Preprint submitted to Hindawi Publishing Corporation.

Non-negative Matrix Factorization with Gaussian Process Priors

Mikkel N. Schmidt
Technical University of Denmark
Informatics and Mathematical Modelling
Richard Petersens Plads, Building 321
2800 Kgs. Lyngby, Denmark
mns@imm.dtu.dk

Hans Laurberg
Aalborg University
Department of Electronic Systems
Niels Jernes Vej 12
9220 Aalborg Ø., Denmark
hla@es.aau.dk

January 16, 2008

Abstract

We present a general method for including prior knowledge in a non-negative matrix factorization (NMF), based on Gaussian process priors. We assume, that the non-negative factors in the NMF are linked by a strictly increasing function to an underlying Gaussian process, specified by its covariance function. This allows us to find NMF decompositions, that agree with our prior knowledge of the distribution of the factors, such as sparseness, smoothness, and symmetries. The method is demonstrated with an example from chemical shift brain imaging.

1 Introduction

Non-negative matrix factorization (NMF) [1, 2] is a recent method for factorizing a matrix as the product of two matrices, in which all elements are non-negative. NMF has found widespread application in many different areas including pattern recognition [3], clustering [4], dimensionality reduction [5], and spectral analysis [6, 7]. Many physical signals, such as pixel intensities, amplitude spectra, and occurence counts, are naturally represented by non-negative numbers. In the analysis of mixtures of such data, non-negativity of the individual components is a reasonable constraint. Recently, a very simple algorithm [8] for computing the NMF was introduced. This has initiated much research aimed at developing more robust and efficient algorithms.

Efforts have been made to enhance the quality of the NMF by adding further constraints to the decomposition, such as sparsity [9], spatial localization

[10, 11], and smoothness [11, 12], or by extending the model to be convolutive [13, 14]. Many extended NMF methods are derived by adding appropriate constraints and penalty terms to a cost function. Alternatively, NMF methods can be derived in a probabilistic setting, based on the distribution of the data [15, 16, 6, 17]. These approaches have the advantage that the underlying assumptions in the model are made explicit.

In this paper we present a general method for using prior knowledge to improve the quality of the solutions in NMF. The method is derived in a probabilistic setting, and it is based on defining prior probability distributions of the factors in the NMF model in a Gaussian process framework. We assume that the non-negative factors in the NMF are linked by a strictly increasing function to an underlying Gaussian process, specified by its covariance function. By specifying the covariance of the underlying process, we can compute NMF decompositions that agree with our prior knowledge of the factors, such as sparseness, smoothness, and symmetries. We refer to the proposed method as non-negative matrix factorization with Gaussian process priors, or GPP-NMF for short.

2 NMF with Gaussian Process Priors

In the following we derive a method for including prior information in an NMF decomposition by assuming Gaussian process priors (for a general introduction to Gaussian processes, see e.g. Rasmussen and Williams [18].) In our approach, the Gaussian process priors are linked to the non-negative factors in the NMF by a suitable link function. To set up the notation, we start by deriving the standard NMF method as a maximum likelihood (ML) estimator and then move on to the maximum a posteriori (MAP) estimator. Then we discuss Gaussian process priors and introduce a change of variable that gives better optimization properties. Finally, we discuss the selection of the link function.

2.1 Maximum Likelihood NMF

The NMF problem can be stated as

$$\boldsymbol{X} = \boldsymbol{D}\boldsymbol{H} + \boldsymbol{N}, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{K\times L}$ is a data matrix that is factorized as the product of two element-wise non-negative matrices, $\boldsymbol{D} \in \mathbb{R}_+^{K\times M}$ and $\boldsymbol{H} \in \mathbb{R}_+^{M\times L}$, where $\mathbb{R}_+$ denotes the non-negative reals. The matrix $\boldsymbol{N} \in \mathbb{R}^{K\times L}$ is the residual noise.

There exists a number of different algorithms [8, 19, 20, 21, 16, 15, 17] for computing this factorization, some of which can be viewed as maximum likelihood methods under certain assumptions about the distribution of the data. For example, least squares NMF corresponds to i.i.d. Gaussian noise [17] and Kullback-Leibler NMF corresponds to a Poisson process [6].

The ML estimate of $\boldsymbol{D}$ and $\boldsymbol{H}$ is given by

$$\{\boldsymbol{D}_{\mathrm{ML}}, \boldsymbol{H}_{\mathrm{ML}}\} = \underset{\boldsymbol{D},\boldsymbol{H}\geq 0}{\arg\min}\, \mathcal{L}_{X|D,H}(\boldsymbol{D}, \boldsymbol{H}), \tag{2}$$

where $\mathcal{L}_{X|D,H}(\boldsymbol{D}, \boldsymbol{H})$ is the negative log likelihood of the factors.

Example 1 (Least squares NMF). *An example of a maximum likelihood NMF is the least squares estimate. If the noise is i.i.d. Gaussian with variance* σ_N^2*, the likelihood of the factors* $\boldsymbol{D}$ *and* $\boldsymbol{H}$ *can be written as*

$$p^{\mathrm{LS}}_{X|D,H}(\boldsymbol{X}|\boldsymbol{D},\boldsymbol{H}) = \frac{1}{\left(\sqrt{2\pi}\sigma_N\right)^{KL}} \exp\left(-\frac{||\boldsymbol{X}-\boldsymbol{D}\boldsymbol{H}||_F^2}{2\sigma_N^2}\right). \tag{3}$$

The negative log likelihood, which serves as a cost function for optimization, is then

$$\mathcal{L}^{\mathrm{LS}}_{X|D,H}(\boldsymbol{D},\boldsymbol{H}) \propto \frac{1}{2\sigma_N^2}||\boldsymbol{X}-\boldsymbol{D}\boldsymbol{H}||_F^2, \tag{4}$$

where we use the proportionality symbol to denote equality subject to an additive constant. To compute a maximum likelihood estimate of $\boldsymbol{D}$ *and* $\boldsymbol{H}$*, the gradient of the negative log likelihood is useful*

$$\nabla_H \mathcal{L}^{\mathrm{LS}}_{X|D,H}(\boldsymbol{D},\boldsymbol{H}) = \frac{1}{\sigma_N^2}\boldsymbol{D}^\top(\boldsymbol{D}\boldsymbol{H}-\boldsymbol{X}), \tag{5}$$

and the gradient with respect to $\boldsymbol{D}$*, which is easy to derive, is similar because of the symmetry of the NMF problem.* □

The ML estimate can be computed by multiplicative update rules based on the gradient [8], projected gradient descent [19], alternating least squares [20], Newton-type methods [21], or any other appropriate constrained optimization method.

