Some notes on Bayesian nonnegative matrix factorisation

Shota Gugushvili

Biometris, Wageningen University & Research

February 1, 2025

Abstract

These notes provide some information on Bayesian non-negative matrix factorisation.

1 Introduction

These notes provide some information on PCA and Bayesian non-negative matrix factorisation. They are partly based on Hendrik Steinbach's master thesis "Matrix factorization techniques for dimensionality reduction and clustering: a comparative study in genomics". Hendrik wrote the thesis at Leiden University and WUR under Fred van Eeuwijk's and my supervision. In order to stay on the didactic side, notes do not go into full details and extensive comparisons. They do contain references to the literature. Description of the singular value decomposition and PCA is just enough to understand similarities to and differences from non-negative matrix factorisation. Knowledge of their other properties is tacitly assumed.

2 Generalities

Given is a data matrix Y of dimensions $U \times I$. Think of U as a number of subjects and I as a number of variables[1]. The uth row of Y is an observation vector for the uth subject. The ith column contains the ith variable values.

I aim at extracting signal Λ from noisy observations Y. The matrix Λ has the same dimensions as Y. A schematic picture is given in Figure 1.

The signal Λ is assumed to be low-rank, as detailed next. Consider a typical situation in modern applications, where the number of subjects is smaller than

[1]Notation comes mostly from Kucukelbir et al. [2017] and might not be ideal, but I left it as such. For instance, U stands for users and I stands for items: the original application there was the movie ratings dataset.

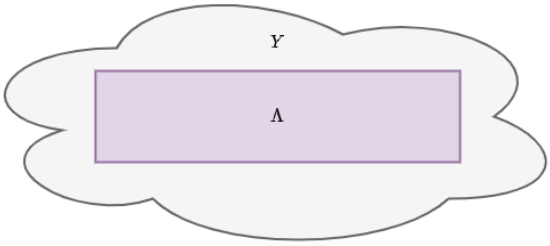

Figure 1: Schematic depiction of signal Λ within noisy observations Y.

the number of variables: $U < I$ (often $U \ll I$). Assuming that there are no linear dependencies among the rows of Y, the rank of Y is $\min(U, I) = U$. A low-rank approximation to Y is

$$\Lambda = \Theta \times B. \tag{1}$$

Here Θ is $U \times K$-dimensional with $K < I$, while B is $K \times I$-dimensional. See Figure 2 for a visualisation of a relationship between various matrices at hand. The rank of Λ is at most K, and we know $K < U$. Typically one chooses K such that

$$K(U + I) < UI.$$

This ensures that the matrices Θ and B combined have less entries than the matrix Λ. Thereby less free parameters are needed to describe Θ and B than a totally unstructured Λ would require.

Columns of matrix Θ can be thought of as latent meta-variables. Rows of matrix B give a basis in which these meta-variables are expressed. A single row of matrix Λ is a linear combination of the rows of matrix B, where weights come from the columns of matrix Θ; see Figure 3.

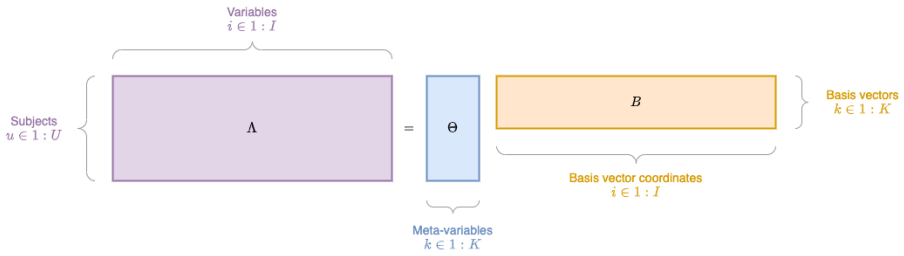

Figure 2: Schematic depiction of the relationships between matrices Λ, Θ and B.

In the audio signal processing literature the matrix Y is a spectrogram, B is a codebook of spectra consisting of basis vectors, while Θ is a matrix of their gains in each frame; see, e.g., Virtanen et al. [2008].

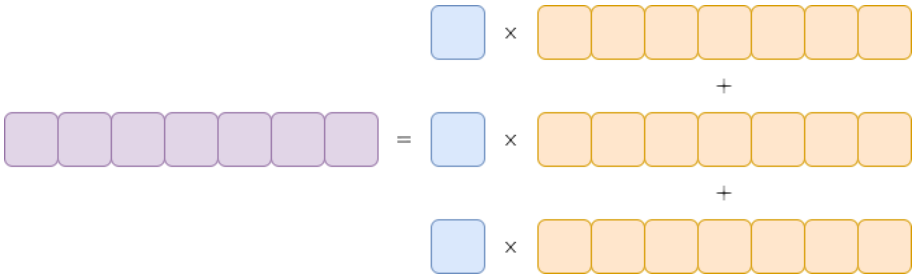

Figure 3: Each row of matrix Λ is a linear combination of the rows of basis matrix B with weights from columns of matrix Θ.

Matrix Θ can be used to study patterns in the data and for clustering. A low-rank representation aids in that. Feature engineering is another natural application.

3 SVD and PCA

SVD gives

$$Y = \mathcal{U} \times \Sigma \times \mathcal{V}^T.$$

Here the matrix Σ is the diagonal matrix of singular values of Y in decreasing order, $\mathcal{U}$ is the matrix of the corresponding left singular vectors of Y and $\mathcal{V}$ is the matrix of the right singular vectors. Matrices $\mathcal{U}$ and $\mathcal{V}$ have orthonormal columns. The main diagonal of the matrix Λ, on the other hand, has positive entries. See Strang [2016], Section 7.2.

A low-rank approximation to Y is obtained by retaining the first K columns of $\mathcal{U}$ and $\mathcal{V}$, and the upper-left $K \times K$ part of Σ,

$$\Lambda = \mathcal{U}_K \times \Sigma_K \times \mathcal{V}_K^T,$$

see Murphy [2022], Section 7.5.5. The Eckart-Young-Mirsky theorem states that this Λ gives the lowest reconstruction error in the Frobenius norm over all matrices X of rank K:

$$\Lambda = \underset{X:\mathrm{rank}(X)=K}{\arg\min} \|Y - X\|_F.$$

See Strang [2016], page 393. There are various ways in which the Frobenius norm can be introduced, but for now it is enough to say that it is the square root of the sum of the squared entries of a matrix.

