RECOVERY OF CONSTITUENT SPECTRA IN SIMULATED SURFACE ENHANCED RAMAN MAPS WITH NON-NEGATIVE MATRIX FACTORIZATION

David Frich Hansen, Rasmus Arpe Fogh Jensen, Søren Emil Schmidt, Peter Edsberg Møllgaard

DTU Compute

ABSTRACT

We develop and apply non-negative matrix factorization for recovery of constituent spectra in simulated Raman spectroscopy data. We set up the model $\boldsymbol{X} = \boldsymbol{DH} + \boldsymbol{N}$ where we impose non-negativity on the matrix factors, i.e. $\boldsymbol{D}, \boldsymbol{H} \geq 0$. We apply smoothing priors via Gaussian processes and show that incorporating a spatial covariance for the Gaussian process recovers spectra with higher resolution when a mixture of Raman maps is used. Finally, we show that posterior sampling with Hamiltonian Monte Carlo is feasible for this problem.

1. INTRODUCTION

Non-negative matrix factorization (NMF) is an unsupervised matrix decomposition method that decomposes a data matrix $\boldsymbol{X} \approx \boldsymbol{DH}$, $\boldsymbol{D}, \boldsymbol{H} \geq 0$. The non-negativity constraint on the matrix factors allows for part based learning compared to other unsupervised decomposition methods such as Principal Component Analysis and Vector Quantization where complex cancellations of scores and loadings greatly decreases interpretability of the learned basis - and the authors of the original paper, Lee and Seung, makes such comparisons on images of human faces [1].

Besides image reconstruction, NMF has seen applications in a wide range of fields where non-negativity is a reasonable constraint. Examples of these applications include several kinds of spectral data [2, 3, 4], clustering [5, 6] and dimensionality reduction [7, 8, 9]. In this project, we focus on recovery of Raman spectra for molecule identification.

Molecule identification is an increasingly important task for research in many fields. A method for finding the 'fingerprint' of a molecule is through the physical phenomenon of Raman scattering [10] - a rarely occuring type of photon scattering. The wavelengths of the emitted photons will depend on the molecular structure, and will therefore provide a 'fingerprint' of the molecule that emitted the photons, thus allowing for identification of the molecule - that is, we look for the location of peaks in the spectrum, as this is what characterizes the molecule - see Figure 1.

In practice data from Raman spectroscopy is very sparse due to the rarity of the phenomenon. Furthermore, it is very noisy and will also include a background component from either residual molecules on the sample plate or contamination. As seen on Figure 2, successful Raman scattering is visible as hotspots with high intensity, with the corresponding data matrix $\boldsymbol{X}$ containing the Raman spectrum for each cell in the Raman map. As such, because of the rarity of Raman scattering, the data matrix $\boldsymbol{X}$ consists largely of spectra describing the background, leading to the sparsity of the data.

This means that an important task in analysis of this kind of data is separating signal (i.e. the Raman spectra) from noise and from the background. In this project, we develop a Bayesian NMF model with smoothing priors similar to [11], and try to recover constituent spectra in simulated Raman spectroscopy data.

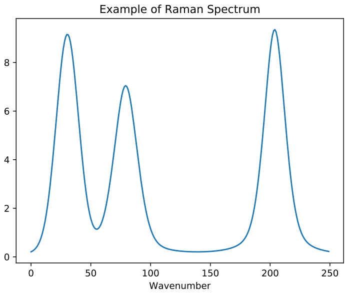


Fig. 1. Example of a simulated Raman spectrum

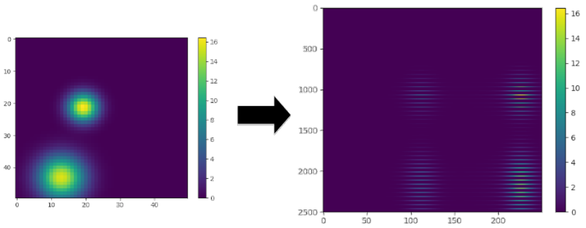


Fig. 2. Raman map (left) and data matrix $\boldsymbol{X}$ (right). The Raman map (50×50) show clear concentrations in two hotspots. The data matrix shows the 2500 measurements from the Raman map each using 250 measured wavenumbers.