2.2 Maximum a Posteriori NMF

In this paper, we propose a method to build prior knowledge into the solution of the NMF problem. We choose a prior distribution $\mathrm{p}_{D,H}(\boldsymbol{D},\boldsymbol{H})$ over the factors in the model, that captures our prior beliefs and uncertainties of the solution we seek. We then compute the maximum a posteriori (MAP) estimate of the factors. Using Bayes rule, the posterior is given by

$$\mathrm{p}_{D,H|X}(\boldsymbol{D},\boldsymbol{H}|\boldsymbol{X}) = \frac{\mathrm{p}_{X|D,H}(\boldsymbol{X}|\boldsymbol{D},\boldsymbol{H})\mathrm{p}_{D,H}(\boldsymbol{D},\boldsymbol{H})}{\mathrm{p}_X(\boldsymbol{X})}. \tag{6}$$

Since the numerator is constant, the negative log posterior is the sum of a likelihood term that penalizes model misfit, and a prior term that penalizes solutions that are unlikely under the prior

$$\mathcal{L}_{D,H|X}(\boldsymbol{D},\boldsymbol{H}) \propto \mathcal{L}_{X|D,H}(\boldsymbol{D},\boldsymbol{H}) + \mathcal{L}_{D,H}(\boldsymbol{D},\boldsymbol{H}). \tag{7}$$

The MAP estimate of $\boldsymbol{D}$ and $\boldsymbol{H}$ is

$$\{\boldsymbol{D}_{\mathrm{MAP}},\boldsymbol{H}_{\mathrm{MAP}}\} = \underset{\boldsymbol{D},\boldsymbol{H}\geq 0}{\arg\min}\, \mathcal{L}_{D,H|X}(\boldsymbol{D},\boldsymbol{H}), \tag{8}$$

and it can again be computed using any appropriate optimization algorithm.

Example 2 (Non-negative sparse coding). *An example of a MAP NMF is non-negative sparse coding (NNSC) [9, 22], where the prior on* $\boldsymbol{H}$ *is i.i.d.*

exponential, and the prior on $\boldsymbol{D}$ *is flat with each column constrained to have unit norm*

$$p_{D,H}^{\mathrm{NNSC}}(\boldsymbol{D},\boldsymbol{H}) = \prod_{i,j} \lambda \exp\left(-\lambda \boldsymbol{H}_{i,j}\right), \quad ||\boldsymbol{D}_k|| = 1 \ \forall k, \tag{9}$$

where $||\boldsymbol{D}_k||$ *is the Euclidean norm of the k'th column of* $\boldsymbol{D}$*. This corresponds to the following penalty term in the cost function*

$$\mathcal{L}_{D,H}^{\mathrm{NNSC}}(\boldsymbol{D},\boldsymbol{H}) \propto \lambda \sum_{i,j} \boldsymbol{H}_{i,j}. \tag{10}$$

The gradient of the negative log prior with respect to $\boldsymbol{H}$ *is then*

$$\nabla_H \mathcal{L}_{D,H}^{\mathrm{NNSC}} = \lambda, \tag{11}$$

and the gradient with respect to $\boldsymbol{D}$ *is zero, with the further normalization constraint given in Equation (9).* □

2.3 Gaussian Process Priors

In the following, we derive the MAP estimate under the assumption that the non-negative matrices $\boldsymbol{D}$ and $\boldsymbol{H}$ are independently determined by a Gaussian process [18] connected by a link function. The Gaussian process framework provides a principled and practical approach to the specification of the prior probability distribution of the factors in the NMF model. The prior is specified in terms of two functions: i) a covariance function that describes corellations in the factors and ii) a link function, that transforms the Gaussian process prior into a desired distribution over the non-negative reals.

We assume that $\boldsymbol{D}$ and $\boldsymbol{H}$ are independent, so that we may write

$$\mathcal{L}_{D,H}(\boldsymbol{D},\boldsymbol{H}) = \mathcal{L}_D(\boldsymbol{D}) + \mathcal{L}_H(\boldsymbol{H}). \tag{12}$$

In the following, we consider only the prior for $\boldsymbol{H}$, since the treatment of $\boldsymbol{D}$ is equivalent due to the symmetry of the NMF problem. We assume that there is an underlying variable vector, $\boldsymbol{h} \in \mathbb{R}^{LM}$, which is zero mean multivariate Gaussian with covariance matrix $\boldsymbol{\Sigma}_h$

$$\mathrm{p}_h(\boldsymbol{h}) = \left(2\pi|\boldsymbol{\Sigma}_h|^2\right)^{-\frac{1}{2}NL} \exp\left(-\frac{1}{2}\boldsymbol{h}^\top \boldsymbol{\Sigma}_h^{-1}\boldsymbol{h}\right), \tag{13}$$

and linked to $\boldsymbol{H}$ via a link function, f_h: $\mathbb{R}_+ \to \mathbb{R}$

$$\boldsymbol{h} = f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big), \tag{14}$$

which operates element-wise on its input. The $\mathrm{vec}\,(\cdot)$ function in the expression stacks its matrix operand column by column. The link function should be strictly increasing, which ensures that the inverse exists. Later, we will further assume that the derivatives of f_h and f_h^{-1} exist. Under these assumptions, the prior over $\boldsymbol{H}$ is given by (using the change of variables theorem)

$$\mathrm{p}_H(\boldsymbol{H}) = \mathrm{p}_h\Big(f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big)\Big) \left|\mathcal{J}\Big(f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big)\Big)\right| \tag{15}$$

$$\propto \exp\left(-\frac{1}{2} f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big)^\top \boldsymbol{\Sigma}_h^{-1} f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big)\right) \prod_i \left|f_h'\big(\mathrm{vec}\,(\boldsymbol{H})\big)\right|_i, \tag{16}$$

where $\mathcal{J}(\cdot)$ denotes the Jacobian determinant and f_h' is the derivative of the link function. The negative log prior is then

$$\mathcal{L}_H(\boldsymbol{H}) \propto \frac{1}{2} f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big)^\top \boldsymbol{\Sigma}_h^{-1} f_h\big(\mathrm{vec}\,(\boldsymbol{H})\big) - \sum_i \log\big|f_h'\big(\mathrm{vec}\,(\boldsymbol{H})\big)\big|_i. \quad (17)$$

This expression can be combined with an appropriate likelihood function, such as the least squares likelihood in Equation (4), and be optimized to yield the MAP solution; however, in our experiments, we found that a more simple and robust algorithm can be obtained by making a change of variable as explained next.