To link SVD to (1), set $\Theta = \mathcal{U}_K \times \Sigma_K$ and $B = \mathcal{V}_K^T$.

Finally, the relationship of SVD to PCA is as follows: assume Y is centred (this is essential). Then the columns of the matrix $\mathcal{U} \times \Sigma$ are principal components or scores, while the columns of $\mathcal{V}$ are principal axes. Furthermore, the standardised scores are given by columns of $\sqrt{U-1} \cdot \mathcal{U}$, while loadings are the

columns of $\mathcal{V}\Sigma/\sqrt{U-1}$. See Murphy [2022], page 659 for details and further pointers.

SVD is a modern way of performing the PCA decomposition[2].

There are probabilistic and Bayesian reformulations of PCA, but I will not consider them here. See, e.g., Section 12.2 in Bishop [2006] for details.

4 NMF

Assume Y has non-negative entries (this is important), but otherwise they are allowed to be integer- or real-valued. There are two commonly used versions of NMF. Either assumes the matrices Θ and B in (1) to have non-negative entries. One possibility is that the pair (Θ, B) is a minimiser of the criterion $\|Y - X\|_F$ over X such that $X = W \times H$ for W and H with non-negative entries. This is least squares estimation of an approximating Λ, subject to the stated constraints on Θ and B.

Another possibility is to maximise the criterion

$$\sum_{u=1}^{U}\sum_{i=1}^{I}[Y_{ui}\log(W\times H)_{ui} - (W\times H)_{ui}] \tag{2}$$

subject to non-negativity constraints on the entries of W and H. If Y were integer-valued, this would amount to a Poisson generative model for Y, i.e.

$$Y \sim \text{Poisson}(\Lambda),$$

and maximisation of the corresponding log-likelihood $\ell_Y(X)$ over matrices X subject to the constraint $X = W \times H$ for W and H with non-negative entries. Cf. Lee and Seung [1999]. An iterative algorithm for non-negative matrix factorisation from Lee and Seung [1999] reduces to the EM algorithm; this can be shown by examining the equations for the E and M steps, see Cemgil [2009]. Strengths and limitations of the EM algorithm are part of the statistical folklore.

There are no further restrictions on Θ and B except non-negativity of their entries. This is unlike SVD/PCA. However, both methods try to solve a matrix approximation problem.

When further restrictions are imposed on the factors in the decomposition, links with various clustering algorithms emerge; see Ding et al. [2005]. For instance, the K-means algorithm can be recovered. However, introducing extra (non-linear) constraints is not necessarily the best strategy to proceed with matrix factorisations from the numerical point of view.

NMF has some Bayesian roots, see Richardson [1972].

[2]Traditionally, PCA has been done via spectral decomposition (eigendecomposition) of the sample covariance or correlation matrix. However, forming the sample covariance matrix when the number of features is large is costly, and furthermore, SVD yields numerically more stable computations. Cf. Ripley [1996], page 290. Thus, `prcomp` in **R** relies on SVD, while `princomp` follows an older route for compatibility with S-PLUS. PCA in **scikit-learn** in **Python** is implemented through SVD and randomised truncated SVD.

5 Benefits of NMF

PCA has been widely used in, e.g., social sciences, but has found a limited use in natural sciences (Paatero and Tapper [1994]). The reason is the non-negativity constraint, which is natural in many physical problems. For instance, there cannot be a negative amount of a basic constituent in any sample, nor can the composition of any basic constituent contain a negative percentage of any element; see Paatero and Tapper [1994], page 112. PCA cannot guarantee non-negativity of the matrix decomposition, while this property is built into NMF.

Paatero and Tapper [1994], page 120, argue that orthogonality in matrix decompositions is irrelevant in physical sciences. PCA delivers orthogonality, while it is in general precluded by the non-negativity constraint.

Lee and Seung [1999] illustrate and contrast performance of NMF and PCA on image data. According to them, the non-negativity constraints are compatible with the intuitive notion of combining parts to form a whole, which is how NMF, they say, infers a parts-based representation[3], as it focusses on learning localised features in the data. This is not possible with PCA.

NMF often delivers already quite sparse solutions in those situations where PCA does not. Sparsity helps with interpretability, and sparse solutions fed into other procedures often give better practical results than non-sparse ones. Think of clustering algorithms: it is easier to differentiate cases where only a few components are large, as it is less likely that different clusters have the same components large. Obviously, there exist also sparse versions of PCA. But these are non-trivial extensions computationally, whereas the basic SVD algorithm is straightforward.

Where the uses of PCA have outweighed those of NMF, this seems to an extent a consequence of inertia in those fields.

6 Bayesian NMF

Bayesian NMF postulates a parametric data generation mechanism. Model parameters are equipped with priors. Inference is based on the posterior distribution of the parameters.

A parametric statistical model for Bayesian NMF from Virtanen et al. [2008] and Cemgil [2009] assumes

$$Y \sim \text{Poisson}(\Lambda),$$

for $\Lambda = \Theta \times B$. This notation has to be understood entry-wise. It is assumed that the entries of Y are non-negative integers (note that restriction to integers is not necessary for optimisation of expression (2)). Other possibilities for the likelihood are, e.g., a binomial likelihood or a Gaussian likelihood (cf. Mnih and Salakhutdinov [2007]).

[3]Later work has shown that parts-based representations are not always automatically delivered by NMF. However, this can be achieved with some suitable modifications of the basic algorithm; see, e.g., Hoyer [2004].

A Bayesian model is obtained from this parametric model by imposing priors on Θ and B. There are various possibilities. Earlier works focussed on Gamma priors due to the well-known Poisson-Gamma conjugacy property. This came in handy in the Gibbs sampler and the coordinate ascent variational inference; see Cemgil [2009]. However, with the advent of modern Bayesian software such as **Stan**, see Carpenter et al. [2017] and Stan Development Team [2024], conjugacy is thought of being less important. Furthermore, a sequential nature of coordinate updates in the Gibbs sampler and the coordinate ascent variational inference, and possible strong correlations between latent variables[4] renders these methods inherently slow for modern applications focussing on big data; cf. Blei et al. [2017], page 869 and Lindsten and Schön [2013], Section 1.3.