2. METHODS

The general NMF problem may be stated as finding two non-negative matrices $\boldsymbol{D}$ and $\boldsymbol{H}$ such that the data matrix $\boldsymbol{X}$ decomposes,

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

where $\boldsymbol{X} \in \mathbb{R}^{K \times L}$, $\boldsymbol{D} \in \mathbb{R}_{+}^{K \times M}$, $\boldsymbol{H} \in \mathbb{R}_{+}^{M \times L}$ and $\boldsymbol{N} \in \mathbb{R}^{K \times L}$.

Here M is a hyperparameter, which can be chosen or estimated, denoting the number of components in the NMF model.

2.1. Least Squares NMF with multiplicative updates

A way to find suitable matrix factors is by maximum likelihood estimation through simple multiplicative updates of $\boldsymbol{D}$ and $\boldsymbol{H}$. We provide a description of this approach in Appendix A and denote this model the *Least Squares NMF* (LS-NMF). This model is the one originally suggested by Seung and Lee in [1].

2.2. NMF with Gaussian process priors

As discussed before, data from Raman spectroscopy is often noisy and may contain several sources that we wish to separate. As such, we consider the NMF model in a Bayesian framework that allows us to impose priors on the matrix factors based on our subjective belief that the found spectra are smooth (see Figure 1). This is exactly the formulation of Schmidt and Laurberg in [11], which we will introduce here. We denote the resulting model NMF-GPP. Schmidt and Laurberg suggest a loss based on a least squares likelihood such that we have

$$p_{\text{GPP-NMF}}(\boldsymbol{X}|\boldsymbol{D},\boldsymbol{H}) = \frac{1}{(\sqrt{2\pi}\sigma)^{KL}} \exp\left[-\frac{\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{H}\|_F^2}{2\sigma^2}\right], \tag{2}$$

where σ is the standard deviation of the noise of the data. We then introduce the negative log likelihood as a loss for parameter estimation,

$$\mathcal{L}_{\text{GPP-NMF}}(\boldsymbol{D},\boldsymbol{H}) = \frac{1}{2\sigma^2}\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{H}\|_F^2 + \text{const.} \tag{3}$$

where we will omit the constant not dependent on $\boldsymbol{D}$ and $\boldsymbol{H}$.

We introduce Gaussian process priors (GPP) on $\boldsymbol{D}$ and $\boldsymbol{H}$ as this ensures the smoothing of the spectra and the spatial correlation of the loadings which is feasible due to 1) the smoothness of the curves seen in Figure 1, and 2) the Gaussian like clustering of the concentration of molecule on the plate (see Figure 2). We handle the non-negativity constraint on the matrix factors by introducing vectors $\boldsymbol{d} \in \mathbb{R}^{KM}$ and $\boldsymbol{h} \in \mathbb{R}^{ML}$ related to $\boldsymbol{D}$ and $\boldsymbol{H}$ through non-linear link functions $f_h : \mathbb{R}_{+} \to \mathbb{R}$ operating element-wise on it's vector input such that,

$$\boldsymbol{h} = f_h(\text{Vec}(\boldsymbol{H})) \tag{4}$$

where $\text{Vec}(\cdot)$ takes a matrix and reshapes it into appropriate size. Following [11] we impose strict monotonicity on the link functions such that the inverse exists and we can recover $\boldsymbol{H}$ from the vector $\boldsymbol{h}$. Furhtermore, the link functions should be differentiable. We introduce a vector $\boldsymbol{d}$ for $\boldsymbol{D}$ in a completely analogous fashion. For examples of link functions, we also refer to [11].

