforced in divergence-based NMF by different sparsity promoting regularizers, e.g. [Hoy04], and in probabilistic formulations by imposing sparse prior distributions on the coefficients [Cem09].
Throughout this paper, in context of document representation for categorization purposes, $\boldsymbol{X}$ will be regarded as a collection of documents organized columnwise and represented by TF-IDF features, $\boldsymbol{T}$ as a low-rank collection of latent semantic components organized columnwise, and $\boldsymbol{V}$ as matrix of coefficients when $\boldsymbol{X}$ is projected onto the space of latent semantic components $\boldsymbol{T}$. In other words, each document is modeled as a strict superposition of the nonnegative latent semantic components.

2 Methodology

2.1 Supervised NMF Model

The generative model [Ive14] assumes that each column of data, $x_{:\tau}$, is a result of latent components $t_{:i}$ consisting of independent gamma-distributed variables,

$$p\left(t_{\nu i}|a^t_{\nu i}, b^t_{\nu i}\right) = \mathcal{G}\left(t_{\nu i}|a^t_{\nu i}, b^t_{\nu i}\right), \tag{4}$$

interacting through linear mixing with coefficients $v_{i\tau}$ under Poissonian noise:

$$p\left(s_{\nu i\tau}|t_{\nu i}, v_{i\tau}\right) = \mathcal{P}\left(s_{\nu i\tau}|t_{\nu i}, v_{i\tau}\right) \tag{5}$$

$$p\left(x_{\nu\tau}|s_{\nu:\tau}\right) = \delta\left(x_{\nu\tau} - \sum_i s_{\nu i\tau}\right). \tag{6}$$

Mixing coefficients $v_{i\tau}$ are assumed to be exponentially distributed with different scale parameters for different selections of label indicators $z_\tau \in \mathcal{L}$, formulated as mixtures of variables λ_{il} with z_τ as discrete numerical mixture selection variables,

$$p\left(v_{i\tau}|z_\tau, \lambda_{i:}\right) = \mathcal{G}\left(v_{i\tau}\middle|1, \sum_{l\in\mathcal{L}} \delta(z_\tau - l)\lambda_{il}^{-1}\right) \tag{7}$$

Note that label indicators z_τ are elements of a discrete set of (integer) numbers $\mathcal{L}$ for convenience of notation. Variables λ_{il}^{-1}, representing expectations of magnitudes of coefficient components i for all samples labeled as l are constrained by inverse-gamma priors,

$$p\left(\lambda_{il}\middle|a_{il}^{\lambda}, b_{il}^{\lambda}\right) = \mathcal{G}\left(\lambda_{il}\middle|a_{il}^{\lambda}, b_{il}^{\lambda}\right). \tag{8}$$

Because inverse-gamma is a heavy-tailed distribution, by setting the probability mass to be concentrated around some small value, significantly larger values of λ_{il}^{-1} will occur rarely. Thus, such a prior imposes an additional bias to produce models having only a minority of indicators λ_{il}^{-1} with significantly large mean values on average, which, hierarchically propagating to activation coefficients $v_{i\tau}$, constrain samples having the same label to have only a small shared subgroup of latent patterns significantly active.
Using compact notation

$$\begin{aligned}
p\left(\boldsymbol{X}|\boldsymbol{S}\right) &= \prod_{\nu,\tau} p\left(x_{\nu\tau}|s_{\nu:\tau}\right) \\
p\left(\boldsymbol{S}|\boldsymbol{T},\boldsymbol{V}\right) &= \prod_{\nu,\tau} p\left(s_{\nu:\tau}|t_{\nu i}, v_{i\tau}\right) \\
p\left(\boldsymbol{T}\middle|\boldsymbol{A^t},\boldsymbol{B^t}\right) &= \prod_{\nu,i} p(t_{\nu i}|a_{\nu i}^t, b_{\nu i}^t) \\
p\left(\boldsymbol{V}\middle|\boldsymbol{\Lambda},\vec{z}\right) &= \prod_{i,\tau} p\left(v_{i\tau}|\lambda_{i:}, z_\tau\right) \\
p\left(\boldsymbol{\Lambda}\middle|\boldsymbol{A^\lambda},\boldsymbol{B^\lambda}\right) &= \prod_{i,l} p\left(\lambda_{il}\middle|a_{il}^{\lambda}, a_{il}^{\lambda}\right),
\end{aligned}$$

joint distribution of the supervised NMF model can be written as

$$\begin{aligned}
&p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\middle|\boldsymbol{A^t},\boldsymbol{B^t},\boldsymbol{A^\lambda},\boldsymbol{B^\lambda},\vec{z}\right) \\
&\quad = p\left(\boldsymbol{X}|\boldsymbol{S}\right) p\left(\boldsymbol{S}|\boldsymbol{T},\boldsymbol{V}\right) p\left(\boldsymbol{T}\middle|\boldsymbol{A^t},\boldsymbol{B^t}\right) p\left(\boldsymbol{V}\middle|\boldsymbol{\Lambda},\vec{z}\right) p\left(\boldsymbol{\Lambda}\middle|\boldsymbol{A^\lambda},\boldsymbol{B^\lambda}\right).
\end{aligned} \tag{9}$$

Linear mixing as described by (4), (5) and (6) is the same as in Poisson-gamma NMF [Cem09]. Equations (7) and (8) additionally formulate a sparsity structure abstracted from the level of data samples to the level of labels, making it possible to pursue decompositions with recognizable sparsity patterns characteristic of data samples which share the same label tag.

