Journal of Econometrics and Statistics
Vol. 5, Issue 2, 2025, pp. 129-149

Bayesian Heteroscedastic Matrix Factorization

Leonara Alves Cesario da Silva[a] and Ralph dos Santos Silva[b]

[a] *Credsystem, Brazil*
[b] *Instituto de Matemática, Universidade Federal de Rio de Janeiro, Brazil*

ABSTRACT
This paper proposes a Bayesian heteroscedastic matrix factorization to deal with sparse and high dimensional data. This methodology uses techniques of latent factors, considered the state-of-the-art in recommender systems based on collaborative filtering models. The proposal is to include variations among users to accommodate divergent opinions on various items, which may or may not be more generous in their critiques of the products. For this reason, tailored priors are presented for the parameters to obtain greater scalability and a specific variation for each user. We compare the results, obtained with the probabilistic matrix factorization (PMF) and our Bayesian heteroscedastic matrix factorization (BHMF), in the Netflix and the MovieLens datasets, based on the root mean square error (RMSE).

KEYWORDS
Collaborative Filtering, Latent Factors, Recommender Systems, Sparse Datas

1. Introduction

In a reality where there is a growing increase in online sales, companies that leverage the digital age and adapt to these new types of consumers to profit are likely to sell more. With the popularization of this consumption, there is a noticeable increase in interaction between consumers and companies to determine if purchasing a particular product is worthwhile - through the assignment of certain reviews that identify whether the item would be a good buy or not. The result of these opinions could offer all kinds of companies an opportunity to see business prospects, as they specify unique characteristics of each customer regarding their experiences with the company.

If a company has specific information about all its customers' past purchases, one could conjecture the range of possibilities it would have if it could anticipate future purchases by its customers. The consequence of this is what will be referred to as recommender systems. These systems aim to synthesize information from each user and attempt to anticipate each customer's evaluations of a given product. Based on this, the process could preemptively recommend products that the user might be interested in before the customer even realizes they want that item.

With the increasing ease of obtaining user review data, it is important to develop specific analytical procedures that reasonably capture the obtained information and can help companies gain insights about their customers. Based on this, recommender

CONTACT Author[b]. Email: ralph@im.ufrj.br

Article History
Received : 8 January 2025; Revised : 7 February 2025; Accepted : 16 February 2025; Published : 28 June 2025

To cite this paper

Leonara Alves Cesario da Silva & Ralph dos Santos Silva (2025). Bayesian Heteroscedastic Matrix Factorization. *Journal of Econometrics and Statistics.* 5(2), 129-148.

systems can be classified into various categories: collaborative filtering, content-based, knowledge-based, demographic, and hybrid. Collaborative filtering methods use information from similar users or items to make recommendations, while content-based methods use knowledge about past consumed products (such as genre, description, actors, among others).

Despite the various strategies for recommending products to users, collaborative filtering models are more commonly used because they feature characteristics similar to those applied in machine learning in the context of classification. This fact justifies the use of established models from the literature for making recommendations, such as neural networks, support vector machines, decision trees, Bayesian models, etc. Notwithstanding, in the era of big data, there is increasing concern about the amount of available information, and in this case, commonly found models may not be efficient.

In the literature, there are several techniques proposed for analyzing review data. Nevertheless, most of these data do not have high dimensionality - millions of users rating thousands of products. Canny [2] developed a new collaborative filtering procedure that protects user privacy while also applying probabilistic factor analysis to handle missing data. The author used a linear approach that generally extends singular value decomposition (SVD) methods and linear regression. For his applications, the author employed the iterative method known as the Expectation-Maximization (EM) algorithm, which is suited for handling sparse data and simple recursive definitions that can be combined with his main idea: privacy.

Xian et al. [11] applied an extension of SVD, called SVD++. This method is characterized by the addition of implicit feedback information from users. They used this methodology with a focus on respecting user privacy, employing a terminology called differential privacy, which ensures that other people cannot discover personal characteristics of individuals in the database.

Gemulla et al. [4] constructed an algorithm to approximate large matrices—with millions of user ratings for millions of products, amounting to billions of non-empty elements. This new algorithm is a distributed extension of the method using stochastic gradient descent, a stochastic optimization algorithm. Thus, it has been shown that it is possible to handle web-scale matrices with rapid convergence and scalability.

A study by Devooght, Kourtellis and Mantrach [3] explained that since it is commonly assumed that the distribution of missing ratings is the same as that of observed ratings, they introduced a dynamic matrix factorization structure allowing for explicit priors for these unknown values. The uniqueness of their work lies in the fact that for new users or items, the factorization can be updated regardless of the size of the data, allowing for rapid recommendations for new users.