Instead of imposing the GPP's directly on the matrix factors, we introduce them on the vectors $\boldsymbol{d}$ and $\boldsymbol{h}$ as these do not have the same non-negativity constraints due to the link function. This means that the densities of the vectors are,

$$p(\boldsymbol{h}) = (2\pi|\Sigma_h|^2)^{-\frac{ML}{2}} \exp\left[-\frac{\boldsymbol{h}^T\Sigma_h^{-1}\boldsymbol{h}}{2}\right] \tag{5}$$

such that $\boldsymbol{h}$ is a 0 mean multivariate Gaussian with covariance Σ_h, and similarly for $\boldsymbol{d}$. We get the negative log-prior for $\boldsymbol{H}$ from the change of variables theorem of probability theory using the Jacobian determinant, i.e.

$$\mathcal{L}_H(\boldsymbol{H}) = \frac{1}{2}\boldsymbol{h}^T\Sigma_h^{-1}\boldsymbol{h} - \underbrace{\sum_i \log|f_h'(\text{Vec}(\boldsymbol{H}))|_i}_{\text{Log of Jacobian determinant}} + \text{const.} \tag{6}$$

Here, f_h' is the derivative of the link function. A completely similar statement can be made for $\boldsymbol{D}$.

Finally we assume independence between $\boldsymbol{D}$ and $\boldsymbol{H}$ such that we arrive at the negative log posterior,

$$\begin{aligned}\mathcal{L}_{D,H|X}(\boldsymbol{D},\boldsymbol{H}) = {} & \mathcal{L}_{\text{GPP-NMF}}(\boldsymbol{D},\boldsymbol{H}) \\ & + \mathcal{L}_D(\boldsymbol{D}) + \mathcal{L}_H(\boldsymbol{H}) + \text{const.},\end{aligned} \tag{7}$$

where we absorb all constant terms into one not dependent on $\boldsymbol{D}$ or $\boldsymbol{H}$. This can be optimized directly to yield MAP estimates, after covariances for the GPP's have been specified.

2.3. Choice of covariance function on GPPs

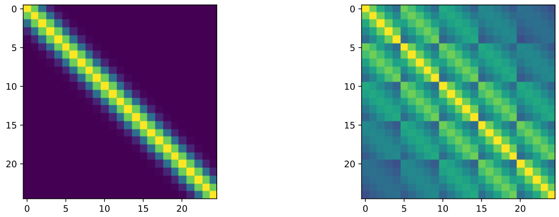
As we have prior knowledge from physicists on how a spectrum is shaped (see Figure 1), and we know that molecules tend to be clustered Gaussian like on the Raman map (Figure 2), we can incorporate this knowledge into our priors through appropriate choices of covariance functions. In this setup, the columns of $\boldsymbol{H}$ captures the spectrum. The smoothness of the spectrum can be incorporated by a spatially one-dimensional radial basis kernel, such that each measurement is correlated with its nearest right and left neighbours. For the loadings $\boldsymbol{D}$, we know that they tend to be correlated in two dimensions on the Raman map.

We define the one-dimensional covariance as a Gaussian radial basis function, (8), and the two-dimensional covariance function as the exponential covariance function [12, table 4.1], (9), where $\|x - x^*\|$ is the Euclidean distance between two coordinates on the Raman map. In Figure 3 we show the structure of the two covariance matrices of a 5×5 toy example with $\beta = 2$.

Introducing the two-dimensional covariance matrix for the prior on $\boldsymbol{D}$ can be seen as our own addition to the NMF-GPP model proposed by [11]. Hence we will refer to this model as the NMF-GPP 2d. Using the same one-dimensional covariance matrix on both priors will be referred to as the NMF-GPP 1d model.

$$K_{1d}(x_i, x_j) = \exp\left\{-\frac{|i-j|^2}{\beta^2}\right\} \tag{8}$$

$$K_{2d}(x, x^*) = \exp\left\{-\frac{\|x - x^*\|}{\beta^2}\right\} \tag{9}$$



(a)

(b)

Fig. 3. (a) shows the structure of the 1d covariance and (b) the 2d covariance function.

2.4. Inference

We will do two different kinds of inference for this model, MAP estimation and Hamiltonian Monte Carlo. We also show infeasibility of Mean Field Variational Bayes.

2.4.1. Mean field variational inference

A fast way to approximate the posterior distribution of a Bayesian model is by approximate inference. In the supplementary material, Appendix B, we show that in all but very limited modeling cases, mean field VB is infeasible for this problem.