2.2 Variational Bayesian Learning Algorithm

To give a concise outline of general treatment of learning by VB, let the observed variables be denoted by $\boldsymbol{D}$, the hyperparameters of a model by $\boldsymbol{H}$ and both the unobserved variables and the model parameters by $\boldsymbol{\Theta}$. Minimization of discrepancy between posterior $p\left(\boldsymbol{\Theta}|\boldsymbol{D},\boldsymbol{H}\right)$ (which is in general difficult to optimize directly, especially in a fully Bayesian manner) and an introduced instrumental approximation $q(\boldsymbol{\Theta})$ measured by Kullback-Liebler divergence gives rise to a lower bound on the posterior,

$$\mathcal{L} = \left\langle \ln p\left(\boldsymbol{D},\boldsymbol{\Theta}|\boldsymbol{H}\right)\right\rangle_{q(\boldsymbol{\Theta})} + \mathcal{H}\left[q(\boldsymbol{\Theta})\right], \tag{10}$$

where entropy of the probability density function in the argument is denoted by $\mathcal{H}\left[.\right]$. Supposing that $q(\boldsymbol{\Theta})$ is of factorized form $q(\boldsymbol{\Theta}) = \prod_{\alpha\in C} q(\boldsymbol{\Theta_\alpha})$, it can be shown that the iterative local updates at iteration (n+1) alternating over C in form of

$$q(\boldsymbol{\Theta_\alpha})^{(n+1)} \propto \exp\left(\left\langle \ln p\left(\boldsymbol{D},\boldsymbol{\Theta}|\boldsymbol{H}\right)\right\rangle_{\frac{q(\boldsymbol{\Theta})^{(n)}}{q(\boldsymbol{\Theta_\alpha})^{(n)}}}\right) \tag{11}$$

improve the lower bound (10) monotonically. Moreover, should the model be conjugate-exponential, for a fully factorized approximation, expressions in (11) neccessarily assume analytical forms [Win03].
For the model (9) variational Bayesian update expressions are derived from (11) by specifying $p\left(\boldsymbol{D},\boldsymbol{\Theta}|\boldsymbol{H}\right) = p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\left|\boldsymbol{A^t}\right.,\boldsymbol{B^t},\boldsymbol{A^v},\boldsymbol{B^v}\right)$ together with instrumental distribution $q\left(\boldsymbol{\Theta}\right) = q\left(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\right)$, appropriatedly factorized as $q\left(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\right) = \prod_{\nu,\tau} q\left(s_{\nu:\tau}\right)\prod_{\nu,i} q\left(t_{\nu i}\right)\prod_{i,\tau} q\left(v_{i\tau}\right)\prod_{i,l} q\left(\lambda_{il}\right)$ for computational convenience. Still, the lower bound according to (10) includes a difficult term related to optimization of $q\left(\lambda_{il}\right)$. For this reason, the lower bound has been relaxed using Jensen's inequality and optimization is done with respect to this relaxed bound [Ive14]. An outline of the treatment of the learning algorithm can be found in Appendix B.

3 Experiments

All experiments have been performed on *20Newsgroups*[1] dataset, *bydate* version split into training and test sets; rather than estimating the generalization error of classification by crossvalidation techniques, the underlying ambition is merely to explore peak potentials of classification using different representation spaces evaluated on a single train-test split in same conditions.

3.1 Dataset

Experiments have been performed on *20Newsgroups* dataset sorted by date wih duplicates and headers removed, with documents having multiple labels left out

[1] Available from Jason Rennie's web Page, http://qwone.com/~jason/20Newsgroups/

and preprocessed to obtain a bag-of-words representation. The dataset is split into a training set and a test set. To alleviate computational load, the set of features has been heuristically reduced to 10000 terms, based on maximum TF-IDF score accross all documents.

3.2 Experimental Setup

Representation spaces in which consequently classification takes place which are taken under consideration are the ones obtained by PCA, Poisson-gamma unsupervised vbNMF [Cem09], and the supervised vbNMF, all decomposing the matrix of TF-IDF scores of the training set only. Having learned a specific space of reduced dimensionality, representation of the test set in this space is found by projection on the vector basis in case of PCA or by optimizing the matrix of coefficients only using Poisson-gamma vbNMF formulation (i.e. the matrix of latent components is fixed to what has been learned in the training step) in case of both unsupervised and supervised vbNMF methods.
For Poisson-gamma vbNMF sparse decompositions have been pursued by fixing shape hyperparameters of the gamma distributed coefficients to a value less than or equal to 1 throughout the entire run, while other hyperparameters (constrained to be the same for all elements of matrices $\boldsymbol{T}$ and $\boldsymbol{V}$, a single one for each of the matrices) have been optimized automatically by maximization of the lower bound directly in a non-Bayesian manner [Cem09]. For supervised vbNMF, hyperparameters aLambda have been fixed and varied, while other hyperparameters have been left to the algorithm to optimize, by direct optimization as in [Cem09]. Specifically, following initialization, λ parameters are chosen to be all equal and fixed for a burn-in period of 10 iterations, not until after which they start to get optimized according to the algorithm in Table 3.
To accentuate the predictive potentials of the considered representation spaces by themselves, rather than in conjunction with a strong classifier, the classifier of choice is k-NN using cosine similarity metric, with k chosen heuristically as the square root of the cardinality of the training set.
Dimension of space of latent components has been varied as a parameter for all decomposition methods. Because at each run the NMF algorithms converge to some local minimum, to explore these local minima, for each parameter set they have been run 10 times with random initializations.