Kabbur, Ning and Karypis [6] highlighted that the effectiveness of recommendation methods declines for the top-k nearest users or items with increased data sparsity. Based on this, they proposed an item-based method to generate top-k recommendations that learns the similarity matrix based on latent factors. Results from different datasets with varying degrees of sparsity indicated that their method performs better compared to methodologies seen in the literature.

Shen et al. [10] constructed a new recommendation system that utilizes specific characteristics of the locations users have frequented in the past to make recommendations. The technique is called landmark recommendation and attempts to identify, through a unified classifier, preference styles based on domain adaptation, leveraging photos from sites in the source domain and reference images in the target domain. The detected styles are then used to learn users' best preferences and make recommendations. They highlight that personalized methodologies for each user are a highly

effective approach in real-world travel data applications.

The overall objective of this paper is to present a procedure for predicting user ratings of specific items using real, high-dimensional data, particularly when conventional mechanisms employed in the literature are not suitable. We develop a methodology for estimating user ratings for a predefined set of items using matrix factorization, specifically probabilistic matrix decomposition, Bayesian approaches to probabilistic matrix decomposition, and propose generalizations that include heteroscedasticity among items or users. Moreover, we assess the ability of the proposed method to more accurately predict user ratings using appropriately selected training and test data. Finally, we compare, using the same dataset, the performance of the probabilistic matrix factorization and our Bayesian heteroscedastic matrix factorization.

The remainder of this paper is structured as follows: Section 2 briefly reviews latent factor models, including the probabillistic matrix factorization and the Bayesian matrix factorization models. Moreover, it introduces the new Bayesian heterocedastic matrix factorization model. Section 3 brings two real data applications which are based on the Netflix and the MovieLens datasets. Section 4 concludes.

2. Latent factor models

With the development of technology, there is a growing interaction between users of a website on the internet and their opinions about the products consumed. In such cases, the seller may be interested in increasing sales based on these customer responses by employing techniques to evaluate the relevance of certain items to users. To this end, the merchant can leverage various data characteristics to make recommendations, such as similarities between users, similarities between items, and similarities between users and items. Here, "user" refers to the individual to whom the recommendation is provided, and "item" refers to the product that will be recommended to the user.

Let $\boldsymbol{\mathcal{R}} = [r_{ij}]$ be the $m \times n$ matrix of user-item ratings, where m is the number of users, n is the number of items, and r_{ij} is the rating given by the i-th user for item j. It is important to note that there may be missing data in any of the rows or columns of this matrix.

In the basic matrix factorization model, a way to decompose the rating matrix $\boldsymbol{\mathcal{R}}$ is by approximating it as the product of a matrix $\boldsymbol{U}$ $(m \times p)$ and a matrix $\boldsymbol{V}$ $(n \times p)$ such that:

$$\boldsymbol{\mathcal{R}} \approx \boldsymbol{U}\boldsymbol{V}^{\top}. \tag{1}$$

Each column of $\boldsymbol{U}$ (or $\boldsymbol{V}$) can be referred to as a latent vector or component, and each row of $\boldsymbol{U}$ (or $\boldsymbol{V}$) is classified as a latent factor. The approximation error is given by:

$$||\boldsymbol{\mathcal{R}} - \boldsymbol{U}\boldsymbol{V}^{\top}||^2,$$

where $||\cdot||$ denotes the Frobenius norm (squared). Therefore, the objective function corresponds to the sum of the squares of the entries in the resulting residual matrix $(\boldsymbol{\mathcal{R}} - \boldsymbol{U}\boldsymbol{V}^{\top})$:

$$e_{ij} = r_{ij} - \boldsymbol{\upsilon}_i^{\top}\boldsymbol{\nu}_j, \quad i = 1, 2, \ldots, m, \quad \text{and} \quad j = 1, 2, \ldots, n,$$

such that the smaller the value of this function, the better the previously indicated factorization ($\boldsymbol{\mathcal{R}} \approx \boldsymbol{U}\boldsymbol{V}^\top$) will fit. The i-th row, $\boldsymbol{\upsilon}_i = (\upsilon_{i1}, \ldots, \upsilon_{ip})^\top$, of $\boldsymbol{U}$ is called the "user factor" and contains p entries corresponding to the affinities of user i with the p concepts (factors) in the rating matrix. Similarly, each column, $\boldsymbol{\nu}_j = (\nu_{j1}, \ldots, \nu_{jp})^\top$, of $\boldsymbol{V}$ is referred to as the "item factor" and represents the affinity of item j with respect to the p concepts.