2.4.2. MAP estimation by direct minimization of loss

The MAP estimation technique follows [11], where they suggest a simple change of variable where they rotate the vectors $\boldsymbol{d}$ and $\boldsymbol{h}$ by the Cholesky decomposition of the covariance matrices, ie.

$$\boldsymbol{h} = C_h^T \boldsymbol{\eta} \tag{10}$$

where C_h is the Cholesky decomposition of Σ_h - and similarly we introduce $\boldsymbol{\delta}$ for $\boldsymbol{d}$. Then $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ are standard normally distributed and independent. We can then easily retransform back to $\boldsymbol{H}$ by

$$\boldsymbol{H} = \mathrm{Vec}^{-1}(f_h^{-1}(C_h^T \boldsymbol{\eta})) \tag{11}$$

where $\mathrm{Vec}^{-1}(\cdot)$ takes a vector and reshapes to a matrix of appropriate size.

Now, using the transformed variables $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ and due to the choice of likelihood and link functions, we are able to directly optimize the loss (7) for the MAP estimate - where we have the derivative of the loss and the derivative of the link functions.

The concrete equations are given in [11] and can be seen in the supplementary material, Appendix C.

2.4.3. Hamiltonian Monte Carlo

Hamiltonian Monte Carlo (HMC) works by defining the movement in the state space as the Hamiltonian evolution defined by the equation

$$H(q, p) = U(q) + K(p). \tag{12}$$

Here $U(q)$ is the negative log posterior from (7) and

$$K(p) = \frac{1}{2} p^T M^{-1} p, \tag{13}$$

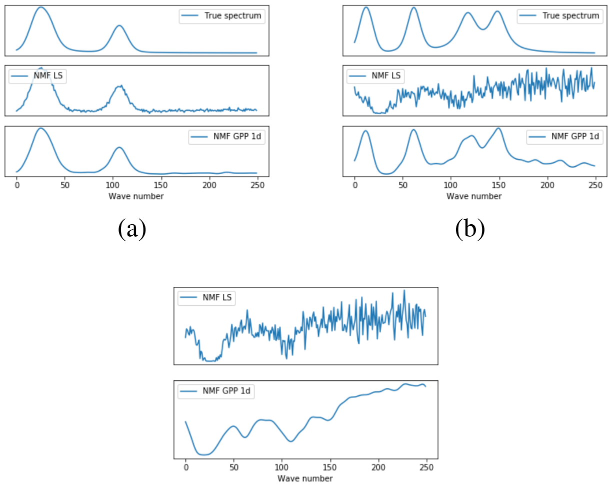
where M is a symmetric and positive definite matrix corresponding to the 'mass' of the system.

The requirements for using HMC is that the distribution of interest must be continuous on $\mathbb{R}^d$ and that a proportional density can be evaluated. Further it is necessary to compute the partial derivatives of the log of the density function [13]. As these requirements are satisfied this is implemented using the `pymc3` framework outlined in [14], allowing for a proof of concept as seen in the following section.

3. RESULTS & DISCUSSION

As recovering a single spectrum from simulated data seemed feasible for all of the proposed models, even when a lot of noise was present, we decided to create an additive mixture of two spectra to complicate the problem. The problem then becomes to extract two spectra and a background component. From this, we know that the optimal value for number of components is $M = 3$. In general however, we should estimate the number of components, perhaps based on prior knowledge. In all experiments, we use the link functions in [11] selected arbitrarily.

First, we compare the LS-NMF against the NMF-GPP 1d model as seen in Figure 4. Both models are able to capture the first spectrum in the mixture. LS-NMF is not smooth and contains an irregularity at approximately wavenumber 170, which could in principle be the behaviour of two overlapping peaks. As for the second spectrum, LS-NMF does not capture anything useful whereas the NMF-GPP 1d captures the spectrum, but also includes some small peaks that are not part the true spectrum. Through this, we argue that the NMF-GPP 1d is better at recovering the true spectrum.