3.3 Evaluation

Metrics of classification performance used in the experiments are micro-averaged accuracy, defined as

$$a^{micro} = \frac{\sum_l N_l^{correct}}{\sum_l N_l^{all}},$$

and macro-averaged accuracy,

$$a^{macro} = \frac{1}{L} \sum_l \frac{N_l^{correct}}{N_l^{all}},$$

where the number of correctly classified documents belonging to the l-th label is denoted by $N_l^{correct}$, the number of documents belonging to l-th label in the test split by N_l^{all} and the number of labels by L. By averaging the accuracies calculated separatedly for each of the labels, macro-averaged accuracy compensates for label-imbalance of test datasets.
As a measure of sparsity, Hoyer's measure [Hoy04], based on ratio of l1 and l2 norms and originally introduced in context of NMF penalization, will be used. For a vector $\vec{x} = [x_1, ..., x_n]^T$ it is defined as

$$sparsity\left(\vec{x}\right) = \frac{1}{\sqrt{n}-1}\left(\sqrt{n} - \frac{\sum_i |x_i|}{\sqrt{(\sum_i x_i^2)}}\right), \tag{12}$$

taking value of 1 in case only a single element is non-zero (maximum sparsity), and a value of 0 if all elements are equal (minimum sparsity). For the purpose of this paper, when referring to sparsity of matrices, matrix is assumed to be vectorized first by appending its columns, then treating it as a vector according to (12).
If labels in a document corpus are meaningfully assigned based on topics of documents, then meaningful discovered latent semantic components are expected to have specific patterns of occurence for documents belonging to a specific label. Using supervised vbNMF, those patterns are modeled as patterns in sparsity of coefficients (i.e. in patterns of support of sparse coefficients) that documents labeled the same have in common. To measure the consistency of occurence of sparsity patterns in labels, let a representation by coefficients of N documents in I dimensional space be denoted by $\boldsymbol{V} \in \mathbb{R}^{IxN}$, i.e. n-th document is represented by coefficient vector $[\boldsymbol{V}]_{:n}$, and let sums of coefficient sets which share the same label be accumulated in matrix $\boldsymbol{L} \in \mathbb{R}^{IxL}$, where L is number of labels as

$$[\boldsymbol{L}]_{:l} = \sum_{n \in N_l} [\boldsymbol{V}]_{:n}, \tag{13}$$

where n iterates over subset of document indices with the same label, N_l.
Now, inter-label sparsity can be introduced, defined as sparsity of matrix $\boldsymbol{L}$. The motivation behind (13) is that $l0$ norm of a sum of vectors with the same sparsity pattern (same support) is the same as the exclusive $l0$ norm of such vectors by themselves, and, the more those vectors deviate from the pattern (i.e. when the vectors have differing supports), the larger the $l0$ norm of the sum will be. Note that the latter rationale holds exactly for $l0$ definition of sparsity, while for more relaxed definitions of sparsity such as (12) the behavior will be only qualitatively similar. For the purpose of this paper, sparsity of (13) will be measured as Hoyer's sparsity (12).

3.4 Results and Discussion

For comparison, as a baseline, classification results of PCA are plotted against the dimension of representation space on Fig. 1. For unsupervised vbNMF,

micro-averaged accuracies averaged accross random initializations for different shape parameters with varying number of latent semantic components are shown in Fig. 2.

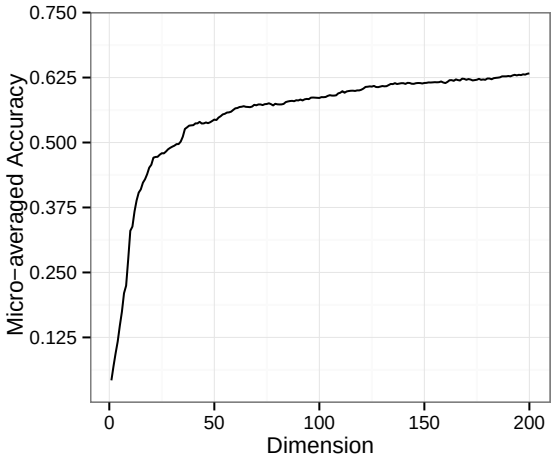

Fig. 1. Classification results using PCA.

Compared to PCA, even with a larger dimension of representation space, vbNMF with sparsity constraints did not bring improvements on average, regardless of the degree of sparsity penalization. The explanation is that, even though sparse representation spaces may be good for clustering, natural clusters may differ greatly from labeling and consequently even be detrimental to classification applications [BM07] when compared to dense representations such as PCA. Better representations for classification purposes are expected to be found by introducing label information to the model, which in spaces obtained by supervised vbNMF (Fig. 3.) indeed manifested as a boost in classification performance.
Both unsupervised vbNMF and supervised vbNMF consistently resulted in sparse decompositions. However, label-driven structure present in supervised vbNMF decompositions (engineered as to be the sole difference in the experiments) is to be accounted for the beneficial effect observed. Examples of sparsities accross labels according to (13) are visualized on on Fig. 4. for the sparse unsupervised vbNMF decomposition which produced peak micro-averaged accuracy of 0.5796 and on Fig. 5. for an arbitrarily chosen supervised variant with matching dimension. The supervised variant produced distinctive sparsity patterns accross labels, which is also reflected quantitatively on inter-label sparsity of the decomposition.
The connection between sparsity on the level of labels and classification performance is further explored using Fig. 6., showing data for all supervised representations obtained in the experiments. Variance of the scatter plot becomes

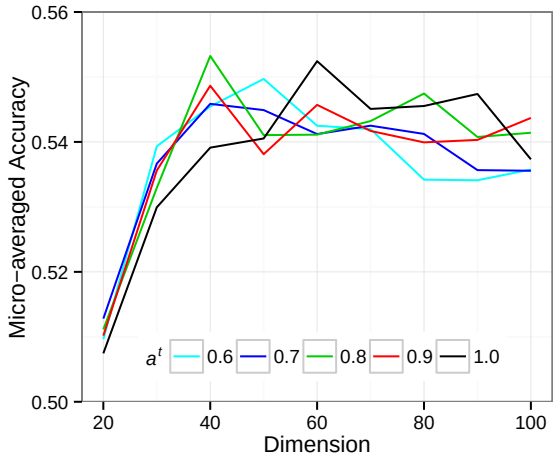

Fig. 2. Classification results of unsupervised vbNMF (averaged accross 10 random initializations) with varying level of sparsity penalization.

tighter with increasing the dimension of representation space, meaning that for a sufficiently large dimension of the decomposition, inter-label sparsity is indeed a good predictor for classification quality on this dataset.