I will bypass conjugacy altogether[5]. The following two properties are required from the prior: act as a reguliser for numerical stability of computations, and encourage sparse representations of the signal.

I start with the prior on B, which is defined as

$$B_{ki} \sim \text{Exponential}(1)$$

for $k = 1, \ldots, K$ and $i = 1, \ldots, I$. The scaling of the prior is immaterial, in that it gets compensated by the multiplicative nature of decomposition (1) through matrix Θ. Here the scale parameter is set to 1 for convenience. The prior ensures positivity of the entries of B. There is no specific structure assumed about the basis vectors, though it may be present in some applications. When such a structure is present, it can be built into the prior, but this has to be done on a case-by-case basis. Using the exponential prior is like using the L_1-penalty on the entries of the matrix B coupled with non-negativity.

The prior for Θ uses a more involved construction. In practice the number K of the required meta-variables is often unknown. Automatic Relevance Determination (ARD) that was first proposed in the context of neural networks by MacKay and Neal (see MacKay [1995]) is a simple, yet efficient strategy to address this issue; cf. Bishop [2006], page 582. This method starts with taking K large for a given application. It then requires using a different shrinkage parameter per each column of Θ that effectively switches it off, should the column turn out to be superfluous. In practice the method may overshrink some columns relative to others. This behaviour can be repaired by using a global shrinkage parameter that acts on the entire matrix Θ. At the same time, as its name suggests, this parameter promotes sparsity throughout the entire matrix Θ. Figure 4 may help in understanding the main ideas. ARD is computationally more convenient than equipping the number of meta-variables K with a prior. The latter leads to a family of models with varying dimensions, inference for which is computationally challenging; see Green [1995] and Godsill [2001]. ARD deals with a model of fixed dimension, but has a built-in ability of essentially turning off the surplus dimensions.

[4] Signals are structured.

[5] For non-Bayesians this should lift a veil of mystery from the use of the Gamma prior in this setting.

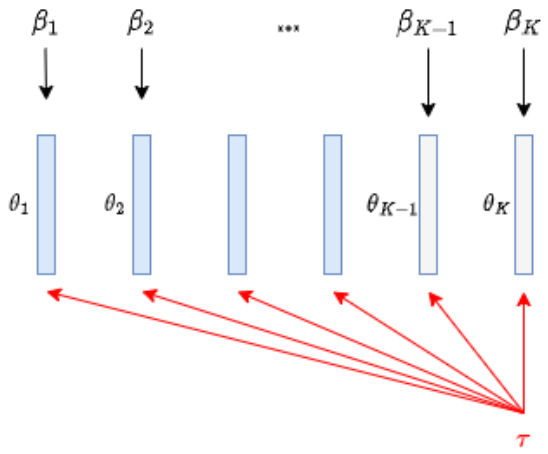

Figure 4: Schematic depiction of ARD for NMF. Columns of matrix Θ are denoted by θ_k. Here the last two columns are superfluous, which is indicated by their lighter colouring. Shrinkage parameters β_k act on individual columns θ_k. The global shrinkage parameter τ acts on the entire matrix Θ.

The exact details of the construction are as follows: at the lowest level,

$$\theta_{uk} \sim \text{HalfNormal}(0, \tau\beta_k)$$

for $u = 1, \ldots, U$ and $k = 1, \ldots, K$. Here $\text{HalfNormal}(0, \tau \cdot \beta_k)$ stands for Normal distribution with mean 0 and standard deviation $\tau \cdot \beta_k$, and constrained to be positive. The parameter τ is the global shrinkage parameter, while β_k's are the local shrinkage parameters. Small values of τ and β_k's correspond to greater degree of shrinkage.

In the next layer of hierarchy,

$$\beta_k \sim \text{HalfStudent}(3, 0, 5)$$

for $k = 1, \ldots, K$, and

$$\tau \sim \text{HalfStudent}(3, 0, 5).$$

Here $\text{HalfStudent}(3, 0, 5)$ stands for Student distribution with 3 degrees of freedom, location 0 and scale 5, and constrained to be positive. Hyperparameters of the priors are not carved in stone and may be modified as required in specific applications. The whole construction is somewhat inspired by the horseshoe prior; see, e.g., Carvalho et al. [2009]. Heavy-tailed priors like Student have better shrinkage properties than the light-tailed ones (like Gaussian)[6]. There is no direct analogue of these priors among the L_p penalisation methods.

[6]See Carvalho et al. [2009], or `https://www.stephanievanderpas.nl/shrinkage-priors` for relevant references dealing with theory.

Aggregate all the parameters and latent variables of the Bayesian model into Z. Bayes' theorem gives the posterior density

$$p(Z|Y) = \frac{p(Y|Z)p(Z)}{\int p(Y|Z)p(Z)\,\mathrm{d}Z}. \tag{3}$$

Posterior can then by used to obtain point estimates of Θ and B, and to conduct inference.

The hyperparameters β_k can be thought of as a rough analogue of the singular values in SVD/PCA. Their posterior point summaries (means or medians) can thus be used to rank meta-variables in order of their importance.

My final remark concerns identifiability. The Bayesian model above is not identifiable, in that swapping columns in Θ together with corresponding rows in B leads to the same product Λ, and hence the same log-likelihood. A consequence of that is a multimodal posterior. Various solutions to fix the identifiability issue in Bayesian factor models are discussed in Murphy [2022], Section 20.2.4; cf. also Kucukelbir et al. [2017]. I will use none out of purely practical considerations: firstly, they incur additional computational cost. Secondly, they are not necessarily natural restrictions in every application. Thirdly, my engine to compute approximate posterior will be variational inference with a unimodal family of approximating distributions. The variational density in such cases locks into one of the posterior modes due to special properties of the variational criterion, see Bishop [2006], page 469. The latter is enough for my purposes. Which mode is selected depends on the initialisation of the algorithm.