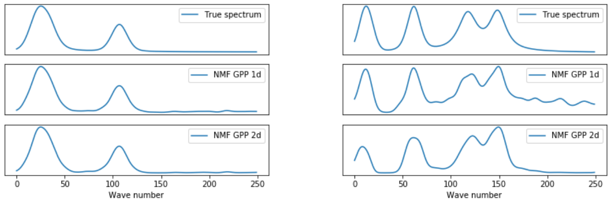
(a)

(b)

(c)

Fig. 4. Comparison of LS-NMF against GPP-NMF with $\beta = 10$. The visualized spectra corresponds to the columns of $\boldsymbol{H}$ in the decomposition. The first spectrum (a), the second spectrum (b) and the estimated background (c).

Next, we compare the NMF-GPP 1d against NMF-GPP 2d illustrated in Figure 5. The modified covariance for the prior on $\boldsymbol{D}$ slightly improves the model and removes the small peaks that are not part of the true spectrum.



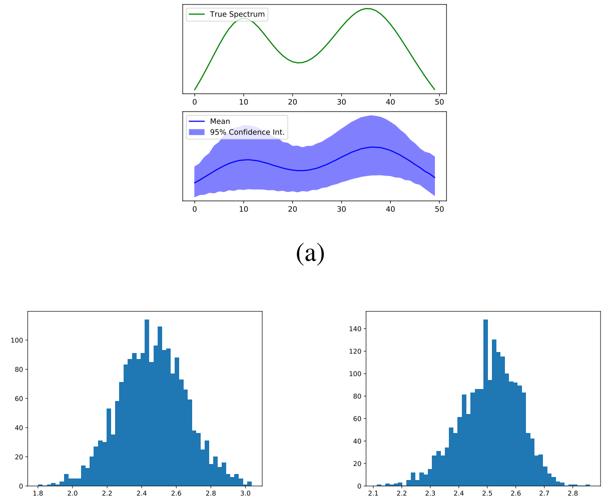
(a)

(b)

Fig. 5. Comparison of GPP-NMF 1d against GPP-NMF 2d with $\beta_{1d} = \beta_{2d} = 10$. The visualized spectra corresponds to the columns of $\boldsymbol{H}$ in the decomposition. The first spectrum (a) and the second spectrum (b).

As the implemented Hamiltonian sampling scheme was computationally inefficient, we decided to run a proof-of-concept example in order to show the general idea. We sampled values from $\boldsymbol{D}$ and $\boldsymbol{H}$ with observed data $\boldsymbol{X} \in \mathbb{R}^{400\times100}$ from a 20 by 20 map with 100 different wavelengths. We generated three sample chains with 2500 samples and discarded the first 500 as burn-in.

By calculating the mean and standard deviation of $\boldsymbol{d}$ and $\boldsymbol{h}$ before transforming back to the original domain using the inverse link function, we obtain the plot in Figure 6 for the first chain. Note that all chains found similar solutions. From this it is seen that we are able to reasonably identify the peaks from the true spectrum through sampling. Ideally we would like the confidence interval to be tight around the peaks, in order to reflect the certainty regarding their placements. This is not the case in Figure 6, but this could be due to the low amount of samples. Furthermore, sampling allows for approximation of hyperparameters, which in this case is the width of the covariance functions for $\boldsymbol{D}$ and $\boldsymbol{H}$.



(a)

(b)

(c)

Fig. 6. Mean value and 95% confidence interval found for smaller problem solved using HMC with 2500 samples and a burn-in of 500. The width of the covariance for $\boldsymbol{D}$ and $\boldsymbol{H}$ is seen in (b) and (c) respectively.

4. CONCLUSION

We implemented the Non-negative matrix factorization with Gaussian Process priors according to [11]. We introduced a covariance structure specifically designed to model the gaussian clustering of molecules on the Raman map. Results on a mixture of simulated Raman spectroscopy data show that the modified GPP-NMF seems to have slighly better performance recovering the true spectra. Furthermore, we provide a proof-of-concept Hamiltonian sampling scheme which can be used to estimate inference for the NMF-GPP model.