Equally importantly, experiments show that in case of supervised vbNMF, inter-label sparsity can elegantly be controlled by a single parameter a^λ alone, regardless of dimension: as illustrated by Fig. 7., a logarithmic increase of a^λ is accompanied by a trend of growth of inter-label sparsity, only to be broken by too extreme regularizations, when tails of the prior have little mass. On the other hand, unsupervised vbNMF resulted in moderate levels of inter-label sparsity because sparsity structure is supported by the structure of data features only, with somewhat higher values in cases of very strong sparsity regularizations and an impractically small number of latent patterns.

Classification results using k-NN classifier with heuristically chosen k on representation spaces obtained by the three methods are summarized in Table 1., reporting peak value of its micro- and macro-averaged accuracies; for the set of parameters which yielded the peak performance, corresponding accuracies averaged accross the 10 random initializations together with minimal achieved accuracies are reported. On Fig. 3.b), showing smooth dependence of micro-averaged accuracy (averaged across random initializations) on an interesting range of a^λ for a selection of dimensions, peak performance as entered in Table 1. can be noticed marked.

To conclude the remarks on the experiments, it is worth mentioning that, if sparse representations are pursued, care is advised when choosing and optimizing hyperparameters by non-Bayesian minimization of bound. Because sparse constraints both on matrices $\boldsymbol{T}$ and $\boldsymbol{V}$ act as two competing penalizations, useful decompositions are obtained more easily by constraining only one of the matri-

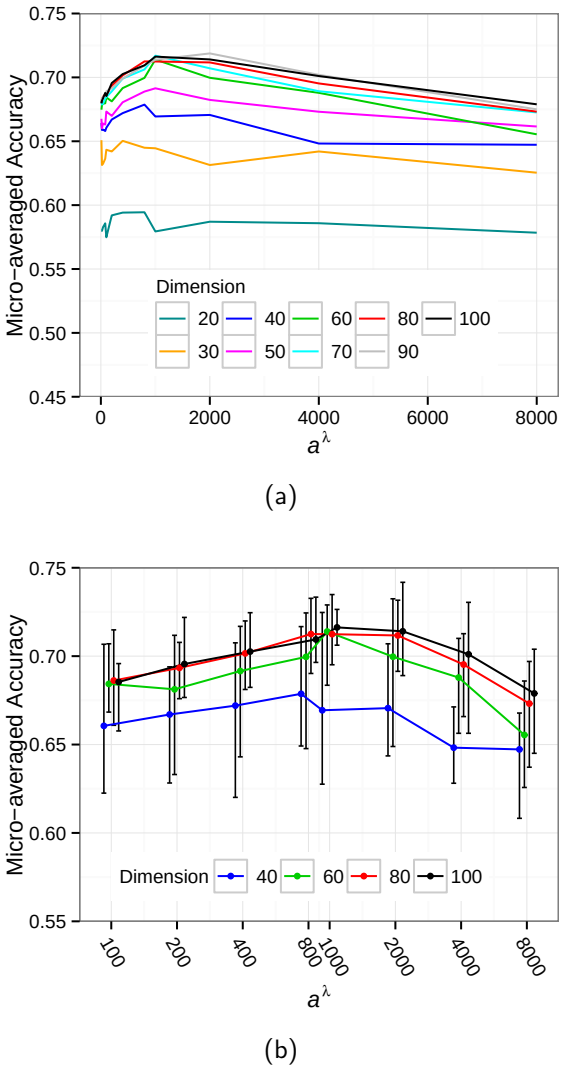

Fig. 3. Supervised vbNMF classification results, averaged over 10 random initializations. a) Dependence on level of sparsity penalization, varying dimensions of representation spaces. b) Dependence on level of sparsity penalization, varying dimensions of representation spaces. Error bars represent maximum and minimum values among the random initializations. x-axis is shown on logarithmic scale.

Fig. 4. Sparsity accross labels according to (13) for the unsupervised vbNMF decomposition with best classification performance; the more intense the tone of blue, the higher the sparsity. Each of the 40 latent semantic components is represented by its 5 most significant terms. Coefficient sparsity is 0.6862, inter-label sparsity is 0.5784.

Fig. 5. Sparsity accross labels according to (13) for an arbitrarily chosen supervised vbNMF decomposition with 40 latent semantic components, each represented by its 5 most significant terms. Coefficient sparsity is 0.8752, inter-label sparsity is 0.8578.