7 Benefits of Bayesian approach

Bayesian approach solves the problem of choosing the number of meta-variables K. It allows an intuitive specification of sparsity degree of the signal through a suitable choice of prior distributions. Arguments for advantages of sparse coding of a signal in, e.g., sensory systems are given in Field [1994], and conform to the intuition that, e.g., natural images may generally be described in terms of a small number of structural primitives, such as edges or lines; see Olshausen and Field [1997], page 3315. Even if appealing to sensory systems is not fully convincing, sparse coding represents a useful first approximation. NMF in its basic form does not always automatically deliver a good parts-based representation of the signal (see Hoyer [2004], page 1460), but this can be encouraged through the use of prior distributions. Sparsely coded signals are more amenable to inferring hidden structures in the data: for such signals only a few basis vectors are active at a time, and qualitatively different signals have a greater chance of having different components active, and thus being distinguishable. Bayesian priors are more flexible tools than the commonly used L_p penalisation methods. Moreover, a Bayesian approach also allows an extension of the basic NMF model to a probabilistic mixture of NMF models. This is an attractive alternative to non-linear modelling (e.g., Self-Organizing Maps). Finally, a Bayesian approach

can also handle missing data without need of an imputation algorithm. I will not consider this here.

8 Variational inference

The marginal likelihood or the evidence

$$\int p(Y|Z)p(Z)\,\mathrm{d}Z$$

in (3) is intractable, and hence posterior is not readily accessible. Therefore an approximation method has to be used. MCMC is not scalable in the present context; cf. Kucukelbir et al. [2015] and Kucukelbir et al. [2017]. Instead I will use variational inference to approximate the posterior. This converts the problem of finding the approximate posterior into an optimisation task. The variational objective called the evidence lower bound (ELBO) does not involve the untractable marginal likelihood. The key ideas are explained in Figure 5. Further details are given, e.g., in Bishop [2006], Chapter 10, or Murphy [2023], Chapter 10. A recent overview article is Blei et al. [2017]. Variational inference has its roots in statistical physics, where the method is used to approximate the distribution of a system by minimising the variational free energy; see MacKay [2003], Chapter 33.

There are various algorithmic realisations of variational inference. I will employ the automatic differentiation variational inference as implemented in **Stan**; see Kucukelbir et al. [2015] and Kucukelbir et al. [2017].

ADVI starts by internally transforming the constrained latent variables to the real coordinate space, while leaving unconstrained variables unchanged. In the real coordinate space, it postulates a Gaussian approximating family to the posterior. The user has a choice between a Gaussian with a diagonal covariance matrix, and a Gaussian with a general covariance matrix. The former case is referred to as the mean-field inference, while the latter as the full-rank inference. The full-rank Gaussian family is more flexible, while the mean-field family leads to faster computations and often yields numerically more stable runs. The mean-field family takes its name from the mean field theory in physics, which is a special case of a general variational free energy approach of Feynman and Bogoliubov (see MacKay [2003], page 422). Optimisation of the variational objective in ADVI is performed through stochastic gradient descent (SGD) and relies on automatic differentiation (AD) for fast gradient evaluations. Once a (local) optimum has been found, a sample of a given size is drawn from the variational density and back-transformed to the constrained latent space. This enables quick summarisation and plotting of inference results[7]. On the other hand, with a large number of latent variables, the storage might become an issue

[7]Technically, the entire variational density is available analytically. But when the number of latent variables is large (which is raison d'être of the variational approach in the first place), it is a cumbersome object to manipulate. On the other hand, a sample from the variational density is easier to handle in inferential, summarisation and visualisation tasks.

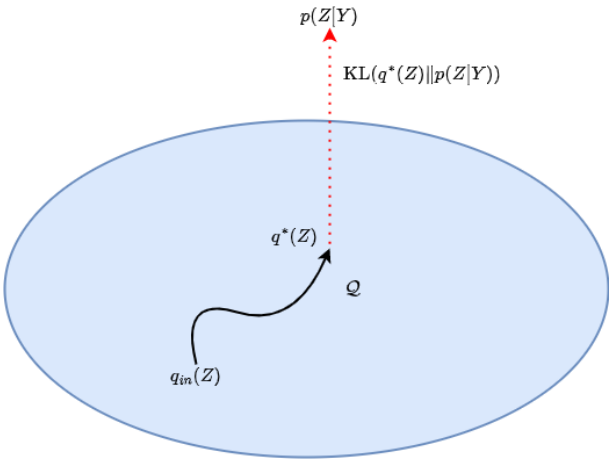

Figure 5: Schematic depiction of variational inference. The intractable posterior density $p(Z|Y)$ lies outside of the tractable approximating family $\mathcal{Q} = \{q(Z)\}$. Variational inference proceeds from the initial approximation $q_{ini}(Z)$ and iteratively improves upon it through optimising the evidence lower bound (ELBO). The iterative process is stopped once arriving at the optimal solution $q^\star(Z)$. This optimal solution minimises the Kullback-Leibler divergence between the family $\mathcal{Q}$ and the posterior $p(Z|Y)$.

(my experience). There are several user-defined tuning parameters in ADVI, but the default choices often work well and provide a natural starting point. It is probably a good practice to run the algorithm several times to assess stability of the obtained results.

Advantages of ADVI include speed, scalability, and its essentially automatic nature. A disadvantage is that the method is approximate and cannot recover the exact posterior distribution in the limit when the number of iterations is taken to infinity. In particular, even if the centre of the posterior distribution is well-approximated, the method may underestimate its spread, and hence inferential uncertainty; see, e.g., Bishop [2006], page 467. Gaussian approximation might be too crude as a distributional assumption in some situations. If the goal is recovery of the centre of the posterior distribution, with some (possibly crude) measure of uncertainty, variational inference is typically fine.