From Equation (1), it is observed that each rating r_{ij} in $\boldsymbol{\mathcal{R}}$ can be approximately represented as the dot product of the i-th user factor and the j-th item factor, described by:

$$r_{ij} \approx \boldsymbol{\upsilon}_i^\top \boldsymbol{\nu}_j = \sum_{s=1}^{p} \upsilon_{is} \times \nu_{js}. \tag{2}$$

Singular value decomposition (SVD) is a matrix factorization method where the columns of $\boldsymbol{U}$ and $\boldsymbol{V}$ are constrained to be mutually orthogonal [1]. This approach has the advantage that the concepts can be completely uncorrelated. Consider the case where a fully specified matrix is available. The rating matrix $\boldsymbol{\mathcal{R}}$ can be factorized using a truncated SVD with rank $p \ll \min\{m, n\}$ (assuming that the matrix $\boldsymbol{\mathcal{R}}$ is of full rank) as follows:

$$\boldsymbol{\mathcal{R}} = \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{K}^\top,$$

where $\boldsymbol{H}$, $\boldsymbol{\Delta}$, and $\boldsymbol{K}$ are matrices of size $m \times p$, $p \times p$, and $n \times p$, respectively.

The matrices $\boldsymbol{H}$ and $\boldsymbol{K}$ contain the p largest eigenvectors of $\boldsymbol{\mathcal{R}}\boldsymbol{\mathcal{R}}^\top$ and $\boldsymbol{\mathcal{R}}^\top\boldsymbol{\mathcal{R}}$, respectively, and the diagonal matrix $\boldsymbol{\Delta}$ contains the p eigenvalues. The eigenvectors provide information about the directions of item-item (or user-user) correlations in the ratings, and consequently, enable the representation of each user (or item) in a reduced number of dimensions [1].

The diagonal matrix $\boldsymbol{\Delta}$ can be incorporated into the user factors $\boldsymbol{H}$ or the item factors $\boldsymbol{K}$. By convention,

$$\boldsymbol{U} = \boldsymbol{H}\boldsymbol{\Delta} \quad \text{and} \quad \boldsymbol{V} = \boldsymbol{K}^\top.$$

Let the sum of squared residuals be defined by

$$S(\boldsymbol{U}, \boldsymbol{V}) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} \mathbb{I}_{ij} e_{ij}^2 = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} \mathbb{I}_{ij} \left(r_{ij} - \boldsymbol{\upsilon}_i^\top \boldsymbol{\nu}_j\right)^2,$$

where $\mathbb{I}_{ij}$ is the indicator variable that equals 1 if user i rated item j and 0 otherwise. Thus, the objective function can be formulated as an optimization problem over the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$, that is, $\min_{(\boldsymbol{U},\boldsymbol{V})} S(\boldsymbol{U}, \boldsymbol{V})$ such that subject to columns of $\boldsymbol{U}$ are mutually orthogonal and columns of $\boldsymbol{V}$ are also mutually orthogonal.

Since the previously described scenario deals with fully observed matrices, applying an iterative process is computationally expensive, with a complexity of $O(m^2n+mn^2+n^3)$ for an $m\times n$ matrix [5]. Therefore, this approach can be modified to an optimization problem with regularization [1, 9]. This treatment has the advantage of preventing overfitting by adding a bias to the model, favoring simplicity over complexity. Thus,

the incorporation of regularization is given by

$$\min_{(\boldsymbol{U},\boldsymbol{V})} \left[S(\boldsymbol{U},\boldsymbol{V}) + \frac{\lambda_{\boldsymbol{\upsilon}}}{2} \sum_{i=1}^{m} \| \boldsymbol{\upsilon}_i \|^2 + \frac{\lambda_{\boldsymbol{\nu}}}{2} \sum_{j=1}^{n} \| \boldsymbol{\nu}_j \|^2 \right], \tag{3}$$

with the same restrictions over $\boldsymbol{U}$ and $\boldsymbol{V}$ as before.

2.1. *Probabilistic matrix factorization*

The probabilistic matrix factorization (PMF), proposed by Salakhutdinov and Mnih [8], is a linear probabilistic model with normally distributed observed errors, where the distribution of $\boldsymbol{\mathcal{R}}$ conditional on $\boldsymbol{U}$ and $\boldsymbol{V}$ (likelihood function) and the prior distributions on $\boldsymbol{U}$ and $\boldsymbol{V}$ are given by

$$\begin{aligned} p(\boldsymbol{\mathcal{R}}|\boldsymbol{U},\boldsymbol{V},\alpha) &= \prod_{i=1}^{m}\prod_{j=1}^{n} \left[\phi_1(r_{ij}|\boldsymbol{\upsilon}_i^\top \boldsymbol{\nu}_j, \alpha^{-1})\right]^{\mathbb{I}_{ij}} \\ p(\boldsymbol{U}|\alpha_{\boldsymbol{\upsilon}}) &= \prod_{i=1}^{m} \phi_p(\boldsymbol{\upsilon}_i|0, \alpha_{\boldsymbol{\upsilon}}^{-1}\mathbf{I}) \\ p(\boldsymbol{V}|\alpha_{\boldsymbol{\nu}}) &= \prod_{j=1}^{n} \phi_p(\boldsymbol{\nu}_j|0, \alpha_{\boldsymbol{\nu}}^{-1}\mathbf{I}), \end{aligned} \tag{4}$$