4.1. Future work

Possible future work is an application of our model to real data in order to properly assess its performance. Further the implemented Hamiltonian MC sampling scheme is very slow, and as such is not feasible for problems of interesting size. Therefore this would have to be implemented more efficiently, in order for a proper comparison of the inference through sampling to the MAP solution determined by GPP-NMF.

Acknowledgements

We would like to thank our supervisors Tommy Sonne Alstrøm and Mikkel Nørgaard Schmidt for great feedback and guidance throughout the duration of the project.

5. REFERENCES

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

[2] P. Sajda, S. Du, and L. C. Parra, "Recovery of constituent spectra using non-negative matrix factorization," *Proceedings of the Spie - the International Society for Optical Engineering*, vol. 5207, no. 1, pp. 321–31, 321–331, 2003.

[3] Tommy Sonne Alstrøm, Kasper Bayer Frøhling, Jan Larsen, Mikkel Nørgaard Schmidt, Michael Bache, Michael Stenbæk Schmidt, Mogens Havsteen Jakobsen, and Anja Boisen, "Improving the robustness of surface enhanced raman spectroscopy based sensors by bayesian non-negative matrix factorization," *Proceedings of the 2014 Ieee International Workshop on Machine Learning for Signal Processing (mlsp)*, p. 6958925, 2014.

[4] Hualiang Li, Tülay Adali, Wei Wang, and Darren Emge, "Non-negative matrix factorization with orthogonality constraints for chemical agent detection in raman spectra," *2005 Ieee Workshop on Machine Learning for Signal Processing*, pp. 1532909, 253–258, 2005.

[5] Cosmin Lazar and Andrei Doncescu, "Non negative matrix factorization clustering capabilities; application on multivariate image segmentation," *Cisis: 2009 International Conference on Complex, Intelligent and Software Intensive Systems, Vols 1 and 2*, vol. 2, no. 2, pp. 924–+, 2009.

[6] Chenglin Liu and Jinwen Ma, "Automatic non-negative matrix factorization clustering with competitive sparseness constraints," *Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8589, pp. 118–125, 2014.

[7] S Tsuge, M Shishibori, S Kuroiwa, and K Kita, "Dimensionality reduction using non-negative matrix factorization for information retrieval," *Ieee International Conference on Systems, Man and Cybernetics*, pp. 960–965, 2002.

[8] Zechao Li, Jing Liu, and Hanqing Lu, "Structure preserving non-negative matrix factorization for dimensionality reduction," *Computer Vision and Image Understanding*, vol. 117, no. 9, pp. 1175–1189, 2013.

[9] Zhenfeng Zhu, Yue-Fei Guo, Xingquan Zhu, and Xiangyang Xue, "Normalized dimensionality reduction using nonnegative matrix factorization," *Neurocomputing*, vol. 73, no. 10-12, pp. 1783–1793, 2010.

[10] SM Nie and SR Emery, "Probing single molecules and single nanoparticles by surface-enhanced raman scattering," *Science*, vol. 275, no. 5303, pp. 1102–1106, 1997.

[11] Mikkel Nørgaard Schmidt and Hans Laurberg, "Nonnegative matrix factorization with gaussian process priors," *Computational Intelligence and Neuroscience*, vol. 2008, pp. 361705, 2008.

[12] Carl Edward. Rasmussen and Christopher K.I. Williams, *Gaussian processes for machine learning*, MIT,, 2006.

[13] R. M. Neal, "MCMC using Hamiltonian dynamics," *ArXiv e-prints*, June 2012.

[14] John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck, "Probabilistic programming in python using pymc3," *PeerJ Computer Science*, vol. 2, pp. e55, Apr. 2016.

[15] Daniel D Lee and H Sebastian Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001, pp. 556–562.