9 Bayesian NMF via ADVI

I provide an implementation of the Bayesian NMF model via ADVI in **Stan**. To that end it is enough to specify relationships between various variables defing the model, while the inference itself takes place under the hood. This is probabilistic programming, see Carpenter et al. [2017]. There is no need to derive manually the iterative update equations for the variational density. **Stan** is the most

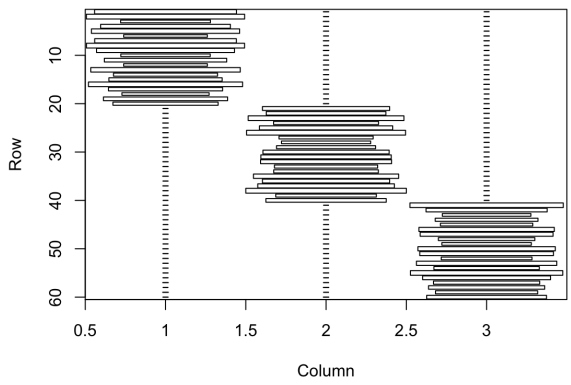

Figure 6: Hinton diagram for matrix Θ. The box sizes represent the relative sizes of the corresponding matrix entries.

widely used general-purpose software for Bayesian computations; see Štrumbelj et al. [2024].

The starting point of the implementation was the one proposed in Figure 19 in Kucukelbir et al. [2017], but this was radically modified, e.g., via incorporating ARD and dropping the identifiability constraint (which proved to cause numerical problems, when I experimented with it). Priors in general are different.

10 Simulated data example

Here is a simulated data example. The meta-variable matrix Θ is 60×3-dimensional. The basis matrix B is 3×900-dimensional. This gives a 60×900-dimensional signal matrix Λ, that serves as a mean matrix for the Poisson-distributed observation matrix Y.

The matrix Θ has roughly a block-diagonal structure: large entries appear in 3 blocks along the main diagonal, while other entries are small. See Figure 6 for a visualisation. A similar structure is assumed for the matrix B. Details can be found in the accompanying computer code, for visualisation of B, Λ and Y see Figures 7, 8 and 9. Some structure in matrix Y is apparent to the bare eye. The main interest lies in recovering the structure of matrix Θ.

Bayesian NMF was performed assuming $K = 5$ latent dimensions. Two dimensions are thus superfluous. The posterior median for Θ is shown in Figure 10. The structure recovery is excellent, as seen from comparison to Figure 6. The 2 surplus dimensions are essentially switched off.

It is instructive to examine posterior for shrinkage parameters β_k in Figure 11: those for surplus variables are squashed towards zero.

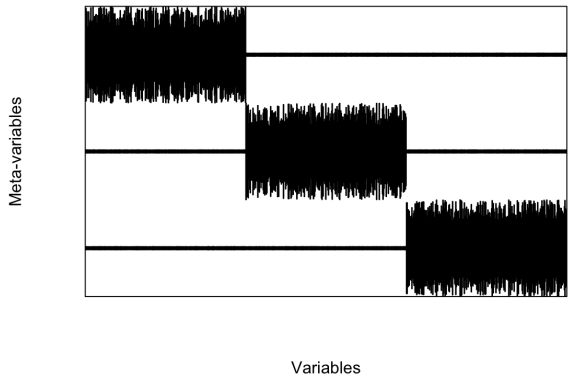

Figure 7: Hinton diagram for basis matrix B.

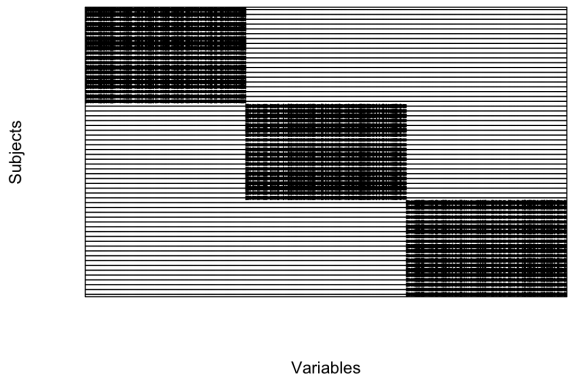

Figure 8: Hinton diagram for signal matrix Λ.

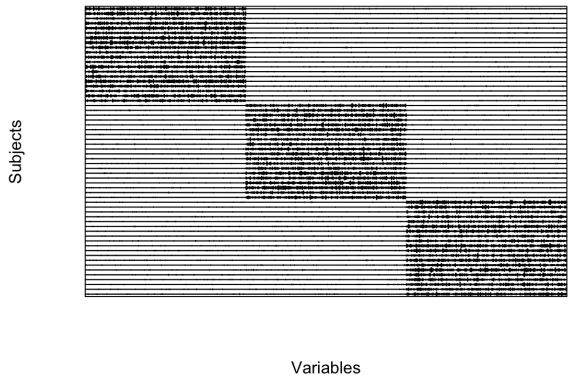

Figure 9: Hinton diagram for observation matrix Y.

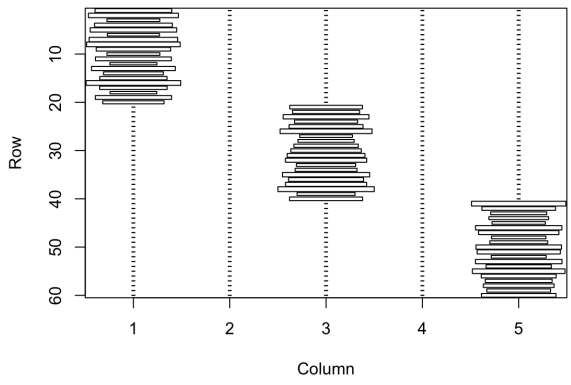

Figure 10: Hinton diagram for the posterior median of matrix Θ, where $K = 5$ latent dimensions were assumed.

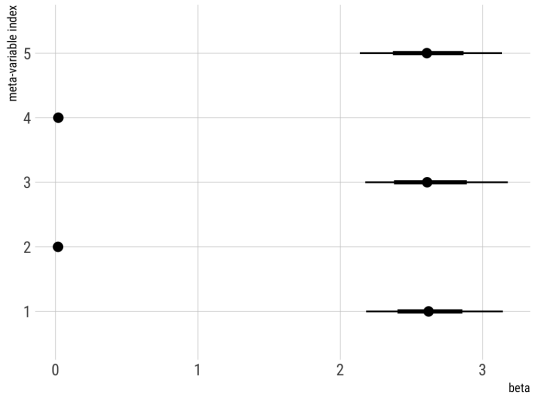

Figure 11: Posterior medians (points) and 90% credible intervals (segments) for shrinkage parameters β_k.