2.4 Change of Optimization Variable

Instead of optimizing over the non-negative factors $\boldsymbol{D}$ and $\boldsymbol{H}$, we introduce the variables $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$, which are related to $\boldsymbol{D}$ and $\boldsymbol{H}$ by

$$\boldsymbol{D} = g_d(\boldsymbol{\delta}) = \mathrm{vec}^{-1}\left(f_d^{-1}(\boldsymbol{C}_d^\top \boldsymbol{\delta})\right), \quad \boldsymbol{H} = g_h(\boldsymbol{\eta}) = \mathrm{vec}^{-1}\left(f_h^{-1}(\boldsymbol{C}_h^\top \boldsymbol{\eta})\right), \quad (18)$$

where the $\mathrm{vec}^{-1}(\cdot)$ function maps its vector input into a matrix of appropriate size. The matrices $\boldsymbol{C}_d$ and $\boldsymbol{C}_h$ are the matrix square roots (Cholesky decompositions) of the covariance matrices $\boldsymbol{\Sigma}_d$ and $\boldsymbol{\Sigma}_h$, such that $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ are standard i.i.d. Gaussian.

This change of variable serves two purposes. First of all, we found that optimizing over the transformed variables was faster, more robust, and less prone to getting stuck in local minima. Second, the transformed variables are not constrained to be non-negative, which allows us to use existing unconstrained optimization methods to compute their MAP estimate.

The prior distribution of the transformed variable $\boldsymbol{\eta}$ is

$$\mathrm{p}_\eta(\boldsymbol{\eta}) = \mathrm{p}_H\big(g_h(\boldsymbol{\eta})\big)\ \big|\mathcal{J}\big(g_h(\boldsymbol{\eta})\big)\big| = \frac{1}{(2\pi)^{\frac{LM}{2}}} \exp\left(-\frac{1}{2}\boldsymbol{\eta}^\top\boldsymbol{\eta}\right), \quad (19)$$

and the negative log prior is

$$\mathcal{L}_\eta(\boldsymbol{\eta}) \propto \frac{1}{2}\boldsymbol{\eta}^\top\boldsymbol{\eta}. \quad (20)$$

To compute the MAP estimate of the transformed variables, we must combine this expression for the prior (and a similar expression for the prior of $\boldsymbol{\delta}$) with a likelihood function, in which the same change of variable is made

$$\mathcal{L}_{\delta,\eta|X}(\boldsymbol{\delta},\boldsymbol{\eta}) = \mathcal{L}_{X|D,H}\big(g_d(\boldsymbol{\delta}), g_h(\boldsymbol{\eta})\big) + \frac{1}{2}\boldsymbol{\delta}^\top\boldsymbol{\delta} + \frac{1}{2}\boldsymbol{\eta}^\top\boldsymbol{\eta}. \quad (21)$$

Then the MAP solution can be found by optimizing over $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$

$$\{\boldsymbol{\delta}_{\mathrm{MAP}}, \boldsymbol{\eta}_{\mathrm{MAP}}\} = \arg\min_{\boldsymbol{\delta},\boldsymbol{\eta}} \mathcal{L}_{\delta,\eta|X}(\boldsymbol{\delta},\boldsymbol{\eta}), \quad (22)$$

and, finally, estimates of $\boldsymbol{D}$ and $\boldsymbol{H}$ can be computed using Equation (18).

Example 3 (Least squares non-negative matrix factorization with Gaussian process priors (GPP-NMF)). *If we use the least squares likelihood in Equation (4), the posterior distribution in Equation (21) is given by*

$$\mathcal{L}^{LS\text{-}GPP}_{\delta,\eta|X}(\boldsymbol{\delta},\boldsymbol{\eta}) = \frac{1}{2}\left(\sigma_N^{-2}||\boldsymbol{X} - g_d(\boldsymbol{\delta})g_h(\boldsymbol{\eta})||_F^2 + \boldsymbol{\delta}^\top\boldsymbol{\delta} + \boldsymbol{\eta}^\top\boldsymbol{\eta}\right) \tag{23}$$

The MAP estimate of $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ is found by minimizing this expression, for which the derivative is useful

$$\nabla_\eta \mathcal{L}^{LS\text{-}GPP}_{\delta,\eta|X}(\boldsymbol{\delta},\boldsymbol{\eta}) = \sigma_N^{-2}\Big(\mathrm{vec}\left(g_d(\boldsymbol{\delta})^\top(g_d(\boldsymbol{\delta})g_h(\boldsymbol{\eta}) - \boldsymbol{X})\right) \odot (f_h^{-1})'(\boldsymbol{C}_h^\top\boldsymbol{\eta})\Big)^\top \boldsymbol{C}_h^\top + \boldsymbol{\eta}^\top, \tag{24}$$

where $\odot$ denotes the Hadamard (element-wise) product. The derivative with respect to $\boldsymbol{\delta}$ is similar due to the symmetry of the NMF problem. □

2.5 Link Function

Any strictly increasing link function that maps the non-negative reals to the real line can be used in the proposed framework; however, in order to have a better probabilistic interpretation of the prior distribution, we propose a simple principle for choosing the link function. We choose the link function such that f_h^{-1} maps the marginal distribution of the elements of the underlying Gaussian process vector $\boldsymbol{h}$ into a specifically chosen marginal distribution of the elements of $\boldsymbol{H}$. Such an inverse function can be found as $f_h^{-1}(\boldsymbol{h}_i) = \mathrm{P}_H^{-1}\big(\mathrm{P}_h(\boldsymbol{h}_i)\big)$ where $\mathrm{P}(\cdot)$ denotes the marginal cumulative distribution functions (cdf).