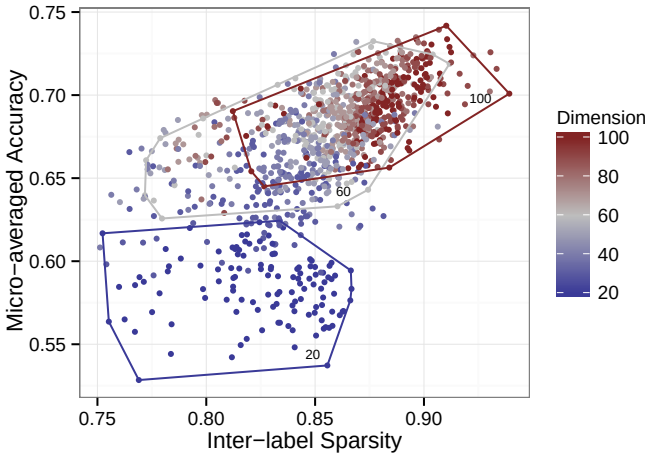

Fig. 6. Correlation between micro-averaged accuracy and inter-label sparsity. Each point in the scatter plot represents a single supervised NMF decomposition. Convex hulls contain points corresponding to choices of dimensions of 20, 60 and 100.

ces to be sparse - either the matrix of latent components to obtain a parts-based representation, or the matrix of coefficients to obtain a sparse representation of data. So, when optimizing the shape parameter of one of the matrices in such a manner next to a fixed hyperparameter of the other matrix which is to be made sparse, due to the automated (and, equally importantly, non-Bayesian) nature of the optimization the former may also come to describe a sparse distribution and in effect impede the desired bias toward the desired type of sparsity.

	Algorithm	PCA	Unsupervised NMF	Supervised NMF
Micro-averaged Accuracy	[Min,Max]		[0.5190,0.5796]	[0.6890,**0.7418**]
	Mean	0.6330	0.5532	**0.7141**
	Dimension	200	40	100
Macro-averaged Accuracy	[Min,Max]		[0.5033,0.5655]	[0.6758,**0.7277**]
	Mean	0.6179	0.5393	**0.6997**
	Dimension	200	40	100

Table 1. Summary of experimental results

4 Conclusion

It has been well documented that using label information in low-rank representation learning is vital to obtain representations with good discriminative

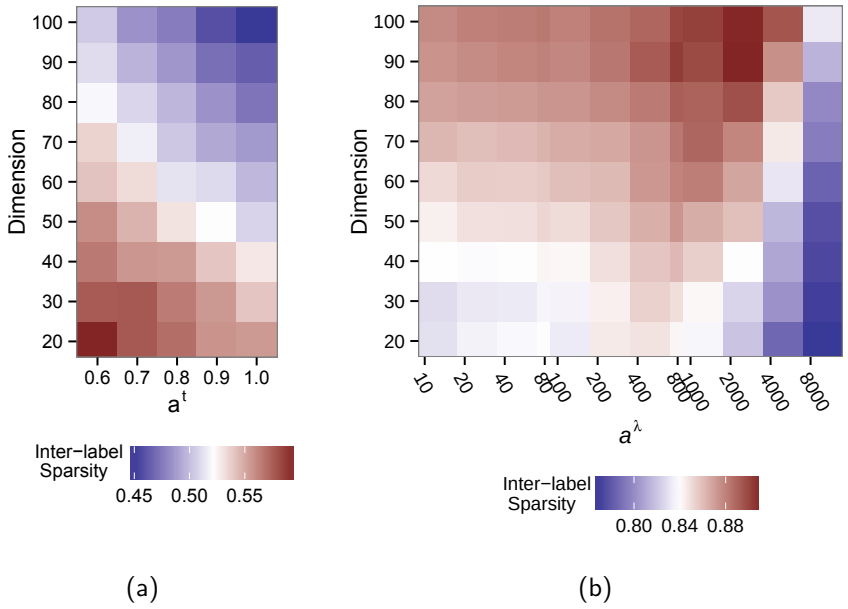

Fig. 7. Dependence of inter-label sparsity (averaged over 10 random initializations) on dimension of representation space and parameters which control sparsity. a) Unsupervised vbNMF with sparsity constraints. b) Supervised vbNMF; inter-label sparsity can be controlled by a^{λ}.

properties. In this context, applied to classification of a document corpus, a probabilistic learning algorithm which combines sparse coding and supervised learning has been presented.

To characterize advantages of using label information, two extreme cases have been juxtaposed, the presented supervised model and a fully unsupervised one, belonging to the same family, having the same noise model and using the same metaalgorithm for parameter learning.

A qualitative inspection motivated the introduction of the notion of inter-label sparsity, abstracting sparsity of coefficients on the level of documents to sparsity on the level of document labels. Experiments point to a strong connection between the inter-label sparsity of the representation and the classification performance metrics. Furthermore, inter-label sparsity of decompositions obtained by supervised vbNMF can elegantly be controlled by a single parameter. However, even though sparsity and nonnegativity constraints intuitively seem appropriate and result in compact and interpretable document representations, a question remains whether there is any actual advantage in using sparse representations over dense ones as classification precursors.

As quality of representation spaces has been primarily addressed in this work,

little regard has been given to quality of the classifier *per se*. Because it is reasonable to expect that a stronger classifier would result in even better classification results, it would be interesting to compare a well-tuned classifier in the representation spaces obtained by supervised vbNMF to state-of-the-art aproaches in the field, on benchmark datasets. Future work based on semi-supervised modifications of the model is considered, to make the model more flexible and applicable in more commonly occuring, semi-supervised, scenarios.

Acknowledgments This work was supported by the Croatian Ministry of Science, Education and Sports through the project "Computational Intelligence Methods in Measurement Systems", No. 098-0982560-2565.

References

BM07. David M Blei and J McAuliffe. Supervised Topic Models. *Neural Information Processing Systems*, 21, 2007.

BNJ12. David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2012.

Cem09. Ali Taylan Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*, 2009, January 2009.

DDF+90. Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

GG05. Cyril Goutte and Eric Gaussier. Relation between PLSA and NMF and implications. *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 601–602, 2005.