Supplementary material

A. LEAST SQUARES NMF WITH MULTIPLICATIVE UPDATES

A way to find suitable matrix factors is by maximum likelihood estimation under certain assumptions of the distribution of $\boldsymbol{X}$. The original NMF paper [1], which we will follow, introduce a Euclidean least squares measure for the loss,

$$p_{\mathrm{LS}}(\boldsymbol{X}|\boldsymbol{D},\boldsymbol{H}) \propto \|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{H}\|^2, \qquad (14)$$

such that the maximum likelihood can be estimated by iteration of some very simple multiplicative updates,

$$\boldsymbol{H}_{ml} \leftarrow \boldsymbol{H}_{ml}\frac{(\boldsymbol{D}^T\boldsymbol{X})_{ml}}{(\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{H})_{ml}} \qquad (15a)$$

$$\boldsymbol{D}_{km} \leftarrow \boldsymbol{D}_{km}\frac{(\boldsymbol{X}\boldsymbol{H}^T)_{km}}{(\boldsymbol{D}\boldsymbol{H}\boldsymbol{H}^T)_{km}}. \qquad (15b)$$

They prove convergence of these updates in [15].

The *Least Squares NMF* (LS-NMF) has several downsides. It cannot in any way deal with negative data - all negative elements of $\boldsymbol{X}$ should be truncated to 0, and due to the fact that it naively minimizes the loss, (14), we have no guarantee that it will not fit to noise or mix the components in the model. On the other hand, it is computationally fast and easy to implement.

B. INFEASIBILITY OF MEAN FIELD VARIATIONAL BAYES FOR GPP-NMF

In this section, we will derive the Variational Bayes (VB) result from section 2.4 until intractable in all but very limited cases seen from a modeling perspective.
Recall that we want to compute the posterior distribution, $p(\boldsymbol{D},\boldsymbol{H}|\boldsymbol{X})$ in the Gaussian process prior framework.
We propose a mean field approach for the approximating distribution. That is, we suggest a family of approximating distributions, $Q(\boldsymbol{D},\boldsymbol{H})$ and assume that the best possible distribution (measured in the Kullback-Leibler divergence) in this family, $q(\boldsymbol{D},\boldsymbol{H})$ subjects to the mean field assumption,

$$q(\boldsymbol{D},\boldsymbol{H}) = q_D(\boldsymbol{D})q_H(\boldsymbol{H}). \tag{16}$$

Very generally, we may under this assumption write

$$\log p(\boldsymbol{X}|\boldsymbol{D},\boldsymbol{H}) = \mathcal{L}(q) + \mathrm{KL}(q\|p), \tag{17}$$

where $\mathcal{L}(q)$ is the free energy defined by

$$\mathcal{L}(q) = \iint q(\boldsymbol{D},\boldsymbol{H}) \log\left[\frac{p(\boldsymbol{X}|\boldsymbol{D},\boldsymbol{H})p(\boldsymbol{D})p(\boldsymbol{H})}{q(\boldsymbol{D},\boldsymbol{H})}\right] \mathrm{d}\boldsymbol{D}\mathrm{d}\boldsymbol{H} \tag{18}$$

and $\mathrm{KL}(q\|p)$ is the Kullback-Leibler divergence between q and p, defined by

$$\mathrm{KL}(q\|p) = -\iint q(\boldsymbol{D},\boldsymbol{H}) \log\left[\frac{p(\boldsymbol{D},\boldsymbol{H}|\boldsymbol{X})}{q(\boldsymbol{D},\boldsymbol{H})}\right] \mathrm{d}\boldsymbol{D}\mathrm{d}\boldsymbol{H}. \tag{19}$$

By simple insertion of the assumption in the free energy term, we get that,

$$\begin{aligned}\mathcal{L}(q) = \iint q_D(\boldsymbol{D})q_H(\boldsymbol{H})[&\log p(\boldsymbol{X},\boldsymbol{D},\boldsymbol{H}) \\ &- \log q_D(\boldsymbol{D}) - \log q_H(\boldsymbol{H})]\mathrm{d}\boldsymbol{D}\mathrm{d}\boldsymbol{H},\end{aligned} \tag{20}$$

from which one can relatively easily show that,

$$\begin{aligned}\mathcal{L}(q) = &\int q_D(\boldsymbol{D})\mathbb{E}_{q_H}[\log p(\boldsymbol{X},\boldsymbol{D},\boldsymbol{H})]\mathrm{d}\boldsymbol{D} \\ &- \int q_D(\boldsymbol{D}) \log q_d(\boldsymbol{D})\mathrm{d}\boldsymbol{D} + \text{const.}.\end{aligned} \tag{21}$$