Since the marginals of a Gaussian process are Gaussian, $\mathrm{P}_h(\boldsymbol{h}_i)$ is the Gaussian cdf, and, using Equation (13), the inverse link function is given by

$$f_h^{-1}(\boldsymbol{h}_i) = \mathrm{P}_H^{-1}\left(\frac{1}{2} + \frac{1}{2}\Phi\left(\frac{\boldsymbol{h}_i}{\sqrt{2}\sigma_i}\right)\right) \tag{25}$$

where σ_i^2 is the i'th diagonal element of $\boldsymbol{\Sigma}_h$ and $\Phi(\cdot)$ is the error function.

Example 4 (Exponential-to-Gaussian link function). *If we choose to have exponential marginals in $\boldsymbol{H}$, as in NNSC described in Example 2, we select P_H as the exponential cdf. The inverse link function is then*

$$f_h^{-1}(\boldsymbol{h}_i) = -\frac{1}{\lambda}\log\left(\frac{1}{2} - \frac{1}{2}\Phi\left(\frac{\boldsymbol{h}_i}{\sqrt{2}\sigma_i}\right)\right), \tag{26}$$

where λ is an inverse scale parameter. The derivative of the inverse link function, which is needed for the parameter estimation, is given by

$$(f_h^{-1})'(\boldsymbol{h}_i) = \frac{1}{\sqrt{2\pi}\sigma_i\lambda}\exp\left(\lambda f_h^{-1}(\boldsymbol{h}_i) - \frac{\boldsymbol{h}_i^2}{2\sigma_i^2}\right). \tag{27}$$

□

Example 5 (Rectified-Gaussian-to-Gaussian link function). *Another interesting non-negative distribution is the rectified Gaussian given by*

$$p(x) = \begin{cases} \frac{2}{\sqrt{2\pi}s}\exp\left(-\frac{x^2}{2s^2}\right) & , \quad x \geq 0 \\ 0 & , \quad x < 0 \end{cases} \tag{28}$$

Using this pdf in Equation (25), the inverse link function is

$$f_h^{-1}(\boldsymbol{h}_i) = \sqrt{2}s\Phi^{-1}\left(\frac{1}{2} + \frac{1}{2}\Phi\left(\frac{\boldsymbol{h}_i}{\sqrt{2}\sigma_i}\right)\right), \tag{29}$$

and the derivative of the inverse link function is

$$(f_h^{-1})'(\boldsymbol{h}_i) = \frac{s}{2\sigma_i}\exp\left(\frac{f_h^{-1}(\boldsymbol{h}_i)^2}{2s^2} - \frac{\boldsymbol{h}_i^2}{2\sigma_i^2}\right). \tag{30}$$

□

2.6 Summary of the GPP-NMF Method

The GPP-NMF method can be summarized in the following steps.

1. Choose a suitable negative log likelihood function $\mathcal{L}_{X|D,H}(\boldsymbol{D}, \boldsymbol{H})$ based on knowledge of the distribution of the data or the residual.
2. For each of the non-negative factors $\boldsymbol{D}$ and $\boldsymbol{H}$, choose suitable link and covariance functions according to your prior beliefs. If necessary, draw samples from the prior distribution to examine its properties.
3. Compute the MAP estimate of $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ by Equation (22) using any suitable unconstrained optimization algorithm.
4. Compute $\boldsymbol{D}$ and $\boldsymbol{H}$ using Equation (18).

Our Matlab implemention of the GPP-NMF method is available online [23].

3 Experimental Results

We will demonstrate the proposed method on two examples, first a toy example, and second an example taken from the chemical shift brain imaging literature.

In our experiments we use the least squares GPP-NMF described in Example 3 and the link functions described in Example 4–5. The specific optimization method used to compute the GPP-NMF MAP estimate is not the topic of this paper, and any unconstrained optimization algorithm could in principle be used. In our experiments we used a simple gradient descent with line search to perform a total of 1000 alternating updates of $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$, after which the algorithm had converged. For details of the implementation, see the accompanying Matlab code [23].

3.1 Toy Example

We generated a 100×200 data matrix, $\boldsymbol{Y}$, by taking a random sample from the GPP-NMF model with two factors. We chose the generating covariance function for both $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ as a Gaussian radial basis function (RBF),

$$\phi(i,j) = \exp\left(-\frac{(i-j)^2}{\beta^2}\right), \tag{31}$$

where i and j are two sample indices, and the length scale parameter, which determines the smoothness of the factors, was $\beta^2 = 100$. We set the covariance between the two factors to zero, such that the factors were uncorrelated. For the matrix $\boldsymbol{D}$ we used the rectified-Gaussian-to-Gaussian link function with $s = 1$, and for $\boldsymbol{H}$ we used the exponential-to-Gaussian link function with $\lambda = 1$. Finally, we added independent Gaussian noise with variance $\sigma_N^2 = 25$, which resulted in a signal-to-noise ratio of approximately -7 dB. The generated data matrix is shown in Figure 1.

We then decomposed the generated data matrix using four different methods:

1. **LS-NMF:** Standard least squares NMF [8]. This algorithm does not allow negative data points, so these were set to zero in the experiment.

2. **CNMF:** Constrained NMF [6, 7], which is a least squares NMF algorithm that allows negative observations.

3. **GPP-NMF: Correct prior:** The proposed method with correct link-functions, covariance matrix, and parameter values.

4. **GPP-NMF: Incorrect prior:** To illustrate the sensitivity of the method to prior assumptions, we evaluated the proposed method with incorrect prior assumptions: We switched the link functions, such that for $\boldsymbol{D}$ we used the exponential-to-Gaussian, and for $\boldsymbol{H}$ we used the rectified-Gaussian-to-Gaussian. We used an RBF covariance function with $\beta^2 = 10$ and $\beta^2 = 1000$ for $\boldsymbol{D}$ and $\boldsymbol{H}$ respectively.

The results of the decompositions are shown as reconstructed data matrices in Figure 1. All four methods find solutions that visually appear to fit the underlying data. Both LS-NMF and CNMF find non-smooth solutions, whereas the two GPP-NMF results are smooth in accordance with the priors. In the GPP-NMF with incorrect prior, the dark areas (high pixel intensities) appear too wide in the first axis direction and too narrow in the section axis direction, due to the incorrect setting of the covariance function. The GPP-NMF with correct prior is visually almost equal to the true underlying data.