Hof99. Thomas Hofmann. Probabilistic Latent Semantic Analysis. In *Uncertainity in Artifitial Intelligence - UAI'99*, page 8, 1999.

Hoy04. Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

Ive14. Ivan Ivek. Supervised Dictionary Learning by a Variational Bayesian Group Sparse Nonnegative Matrix Factorization. May 2014.

Jol02. I T Jolliffe. *Principal Component Analysis*, volume 98. 2002.

LJSJ08. S Lacoste-Julien, F Sha, and MI Jordan. DiscLDA: Discriminative learning for dimensionality reduction and classification. *NIPS*, pages 897–904, 2008.

LS99. D D Lee and H S Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

Mal99. Stéphane Mallat. *A Wavelet Tour of Signal Processing*. 1999.

MK01. Aleix M. Martinez and Avinash C. Kak. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:228–233, 2001.

MPH09. L J P van der Maaten, E O Postma, and H J van den Herik. Dimensionality Reduction: A Comparative Review. *Journal of Machine Learning Research*, 10:1–41, 2009.

RU11. Anand Rajaraman and Jeffrey D Ullman. Mining of Massive Datasets. *Lecture Notes for Stanford CS345A Web Mining*, 67:328, 2011.

Win03. John M Winn. Variational message passing and its applications. *Ph.D. thesis, Department of Physics, University of Cambridge*, 2003.

A Probability Density Functions

By $\Psi(.)$ digamma function, $\Psi(x) = \frac{d}{dx} \ln \Gamma(x)$, is denoted.

A.1 Poisson Probability Density Function

Definition:

$$\mathcal{P}(x|\lambda) = e^{(-\lambda + x \ln \lambda - \ln \Psi(x+1))}, \lambda > 0.$$

Sufficient statistics:

$$\langle x \rangle = \lambda.$$

A.2 Gamma Probability Density Function

Definition:

$$\mathcal{G}(x|a,b) = e^{\left(-\frac{1}{b}x + (a-1)\ln x - a \ln b - \ln \Psi(a)\right)}, a > 0, b > 0.$$

Sufficient statistics:

$$\langle x \rangle = ab,$$
$$\langle \ln x \rangle = \Psi(a) + \ln b.$$

Entropy:

$$\mathcal{H}\left[\mathcal{G}(x|a,b)\right] = -(a-1)\Psi(a) + \ln b + a + \ln \Gamma(a).$$

A.3 Multinomial Probability Density Function

Definition:

$$\mathcal{M}\left(\overrightarrow{x} \middle| s, \overrightarrow{p}\right) = \delta\left(s - \sum_i x_i\right) e^{\left(\ln \Gamma(s+1) + \sum_i (x_i \ln p_i - \ln \Gamma(x_i+1))\right)},$$
$$\sum_i p_i = 1, \sum_i x_i = s.$$

Sufficient statistics:

$$\begin{bmatrix} \langle \ln x_1 \rangle \\ \vdots \\ \langle \ln x_C \rangle \end{bmatrix} = s \begin{bmatrix} p_1 \\ \vdots \\ p_C \end{bmatrix}.$$

Entropy:

$$\mathcal{H}\left[\mathcal{M}\left(\overrightarrow{x} \middle| s, \overrightarrow{p}\right)\right] = -\ln \Gamma(s+1) - \sum_i \langle x_i \rangle \ln p_i + \sum_i \langle \ln \Gamma(x_i+1) \rangle - \left\langle \ln \delta\left(s - \sum_i x_i\right) \right\rangle.$$

B Variational Bayesian Learning Algorithm

Iterative alternating update rules follow from (11) by plugging in

$$p\left(\boldsymbol{D},\boldsymbol{\Theta}\right)=p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\left|\boldsymbol{A}^{\boldsymbol{t}}\right.,\boldsymbol{B}^{\boldsymbol{t}},\boldsymbol{A}^{\boldsymbol{v}},\boldsymbol{B}^{\boldsymbol{v}}\right)$$

and

$$q\left(\boldsymbol{\Theta}\right)=q\left(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\right).$$

Derivation of the iterative alternating update rules will be sketched as follows: for each instrumental distribution first its analytical form will be specified as follows from (11), followed by its natural parameters in the top and its sufficient statistics in the bottom row of the following table. Notice that the algorithm stores these distributions from iteration to iteration in form of sufficient statistics.

$$\begin{aligned}q\left(s_{\nu:\tau}\right)^{(t+1)} &\propto e^{\left\langle \ln p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\left|\boldsymbol{A}^{\boldsymbol{t}},\boldsymbol{B}^{\boldsymbol{t}},\boldsymbol{A}^{\boldsymbol{\lambda}},\boldsymbol{B}^{\boldsymbol{\lambda}},\vec{z}\right.\right)\right\rangle_{\frac{q(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda})^{(t)}}{q(s_{\nu:\tau})^{(t)}}}} \\ &= \mathcal{M}\left(s_{\nu:\tau}|x_{\nu\tau},{p_{\nu:\tau}}^{(t)}\right),\end{aligned}$$

${p_{\nu i\tau}}^{(t)}=\dfrac{\exp\left(\left\langle \ln t_{\nu i}\right\rangle^{(t)}+\left\langle \ln v_{i\tau}\right\rangle^{(t)}\right)}{\sum_i \exp\left(\left\langle \ln t_{\nu i}\right\rangle^{(t)}+\left\langle \ln v_{i\tau}\right\rangle^{(t)}\right)}$
$\left\langle s_{\nu i\tau}\right\rangle^{(t+1)}=x_{\nu\tau}{p_{\nu i\tau}}^{(t)}$

$$\begin{aligned}q\left(t_{\nu i}\right)^{(t+1)} &\propto e^{\left\langle \ln p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\left|\boldsymbol{A}^{\boldsymbol{t}},\boldsymbol{B}^{\boldsymbol{t}},\boldsymbol{A}^{\boldsymbol{\lambda}},\boldsymbol{B}^{\boldsymbol{\lambda}},\vec{z}\right.\right)\right\rangle_{\frac{q(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda})^{(t)}}{q(t_{\nu i})^{(t)}}}} \\ &= \mathcal{G}\left(t_{\nu i}\left|{\alpha^t_{\nu i}}^{(t)},{\beta^t_{\nu i}}^{(t)}\right.\right)\end{aligned}$$