Posterior median for matrix Θ can be replotted with columns arranged according to the sizes of the inferred shrinkage parameters β_k. See Figure 12.

The first two meta-variables are enough to discern 3 clusters in the original signal. This is clear from Figure 12, but also from Figure 13.

For comparison, one can also perform PCA (via `prcomp` in **R**) on the centred and scaled data matrix, retaining 5 components, say. These 5 components explain 63% of variation in data. The resulting matrix of scores is shown in Figure 14. Several conclusions emerge from these plot: if the goal of the analysis were a recovery of 3 clusters in the data, then PCA performs indeed well. Thus, one cluster loads negatively the first principal component and gives low score to others, another cluster loads negatively the second principal component and gives low score to others, while the third cluster loads positively the first and second components, giving low scores to others[8]. Like for Bayesian NMF, two principal components are enough to cluster data, see Figure 15. On the other hand, the original matrix Θ only had non-negative entries, and these are not be retained with PCA.

Final remark concerns computational times: on a MacBook Air (2020) with Apple M1 chip with 8 cores (4 performance and 4 efficiency) and 8 GB RAM, running variational inference took just under 40 seconds. PCA was obviously massively faster. On the other hand, 40 seconds is not an impossibly long waiting time for a toy example.

[8] As noted in the corresponding **R** help page, the signs returned by the decomposition may differ between different builds of **R**.

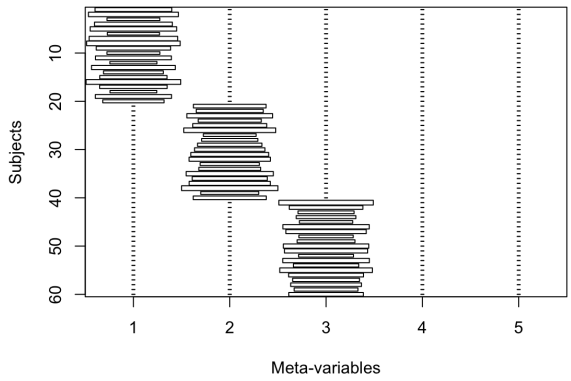

Figure 12: Hinton diagram for the posterior median of matrix Θ, where $K = 5$ latent dimensions were assumed. Columns are reordered according to the sizes of the inferred shrinkage parameters β_k.

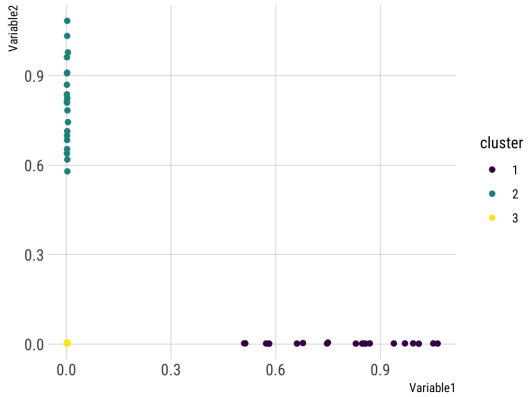

Figure 13: First two latent variables according to the sizes of the inferred shrinkage parameters β_k. Points in the third cluster are visually indistinguishable from each other, as their coordinates are very small.

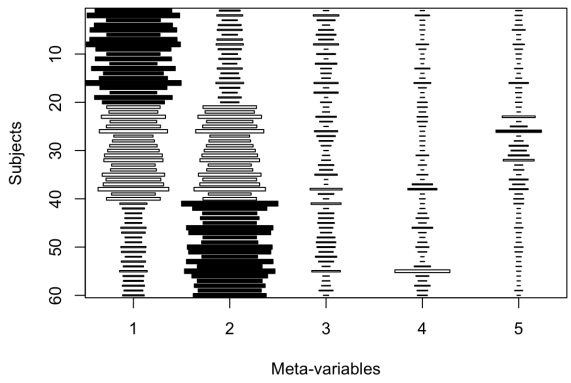

Figure 14: Hinton diagram for the principal components, where $K = 5$ principal components have been retained.

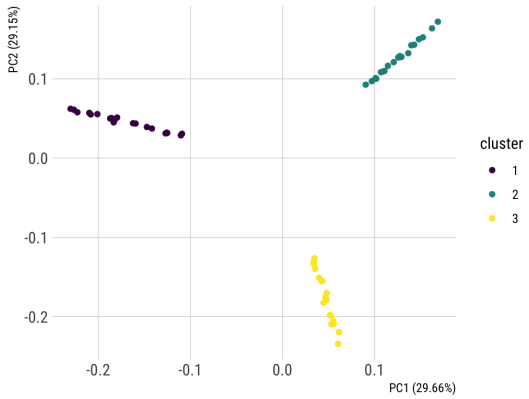

Figure 15: First two principal components. Points are coloured according to clusters in the original data.

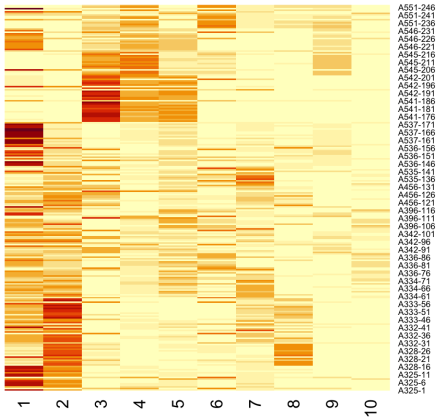

Figure 16: Heatmap for the posterior median of matrix Θ, where $K = 10$ latent dimensions were assumed. Columns are reordered according to the sizes of the inferred shrinkage parameters β_k. Darker colours correspond to larger coefficients.

11 Apple data example

Here is a small real data example. The dataset **data.apple**, that was extracted from the supplementary materials of Kumar et al. [2015], is shipped with the **R** package **ASRgenomics**. It contains genotypic data on 247 apple clones (genotypes) with a total of 2,828 SNP markers (coded as 0, 1, 2, with no missing values). Genotypes were selected from 17 full-sib apple families with 25 advanced selections as parents. There were 15 genotypes per family[9].