Plots of the estimated factors are show in Figure 2. The factors estimated by the LS-NMF and the CNMF methods appear noisy and are non-smooth, whereas the factors estimated by the GPP-NMF are smooth. The factors estimated by the LS-NMF method have a positive bias, because of the truncation of negative data. The GPP-NMF with incorrect prior has too many local extrema in the $\boldsymbol{D}$ factor and too few in the $\boldsymbol{H}$ factor due to the incorrect covariance functions. There are only minor difference between the result of the GPP-NMF with the correct prior and the underlying factors.

Measures of root mean squared error (RMSE) of the four decompositions are given in Figure 3. All four methods fit the noisy data almost equally well. (Note that, due to the additive noise with variance 25, a perfect fit to the underlying factors would result in a RMSE of 5 with respect to the noisy data.) The LS-NMF fits the data worst due to the truncation of negative data points, and the CNMF fits the data best, due to overfitting. With respect to the noise free data and the underlying factors, the RMSE is worst for the LS-NMF and best for the GPP-NMF with correct prior. The GPP-NMF with incorrect prior is better than both LS-NMF and CNMF in this case. This shows, that in this situation it

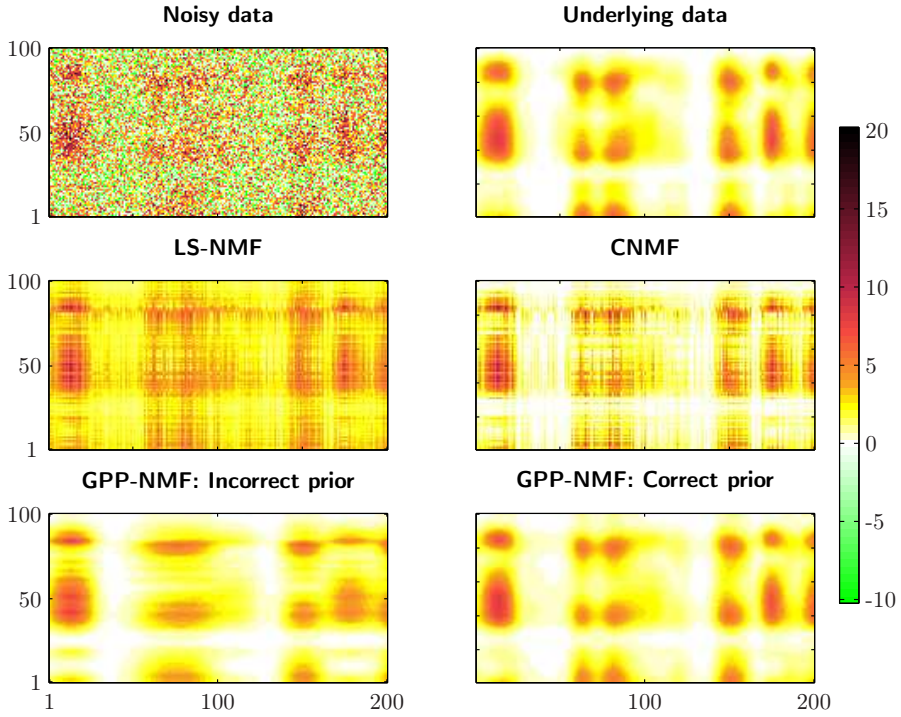

Figure 1: Toy example data matrix (upper left), underlying noise-free non-negative data (upper right), and estimates using the four methods described in the text. The data has a fairly large amount of noise and the underlying non-negative factors are smooth in both directions. The LS-NMF and CNMF decomposition are non-smooth, since these methods do not model of correlations in the factors. The GPP-NMF, which uses a smooth prior, finds a smooth solution. When using the correct prior, the soulution is very close to the true underlying data.

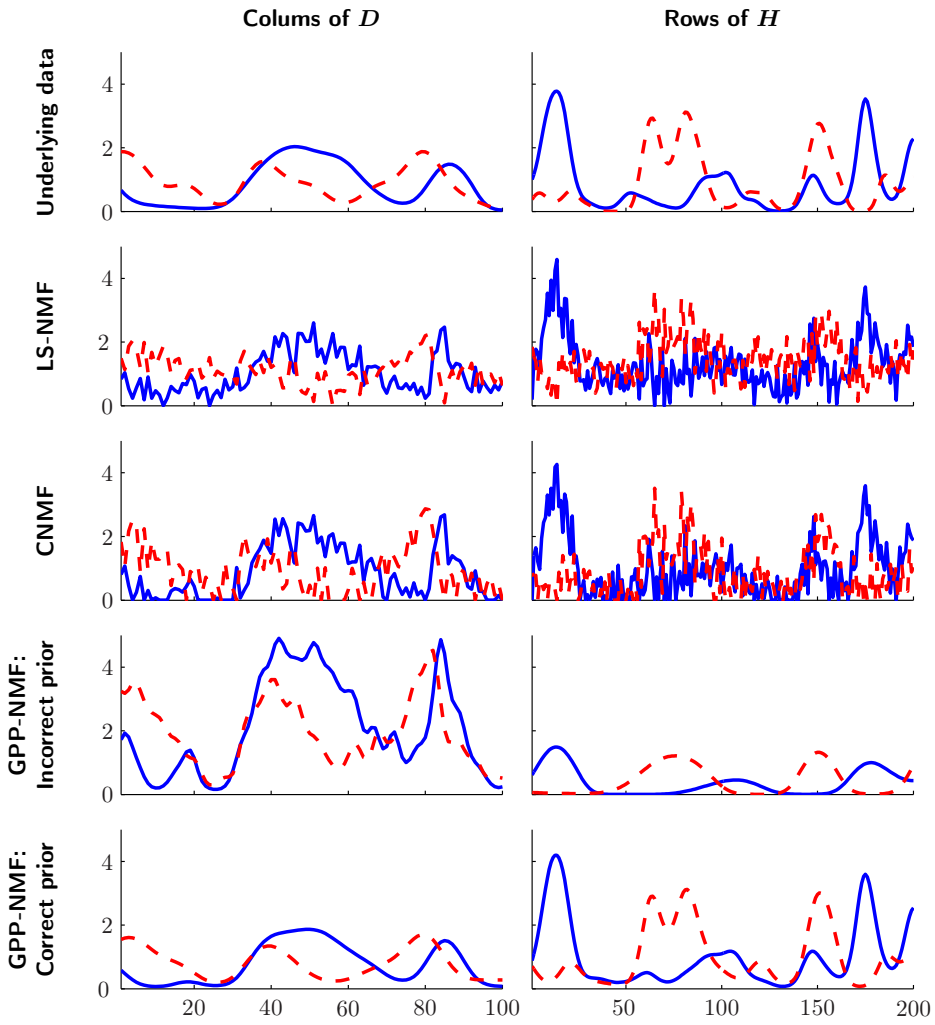

Figure 2: Underlying non-negative factors in the toy example: Columns of $\boldsymbol{D}$ (left) and rows of $\boldsymbol{H}$ (right). The factors found by the LS-NMF and the CNMF algorithm are noisy, whereas the factors found by the GPP-NMF method are smooth. When using the correct prior, the factors found are very similar to the true factors.