${\alpha^t_{\nu i}}^{(t)}=a^t_{\nu i}+\sum_\tau \left\langle s_{\nu i\tau}\right\rangle^{(t)}$ ${\beta^t_{\nu i}}^{(t)}=\left({b^t_{\nu i}}^{-1}+\sum_\tau \left\langle v_{i\tau}\right\rangle^{(t)}\right)^{-1}$
$\left\langle t_{\nu i}\right\rangle^{(t+1)}={\alpha^t_{\nu i}}^{(t)}{\beta^t_{\nu i}}^{(t)}$ $\left\langle \ln t_{\nu i}\right\rangle^{(t+1)}=\Psi\left({\alpha^t_{\nu i}}^{(t)}\right)+\ln {\beta^t_{\nu i}}^{(t)}$

$$q\left(v_{i\tau}\right)^{(t+1)} \propto e^{\left\langle \ln p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V}\middle|\boldsymbol{A^t},\boldsymbol{B^t},\boldsymbol{A^v},\boldsymbol{B^v}\right)\right\rangle_{\frac{q(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda})^{(t)}}{q(v_{i\tau})^{(t)}}}}$$
$$= \mathcal{G}\left(v_{i\tau}\left|\alpha_{i\tau}^{v}{}^{(t)}, \beta_{i\tau}^{v}{}^{(t)}\right.\right)$$

$$\alpha_{i\tau}^{v}{}^{(t)} = 1 + \sum_{\nu} \left\langle s_{\nu i\tau}\right\rangle^{(t)}$$
$$\beta_{i\tau}^{v}{}^{(t)} = \left(\sum_{l} \delta(z_\tau - l) \left\langle \lambda_{il}\right\rangle^{(t)} + \sum_{\nu} \left\langle t_{\nu i}\right\rangle^{(t)}\right)^{-1}$$

$$\left\langle v_{i\tau}\right\rangle^{(t+1)} = \alpha_{i\tau}^{v}{}^{(t)} \beta_{i\tau}^{v}{}^{(t)}$$
$$\left\langle \ln v_{i\tau}\right\rangle^{(t+1)} = \Psi\left(\alpha_{i\tau}^{v}{}^{(t)}\right) + \ln \beta_{i\tau}^{v}{}^{(t)}$$

Update expressions for $q\left(\lambda_{il}\right)^{(t+1)}$ depend on a difficult term in the bound. Based on concavity of logaritmic function, this term can be lower bounded using Jensen's inequality:

$$\left\langle \ln\left(\sum_{l} \delta(z_\tau - l)\lambda_{il}\right)\right\rangle_{\frac{q(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda})^{(t)}}{q(\lambda_{il})^{(t)}}} \geq \sum_{l} \delta(z_\tau - l) \left\langle \ln \lambda_{il}\right\rangle_{\frac{q(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda})^{(t)}}{q(\lambda_{il})^{(t)}}}.$$

In this relaxed bound convergence is preserved and now updates have analytical forms:

$$q\left(\lambda_{il}\right)^{(t+1)} \propto e^{\left\langle \ln p\left(\boldsymbol{X},\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda}\middle|\boldsymbol{A^t},\boldsymbol{B^t},\boldsymbol{A^\lambda},\boldsymbol{B^\lambda},\vec{z}\right)\right\rangle_{\frac{q(\boldsymbol{S},\boldsymbol{T},\boldsymbol{V},\boldsymbol{\Lambda})^{(t)}}{q(\lambda_{il})^{(t)}}}}$$
$$= \mathcal{G}\left(v_{il}\left|\alpha_{il}^{\lambda(t)}, \beta_{il}^{\lambda(t)}\right.\right)$$

$$\alpha_{il}^{\lambda(t)} = a_{il}^{\lambda} + \sum_{l} \delta(z_\tau - l)$$
$$\beta_{il}^{\lambda(t)} = \left(b_{il}^{\lambda -1} + \sum_{\tau} \left\langle v_{i\tau}\right\rangle^{(t)} \delta(z_\tau - l)\right)^{-1}$$

$$\left\langle \lambda_{il}\right\rangle^{(t+1)} = \alpha_{il}^{\lambda(t)} \beta_{il}^{\lambda(t)}$$
$$\left\langle \ln \lambda_{il}\right\rangle^{(t+1)} = \Psi\left(\alpha_{il}^{\lambda(t)}\right) + \ln \beta_{il}^{\lambda(t)}$$

The algorithm can be rewritten in matrix form: using matrix notation from Table A1. and Table A2., the learning algorithm is summarized in Table A3., where by $.*$ and $./$ elementwise matrix product and elementwise matrix division are denoted, respectively, and by $\mathbf{1}$ matrix of ones of appropriate dimensions.
Lower bound can be derived from (10) and used as convergence indicator and also as a basis for model comparison.