The help page of the `snp.pca` command in **ASRgenomics** analyses these data with PCA following the methodology in Patterson et al. [2006] and using 10 principal components. I assumed the latter value is a reasonable starting point and fixed the dimension of the latent space at $K = 10$. Only the SNP markers and no additional genetic information was used when performing the Bayesian NMF decomposition.

Posterior median for the Θ matrix is shown in Figure 16. The last latent variable seems to contribute little to visualisation[10].

As the family information is available, the Bayesian NMF results can be validated. I performed K-means clustering on the posterior median of Θ with $K = 17$ cluster centres via `kmeans` in **R**. The Rand score between this clustering and the truth was 0.93. Results varied slightly by choosing different random

[9] Thus $17 \times 15 = 247$.

[10] Whereas for Hinton diagrams I used the `color2D.matplot` command from the **plotrix** package, this broke down in the present example. Therefore I switched to the heatmap, given that matrix to be plotted is much larger now. Maybe I should have done all the programming in **Python**, not in **R**.

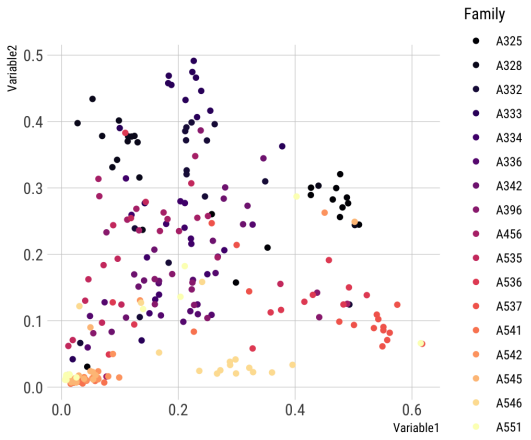

Figure 17: First two latent variables according to the sizes of the inferred shrinkage parameters β_k. Colours indicate families.

seeds, but did not go below 0.93. One can also plot the first two latent variables with the family information superimposed, but I do not find this graph particularly useful or enlightening here. That the Bayesian NMF performed reasonably well should not be taken for granted: data are a collection of 0, 1 and 2, which is not very Poisson-like, but rather more appropriate to be analysed with a binomial likelihood.

The Bayesian NMF Results can be compared to those obtained with PCA. With 10 retained principle components, the proportion of the explained variance is 85%. The score matrix in Figure 18 is visually denser than the one for Bayesian NMF.

The family information can be superimposed on the plot of the first two principal components as in Figure 19. Visually the first two components clearly identify several families, but for others the things are less clear. As already noted in Kumar et al. [2015], in almost every family there are individuals that do not cluster within their pedigree-assigned full-sib family groupings. On the other hand, the K-means clustering on the principal components gave the 0.93 Rand score when compared to the reference truth. This is not different from the Bayesian NMF.

According to Venables and Ripley [2002], page 316, "Do not assume that 'clustering' methods are the best way to discover interesting groupings in the data; in our experience the visualization methods are often far more effective. There are many different clustering methods, often giving different answers, and so the danger of over-interpretation is high". Here visual interpretation would have been hampered by the fact that there are 17 families.

Final remark concerns computational times: on a MacBook Air (2020) with Apple M1 chip with 8 cores (4 performance and 4 efficiency) and 8 GB RAM,

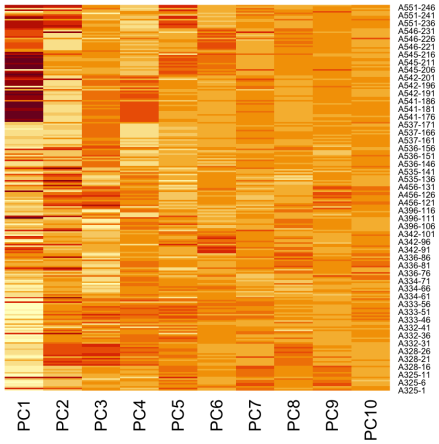

Figure 18: Heatmap for the principal components for the PCA decomposition with $K = 10$ components. Colour palette does not match that in Figure 12.

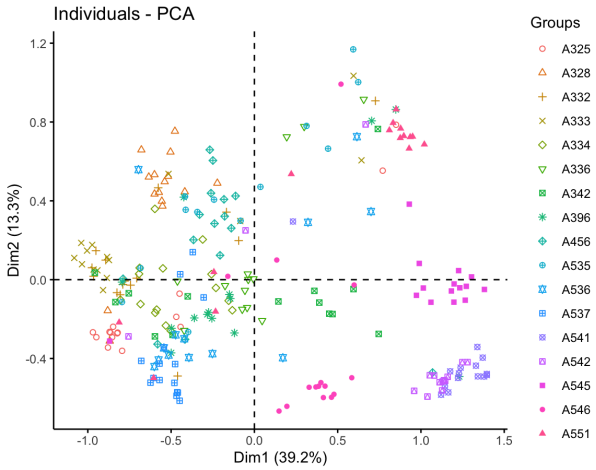

Figure 19: First two principal components. True families are indicated by different symbols.

running variational inference took under 10 minute. PCA was very much faster.

12 Recap

NMF admits a Bayesian reformulation. This Bayesian reformulation offers some advantages over PCA as a tool for dimensionality reduction and exploratory analysis. The Bayesian NMF can be implemented through ADVI, but computationally this is more demanding than performing PCA based on SVD.

References

C. M. Bishop. *Pattern recognition and machine learning.* Inf. Sci. Stat. New York, NY: Springer, 2006. ISBN 0-387-31073-8.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518): 859–877, 2017. doi: 10.1080/01621459.2017.1285773. URL `https://doi.org/10.1080/01621459.2017.1285773`.

B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li, and A. Riddell. Stan: a probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32, 2017. doi: 10.18637/jss.v076.i01. URL `https://www.jstatsoft.org/index.php/jss/article/view/v076i01`.

C. M. Carvalho, N. G. Polson, and J. G. Scott. Handling sparsity via the horseshoe. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 73–80, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL `https://proceedings.mlr.press/v5/carvalho09a.html`.