better to use a prior which is not perfectly correct, compared to using no prior as in the LS-NMF and CNMF methods, (which corresponds to a flat prior over the non-negative reals and no correlations.)

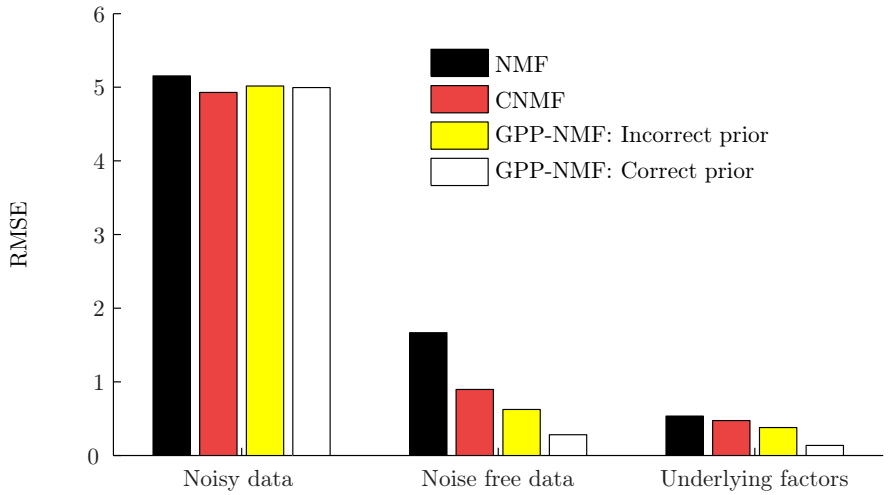

Figure 3: Toy example: Root mean squared error (RMSE) with respect to the noisy data, the underlying noise free data, and the true underlying non-negative factors. The CNMF solution fits the noisy data slightly better, but the GPP-NMF solution fits the underlying data much better.

3.2 Chemical Shift Brain Imaging Example

Next, we demonstrate the GPP-NMF method on ^{1}H decoupled ^{31}P chemical shift imaging data of the human brain. We use the data set from Ochs et al. [24], which has also been analyzed by Sajda et al. [6, 7]. The data set, which is shown in Figure 4, consists of 512 spectra measured on an $8 \times 8 \times 8$ grid in the brain.

Ochs et al. [24] use PCA to determine, that the data set is adequately described by two sources (which correspond to brain and muscle tissue.) They propose a bilinear Bayesian approach, in which they use a smooth prior over the constituent spectra, and force to zero the amplitude of the spectral shape corresponding to muscle tissue at 12 positions deep inside the head. Their approach produces physically plausible results, but it is computationally very expensive and takes several hours to compute.

Sajda et al. [6, 7] propose an NMF approach that is reported also to produce physically plausible results. Their method is several orders of magnitude faster, taking less than a second to compute. The disadvantage of the method of Sajda et al. compared to the Bayesian approach of Ochs et al. is, that it provides no mechanism for using prior knowledge to improve the solution.

The GPP-NMF approach we propose in this paper bridges the gap between the two previous approaches, in the sense that it is a relatively fast NMF approach, in which priors over the factors can be specified. These priors are

specified by the choice of the link and covariance functions. We used prior predictive sampling to find reasonable settings of the the function parameters: We drew random samples from the prior distribution and examined properties of the factors and reconstructed data. We then manually adjusted the parameters of the prior to match our prior beliefs. An example of a random draw from the prior distribution is shown in Figure 5, with the parameters set as described below.

We assumed that the factors are uncorrelated, so the covariance between factors is zero. We used a Gaussian RBF covariance function for the constituent spectra, with a length scale of $\beta = 0.3$ parts per million (ppm), and we used the exponential-to-Gaussian link function with $\lambda_d = 1$. This gave a prior for the spectra that is sparse with narrow smooth peaks. For the amplitude at the 512 voxels in the head, we used a Gaussian RBF covariance function on the 3-D voxel indices, with length scale $\beta = 2$. Furthermore, we centered the left-to-right coordinate axis in the middle of the brain, and computed the RBF kernel on the absolute value of this coordinate, so that a left-to-right symmetry was introduced in the prior distribution. Again, we used the exponential-to-Gaussian link function, and we chose $\lambda_h = 2 \cdot 10^{-4}$ to match the overall magnitude of the data. This gave a prior for the amplitude distribution that is sparse, smooth, and symmetric. The noise variance was set to $\sigma_N^2 = 10^8$ which corresponds to the noise level in the data set.

We then decomposed the data set using the proposed GPP-NMF algorithm and, for comparison, reproduced the results of Sajda et al. [7] using their CNMF method. The results of the experiments are shown in Figure 4. An example of a random draw from the prior distribution is shown in Figure 5. The results of the CNMF is shown in Figure 6, and the results of the GPP-NMF is shown in Figure 7. The figures show the constituent spectra and the fifth axial slice of the spatial distribution of the spectra. The 8×8 spatial distributions are smoothed in the illustration, similar to the way the results are visualized in the literature.

The results show that both methods give physically plausible results. The main difference is that the spatial distribution of the spectra corresponding to muscle and brain tissue is much more separated in the GPP-NMF result, which is due to the exponential, smooth, and symmetric prior distribution. By including prior information, we obtain a solution, where the factor corresponding to muscle tissure is clearly located on the edge of the skull.