Now we see that,

$$\mathcal{L}(q) - \text{const.} = \int q_D(\boldsymbol{D}) \log\left[\frac{\mathbb{E}_{q_H}[\log p(\boldsymbol{X},\boldsymbol{D},\boldsymbol{H})]}{q_D(\boldsymbol{D})}\right] \mathrm{d}\boldsymbol{D} \tag{22}$$

$$= -\mathrm{KL}(q_D(\boldsymbol{D})\|\mathbb{E}[\log p(\boldsymbol{X},\boldsymbol{D},\boldsymbol{H})]) \tag{23}$$

from which we clearly see that choosing,

$$\log q_D(\boldsymbol{D}) = \mathbb{E}_{q_H}[\log p(\boldsymbol{X},\boldsymbol{D},\boldsymbol{H})] \tag{24}$$

will minimize the Kullback-Leibler divergence between q and the posterior, such that we get the best possible approximating distribution.
The mean field variational inference scheme thus relies on the integral in the expectation in (24) being feasible.
This limits modeling flexibility considerably. An obvious example of a model in which this would be feasible is by selecting a Gaussian likelihood, such that the Gaussian Process Prior would be conjugate. However, the non-negativity constraint on the matrix factors requires utilization of a link function, as discussed in section 2. As discussed in the original paper, [11] the choice of link function should be strictly increasing to ensure that the inverse exists - thus eliminating this simple model from this framework.
The above shows the infeasibility of mean field VB in the framework of GPP-NMF.

C. MAP ESTIMATION - EQUATIONS

In this section we will state the equations needed for direct optimization of the posterior, (7), to achieve the MAP estimate under the chosen prior.

We will use the same terminology as in the previous part.

Under the transformation to the i.i.d standard Gaussian vectors $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ we get the negative log-posterior,

$$\mathcal{L}_{\delta,\eta|X}(\boldsymbol{\delta},\boldsymbol{\eta}) = \frac{1}{2}\left(\frac{1}{\sigma^2}\|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{H}\|_F^2 + \boldsymbol{\delta}^T\boldsymbol{\delta} + \boldsymbol{\eta}^{\boldsymbol{T}}\boldsymbol{\eta}\right), \tag{25}$$

where we can recover $\boldsymbol{D}$ and $\boldsymbol{H}$ by the correspondence in (11). Clearly the MAP estimate is achieved by minimizing (25).

The gradient of (25) wrt. $\boldsymbol{\eta}$ is,

$$\begin{aligned}\nabla_\eta \mathcal{L}_{\delta,\eta|X} = \frac{1}{\sigma^2}&[\mathrm{Vec}(\boldsymbol{D}^T(\boldsymbol{D}\boldsymbol{H}-\boldsymbol{X})]* \\ &(f_h^{-1})'(C_h^T\boldsymbol{\eta}))^T C_h + \boldsymbol{\eta}\end{aligned} \tag{26}$$

where $*$ denotes elementwise multiplication.

Similarly, the gradient wrt. $\boldsymbol{\delta}$ is,

$$\begin{aligned}\nabla_\delta \mathcal{L}_{\delta,\eta|X} = \frac{1}{\sigma^2}&[\mathrm{Vec}(\boldsymbol{H}(\boldsymbol{D}\boldsymbol{H}-\boldsymbol{X})^T]* \\ &(f_d^{-1})'(C_d^T\boldsymbol{\delta})^T C_d + \boldsymbol{\delta}.\end{aligned} \tag{27}$$

Using these derivatives, an unconstrained minimization strategy can be used to find the MAP estimate. We use the L-BFGS-B algorithm for this.

D. IMPLEMENTATION

The provided code can be found in the public github repository: Github Repository Link[1]. The `README.md` contains a

[1] https://github.com/Rasmusafj/02460-NMF-with-GP-priors

brief description of the included python files and a setup guide to running the code.