A. T. Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*, 2009:785152, 2009. doi: 10.1155/2009/785152. URL `https://doi.org/10.1155/2009/785152`.

C. Ding, X. He, and H. D. Simon. On the equivalence of Nonnegative Matrix Factorization and spectral clustering. In H. Kargupta, C. Kamath, J. Srivastava, and A. Goodman, editors, *Proceedings of the 2005 SIAM International Conference on Data Mining (SDM)*, pages 606–610, Philadelphia, PA, 2005. Society for Industrial and Applied Mathematics. doi: 10.1137/1.9781611972757.70. URL `https://epubs.siam.org/doi/abs/10.1137/1.9781611972757.70`.

D. J. Field. What is the goal of sensory coding? *Neural Computation*, 6(4): 559–601, 07 1994. ISSN 0899-7667. doi: 10.1162/neco.1994.6.4.559. URL `https://doi.org/10.1162/neco.1994.6.4.559`.

S. J. Godsill. On the relationship between Markov Chain Monte Carlo methods for model uncertainty. *Journal of Computational and Graphical Statistics*, 10(2):230–248, 2001. doi: 10.1198/10618600152627924. URL https://doi.org/10.1198/10618600152627924.

P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 12 1995. ISSN 0006-3444. doi: 10.1093/biomet/82.4.711. URL https://doi.org/10.1093/biomet/82.4.711.

P. O. Hoyer. Non-negative Matrix Factorization with sparseness constraints. *J. Mach. Learn. Res.*, 5:1457–1469, dec 2004. ISSN 1532-4435.

A. Kucukelbir, R. Ranganath, A. Gelman, and D. Blei. Automatic variational inference in Stan. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/352fe25daf686bdb4edca223c921acea-Paper.pdf.

A. Kucukelbir, D. Tran, R. Ranganath, A. Gelman, and D. M. Blei. Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(14):1–45, 2017. URL http://jmlr.org/papers/v18/16-107.html.

S. Kumar, C. Molloy, P. Muñoz, H. Daetwyler, D. Chagné, and R. Volz. Genome-enabled estimates of additive and nonadditive genetic variances and prediction of apple phenotypes across environments. *G3 Genes—Genomes—Genetics*, 5(12):2711–2718, 12 2015. ISSN 2160-1836. doi: 10.1534/g3.115.021105. URL https://doi.org/10.1534/g3.115.021105.

D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999. doi: 10.1038/44565. URL https://doi.org/10.1038/44565.

F. Lindsten and T. B. Schön. Backward Simulation Methods for Monte Carlo Statistical Inference. *Foundations and Trends™ in Machine Learning*, 6(1): 1–143, 2013. ISSN 1935-8237. doi: 10.1561/2200000045. URL http://dx.doi.org/10.1561/2200000045.

D. J. C. MacKay. Probable networks and plausible predictions-a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, 6(3):469, aug 1995. doi: 10.1088/0954-898X/6/3/011. URL https://dx.doi.org/10.1088/0954-898X/6/3/011.

D. J. C. MacKay. *Information theory, inference and learning algorithms*. Cambridge: Cambridge University Press, 2003. ISBN 0-521-64298-1.

A. Mnih and R. R. Salakhutdinov. Probabilistic matrix factorization. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural*

Information Processing Systems, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/d7322ed717dedf1eb4e6e52a37ea7bcd-Paper.pdf.

K. P. Murphy. *Probabilistic machine learning. An introduction*. Adapt. Comput. Mach. Learn. Cambridge, MA: MIT Press, 2022. ISBN 978-0-262-04682-4. URL mitpress.mit.edu/books/probabilistic-machine-learning.

K. P. Murphy. *Probabilistic machine learning. Advanced topics*. Adapt. Comput. Mach. Learn. Cambridge, MA: MIT Press, 2023. ISBN 978-0-262-04843-9; 978-0-262-37600-6. URL probml.github.io/pml-book/book2.html.

B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997. ISSN 0042-6989. doi: https://doi.org/10.1016/S0042-6989(97)00169-7. URL https://www.sciencedirect.com/science/article/pii/S0042698997001697.

P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994. doi: https://doi.org/10.1002/env.3170050203. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/env.3170050203.

N. Patterson, A. L. Price, and D. Reich. Population structure and eigenanalysis. *PLOS Genetics*, 2(12):1–20, 12 2006. doi: 10.1371/journal.pgen.0020190. URL https://doi.org/10.1371/journal.pgen.0020190.

W. H. Richardson. Bayesian-based iterative method of image restoration. *J. Opt. Soc. Am.*, 62(1):55–59, Jan 1972. doi: 10.1364/JOSA.62.000055. URL https://opg.optica.org/abstract.cfm?URI=josa-62-1-55.

B. D. Ripley. *Pattern recognition and neural networks*. Cambridge: Cambridge Univ. Press, 1996. ISBN 0-521-46086-7.

Stan Development Team. *Stan modeling language user's guide and reference manual. Version 2.34. https://mc-stan.org*, 2024. URL https://mc-stan.org.

G. Strang. *Introduction to linear algebra.* Wellesley, MA: Wellesley-Cambridge Press, 5th edition edition, 2016. ISBN 978-0-9802327-7-6. URL math.mit.edu/~gs/linearalgebra/.

W. N. Venables and B. D. Ripley. *Modern applied statistics with S*. Stat. Comput. (Cham). New York, NY: Springer, 4th ed. edition, 2002. ISBN 0-387-95457-0; 978-1-4419-3190-0; 978-0-387-21824-3. doi: 10.1007/b97626.

T. Virtanen, A. Taylan Cemgil, and S. Godsill. Bayesian extensions to non-negative matrix factorisation for audio signal modelling. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1825–1828, 2008. doi: 10.1109/ICASSP.2008.4517987.

E. Štrumbelj, A. Bouchard-Côté, J. Corander, A. Gelman, H. Rue, L. Murray, H. Pesonen, M. Plummer, and A. Vehtari. Past, present and future of software for Bayesian inference. *Statistical Science*, 39(1):46 – 61, 2024. doi: 10.1214/23-STS907. URL https://doi.org/10.1214/23-STS907.