4 Conclusions

We have introduced a general method for exploiting prior knowledge in non-negative matrix factorization, based on Gaussian process priors, linked to the non-negative factors by a link function. The method can be combined with any existing NMF cost function that has a probabilistic interpretation, and any existing unconstrained optimization algorithm can be used to compute the maximum a posteriori estimate.

Experiments on toy data show, that with a suitable selection of the prior distribution of the non-negative factors, the GPP-NMF method gives much better results in terms of estimating the true underlying factors, both when compared to traditional NMF and CNMF.

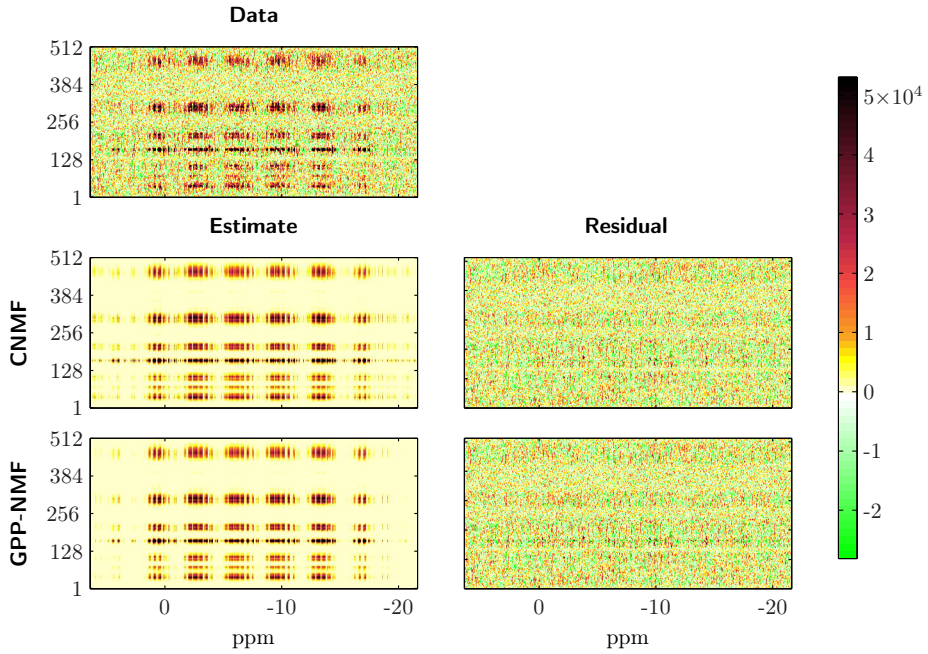

Figure 4: Brain imaging data matrix (top) along with the estimated decomposition and residual for the CNMF (middle) and GPP-NMF (bottom) method. In this view the results of the two decompositions are very similar, the data appears to be modeled equally well and the residuals are similar in magnitude.

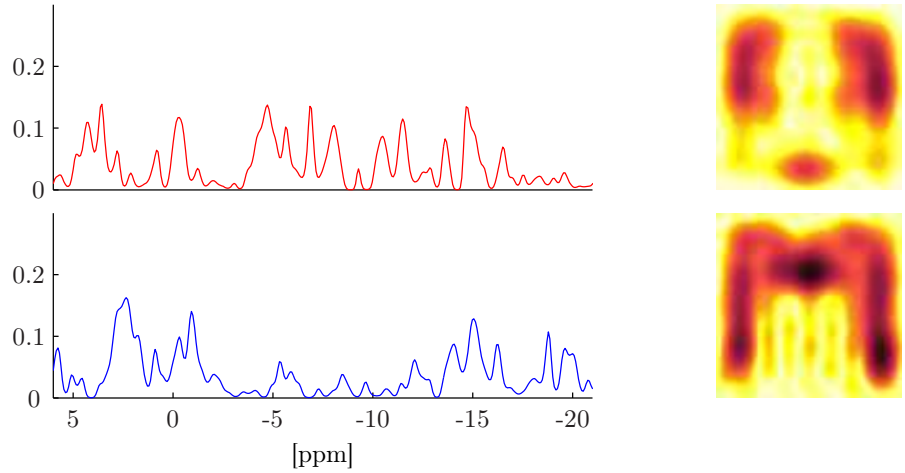

Figure 5: Brain imaging data: Random draw from the prior distribution with the parameters set as described in the text. The prior distribution of the constituent spectra (left) is exponential and smooth and the spatial distribution (right) in the brain is exponential, smooth, and has a left-to-right symmetry.

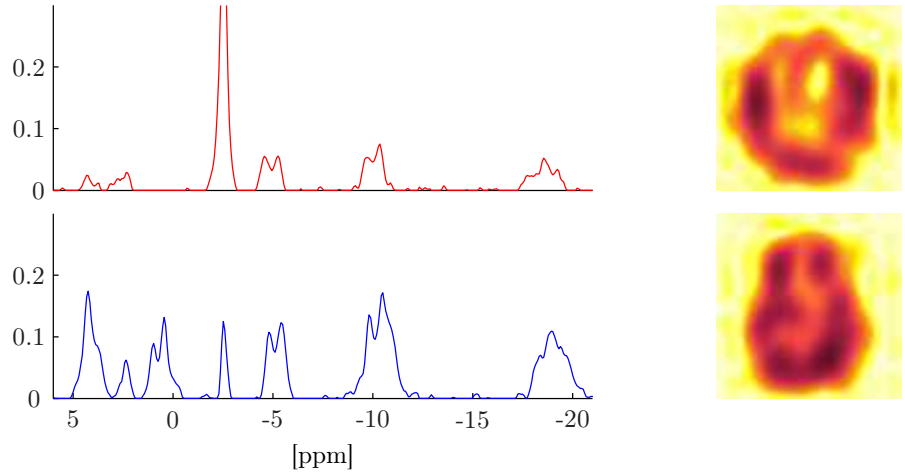

Figure 6: CNMF decomposition result. The recovered spectra are physically plausible, and the spatial distribution in the brain for the muscle (top) and brain (bottom) tissue is somewhat separated. Muscle tissue is primarily found near the edge of the skull, whereas brain tissue is primarliy found at the inside of the head.