$[\boldsymbol{X}]_{\nu\tau} = x_{\nu\tau}$	$\left[\boldsymbol{A}^{\boldsymbol{t}}\right]_{\nu i} = a^t_{\nu i}$	$\left[\boldsymbol{A}^{\boldsymbol{\lambda}}\right]_{il} = a^{\lambda}_{il}$
$[\boldsymbol{\Delta}]_{\tau l} = \delta\left(z_\tau - l\right)$	$\left[\boldsymbol{B}^{\boldsymbol{t}}\right]_{\nu i} = b^t_{\nu i}$	$\left[\boldsymbol{B}^{\boldsymbol{\lambda}}\right]_{il} = b^{\lambda}_{il}$

Table A1. Observed variables and hyperparameters of supervised vbNMF model

$\left[\boldsymbol{E}^{(t)}_{\boldsymbol{t}}\right]_{\nu i} = \langle t_{\nu i}\rangle^{(t)}$	$\left[\boldsymbol{E}^{(t)}_{\boldsymbol{v}}\right]_{i\tau} = \langle v_{i\tau}\rangle^{(t)}$
$\left[\boldsymbol{L}^{(t)}_{\boldsymbol{t}}\right]_{\nu i} = \langle \ln t_{\nu i}\rangle^{(t)}$	$\left[\boldsymbol{L}^{(t)}_{\boldsymbol{v}}\right]_{i\tau} = \langle \ln v_{i\tau}\rangle^{(t)}$
$\left[\boldsymbol{\Sigma}^{(t)}_{\boldsymbol{t}}\right]_{\nu i} = \sum_{\tau} \langle s_{\nu i\tau}\rangle^{(t)}$	$\left[\boldsymbol{E}^{(t)}_{\boldsymbol{\lambda}}\right]_{il} = \langle \lambda_{il}\rangle^{(t)}$
$\left[\boldsymbol{\Sigma}^{(t)}_{\boldsymbol{v}}\right]_{i\tau} = \sum_{\nu} \langle s_{\nu i\tau}\rangle^{(t)}$	$\left[\boldsymbol{L}^{(t)}_{\boldsymbol{\lambda}}\right]_{il} = \langle \ln \lambda_{il}\rangle^{(t)}$

Table A2. Variational parameters of supervised vbNMF model

Inputs:

$$\boldsymbol{X}, \boldsymbol{A^t}, \boldsymbol{B^t}, \boldsymbol{A^\lambda}, \boldsymbol{B^\lambda}, \boldsymbol{\Delta}$$

Initialize:

$$\boldsymbol{E_t^{(0)}}, \boldsymbol{L_t^{(0)}}, \boldsymbol{E_v^{(0)}}, \boldsymbol{L_v^{(0)}}, \boldsymbol{\Sigma_t^{(0)}}, \boldsymbol{\Sigma_v^{(0)}}, \boldsymbol{E_\lambda^{(0)}}, \boldsymbol{L_\lambda^{(0)}}$$

$$t = 0$$

Loop:

$$\boldsymbol{\Xi} = \boldsymbol{X} . \Big/ \left(\left(\exp \boldsymbol{L_t^{(t)}} \right) * \left(\exp \boldsymbol{L_v^{(t)}} \right) \right)$$

$$\boldsymbol{\Sigma_v^{(t+1)}} = \exp \boldsymbol{L_v^{(t)}} . * \left(\left(\exp \boldsymbol{L_t^{(t)}} \right)^T * \boldsymbol{\Xi} \right)$$

$$\boldsymbol{\Sigma_t^{(t+1)}} = \exp \boldsymbol{L_t^{(t)}} . * \left(\boldsymbol{\Xi} * \left(\exp \boldsymbol{L_v^{(t)}} \right)^T \right)$$

$$\boldsymbol{A_t} = \boldsymbol{A^t} + \boldsymbol{\Sigma_t^{(t+1)}}$$

$$\boldsymbol{B_t} = 1. \Big/ \left(1. \Big/ \boldsymbol{B^t} + \boldsymbol{1} * \left(\boldsymbol{E_v^{(t)}} \right)^T \right)$$

$$\boldsymbol{E_t^{(t+1)}} = \boldsymbol{A_t} . * \boldsymbol{B_t}$$

$$\boldsymbol{L_t^{(t+1)}} = \Psi \left(\boldsymbol{A_t} \right) + \ln \boldsymbol{B_t}$$

$$\boldsymbol{A_v} = \boldsymbol{1} + \boldsymbol{\Sigma_v^{(t+1)}}$$

$$\boldsymbol{B_v} = 1. \Big/ \left(\boldsymbol{E_\lambda^{(t)}} * \boldsymbol{\Delta} + \left(\boldsymbol{E_t^{(t)}} \right)^T * \boldsymbol{1} \right)$$

$$\boldsymbol{E_v^{(t+1)}} = \boldsymbol{A_v} . * \boldsymbol{B_v}$$

$$\boldsymbol{L_v^{(t+1)}} = \Psi \left(\boldsymbol{A_v} \right) + \ln \boldsymbol{B_v}$$

$$\boldsymbol{A_\lambda} = \boldsymbol{A^\lambda} + \boldsymbol{\Delta} * \boldsymbol{1}$$

$$\boldsymbol{B_\lambda} = 1. \Big/ \left(1. \Big/ \boldsymbol{B^\lambda} + \boldsymbol{E_v^{(t+1)}} . * \boldsymbol{\Delta} \right)$$

$$\boldsymbol{E_\lambda^{(t+1)}} = \boldsymbol{A_\lambda} . * \boldsymbol{B_\lambda}$$

$$\boldsymbol{L_\lambda^{(t+1)}} = \Psi \left(\boldsymbol{A_\lambda} \right) + \ln \boldsymbol{B_\lambda}$$

Hyperparameter optimization (non-Bayesian)

End loop

Table A3. The learning algorithm in matrix form