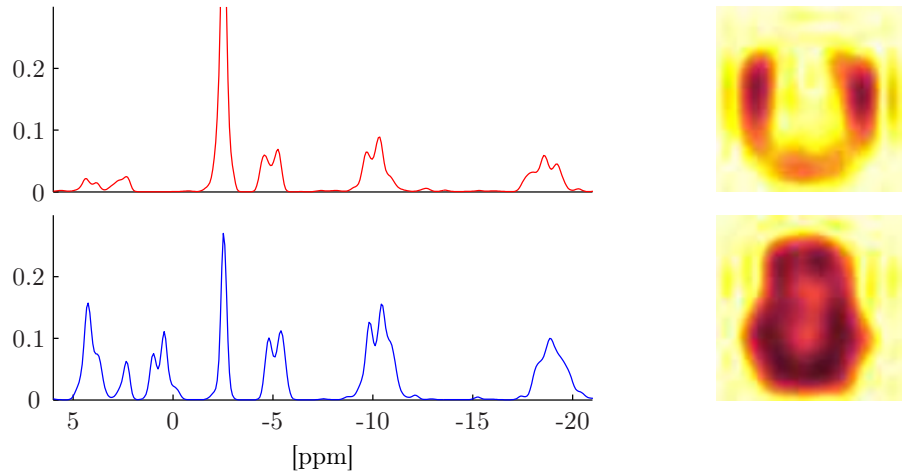

Figure 7: GPP-NMF decomposition result. The recovered spectra are very similar to the spectra found by the CNMF method, but they are slightly more smooth. The spatial distribution in the brain is highly separated between brain and muscle tissue, and it is more symmetric than the CNMF solution.

Experiments on chemical shift brain imaging data show that the GPP-NMF method can be successfully used to include prior knowledge of the spectral and spatial distribution, resulting in better spatial separation between spectra corresponding to muscle and brain tissue.

5 Acknowledgments

We would like to thank Paul Sajda and Truman Brown for making the brain imaging data available to us. This research was supported by the Intelligent Sound project, Danish Technical Research Council grant no. 26–02–0092 and partly also by the European Commission through the sixth framework IST Network of Excellence: Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL), contract no. 506778.

References

[1] P. Paatero and U. Tapper, "Positive matrix factorization: A nonnegative factor model with optimal utilization of error-estimates of data values," *Environmetrics*, vol. 5, pp. 111–126, 1994.

[2] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[3] L. Weixiang, Z. Nanning, and Y. Qubo, "Nonnegative matrix factorization and its applications in pattern recognition," *Chinese Science Bulletin*, vol. 51, no. 1, pp. 7–18, Jan 2006.

[4] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," in *Data Mining, Proceedings of SIAM International Conference on*, 2005.

[5] S. Tsuge, M. Shishibori, S. Kuroiwa, and K. Kita, "Dimensionality reduction using non-negative matrix factorization for information retrieval," in *Systems, Man, and Cybernetics, IEEE International Conference on*, vol. 2, 2001, pp. 960–965.

[6] P. Sajda, S. Du, and L. Parra, "Recovery of constituent spectra using non-negative matrix factorization." in *Wavelets: Applications in Signal and Image Processing, Proceedings of SPIE*, vol. 5207, 2003, pp. 321–331.

[7] P. Sajda, S. Du, T. R. Brown, R. Stoyanova, D. C. Shungu, X. Mao, and L. C. Parra, "Nonnegative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain," *Transactions on Medical Imaging, IEEE*, vol. 23, no. 12, pp. 1453–1465, Dec 2004.

[8] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Neural Information Processing Systems, Advances in (NIPS)*, 2000, pp. 556–562.

[9] P. Hoyer, "Non-negative sparse coding," in *Neural Networks for Signal Processing, IEEE Workshop on*, 2002, pp. 557–565.

[10] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. Cheng, "Learning spatially localized, parts-based representation," in *Computer Vision and Pattern Recognition, Proceedings of the IEEE Computer Society Conference on*, vol. 1, Dec 2001, pp. 207–212.

[11] Z. Chen and A. Cichocki, "Nonnegative matrix factorization with temporal smoothness and/or spatial decorrelation constraints," Laboratory for Advanced Brain Signal Processing, RIKEN, Tech. Rep., 2005.

[12] T. Virtanen, "Sound source separation using sparse coding with temporal continuity objective," in *International Computer Music Conference, ICMC*, 2003.

[13] P. Smaragdis, "Non-negative matrix factor deconvolution; extraction of multiple sound sourses from monophonic inputs," in *Independent Component Analysis and Blind Signal Separation, International Conference on (ICA), Lecture Notes in Computer Science*, vol. 3195, Sep 2004, pp. 494–499.

[14] M. N. Schmidt and M. Mørup, "Nonnegative matrix factor 2-d deconvolution for blind single channel source separation," in *Independent Component Analysis and Blind Signal Separation, International Conference on (ICA), Lecture Notes in Computer Science*, Apr 2006, vol. 3889, pp. 700–707.

[15] O. Winther and K. B. Petersen, "Bayesian independent component analysis: Variational methods and non-negative decompositions," *Digital Signal Processing*, vol. 17, no. 5, 2007.

[16] T. Hofmann, "Probabilistic latent semantic indexing," in *Research and Development in Information Retrieval, Proceedings of the International SIGIR Conference on*, 1999.

[17] A. Cichocki, R. Zdunek, and S. ichi Amari, "Csiszar's divergences for non-negative matrix factorization: Family of new algorithms," in *Lecture Notes in Computer Science*, vol. 3889, 2006, pp. 32–39.

[18] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[19] C.-J. Lin, "Projected gradient methods for non-negative matrix factorization," *Neural Computation*, vol. 19, pp. 2756–2779, 2007.

[20] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization."

[21] D. Kim, S. Sra, and I. S. Dhillon, "Fast newton-type methods for the least squares nonnegative matrix approximation problem," in *Data Mining, Proceedings of SIAM Conference on*, 2007.

[22] J. Eggert and E. Körner, "Sparse coding and NMF," in *Neural Networks, IEEE International Conference on*, vol. 4, 2004, pp. 2529–2533.

[23] M. N. Schmidt. (2008) Non-negative matrix factorization with gaussian process priors. [Online]. Available: http://www.mikkelschmidt.dk/cin2008

[24] M. F. Ochs, R. S. Stoyanova, F. Arias-Mendoza, and T. R. Brown, "A new method for spectral decomposition using a bilinear bayesian approach," *Journal of Magnetic Resonance*, pp. 161–176, 1999.