where $\phi_p(\cdot|\boldsymbol{\mu}, \boldsymbol{\Omega}^{-1})$ denotes the density function of a p-variate normal distribution with mean vector $\boldsymbol{\mu}$ and precision matrix $\boldsymbol{\Omega}$, for $p = 1, 2, \ldots, \min\{m, n\}$.

Note that the structure of the model assumes that all components of $\boldsymbol{U}$, and similarly those of $\boldsymbol{V}$, are a priori independent with the same precision structure controlled by $\alpha_{\boldsymbol{\upsilon}}^{-1}$ (and $\alpha_{\boldsymbol{\nu}}^{-1}$). Thus, learning in this model is achieved by maximizing the log-posterior conditional on the user and item factors with fixed hyperparameters (α, $\alpha_{\boldsymbol{\upsilon}}$, $\alpha_{\boldsymbol{\nu}}$):

$$\begin{aligned} \ln p(\boldsymbol{U},\boldsymbol{V}|\boldsymbol{\mathcal{R}},\alpha,\alpha_{\boldsymbol{\upsilon}},\alpha_{\boldsymbol{\nu}}) &= \ln p(\boldsymbol{\mathcal{R}}|\boldsymbol{U},\boldsymbol{V},\alpha) + \ln p(\boldsymbol{U}|\alpha_{\boldsymbol{\upsilon}}) + \ln p(\boldsymbol{V}|\alpha_{\boldsymbol{\nu}}) \\ &= \eta - \frac{\alpha}{2} \sum_{i=1}^{m}\sum_{j=1}^{n} \mathbb{I}_{ij}(r_{ij} - \boldsymbol{\upsilon}_i^\top \boldsymbol{\nu}_j)^2 \\ &- \frac{\alpha_{\boldsymbol{\upsilon}}}{2} \sum_{i=1}^{m} \boldsymbol{\upsilon}_i^\top \boldsymbol{\upsilon}_i - \frac{\alpha_{\boldsymbol{\nu}}}{2} \sum_{j=1}^{n} \boldsymbol{\nu}_j^\top \boldsymbol{\nu}_j \\ &- \frac{1}{2} \left[\left(\sum_{i=1}^{m}\sum_{j=1}^{n} \mathbb{I}_{ij} \right) \ln \alpha^{-1} + mp \ln \alpha_{\boldsymbol{\upsilon}}^{-1} + np \ln \alpha_{\boldsymbol{\nu}}^{-1} \right], \end{aligned}$$

where η is a constant as well as the last term of the expression.

According to Salakhutdinov and Mnih [8] and Salakhutdinov and Mnih [7], maximizing this expression is equivalent to minimizing the sum of squared errors in the objective function with quadratic regularization terms given in Equation (3), where $\lambda_{\boldsymbol{\upsilon}} = \alpha_{\boldsymbol{\upsilon}}/\alpha$ and $\lambda_{\boldsymbol{\nu}} = \alpha_{\boldsymbol{\nu}}/\alpha$.

2.2. *Bayesian matrix factorization*

The PMF model has the drawback of not automatically tuning the parameters $\lambda_{\boldsymbol{\upsilon}}$, $\lambda_{\boldsymbol{\nu}}$, and α. In this case, it is useful to employ another methodology capable of automatically adjusting the complexity of the model using a fully Bayesian technique. This method is called Bayesian matrix factorization (BMF), as presented by Salakhutdinov and Mnih [7], which uses the same likelihood function proposed by PMF in Equation (4), but with normal prior distributions on the user factor matrix $\boldsymbol{U}$ and the item factor matrix $\boldsymbol{V}$, expressed as:

$$p(\boldsymbol{U}|\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}) = \prod_{i=1}^{m} \phi_p(\boldsymbol{\upsilon}_i|\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}^{-1}) \quad \text{and} \quad p(\boldsymbol{V}|\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}) = \prod_{j=1}^{n} \phi_p(\boldsymbol{\nu}_j|\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}^{-1}).$$

The most notable distinction of this proposal from the previous PMF model is the generalization, allowing the components of the vector $\boldsymbol{\upsilon}_i$ (and also $\boldsymbol{\nu}_j$) to be correlated through a full precision matrix (full covariance matrix) and to have prior means different from zero. Nonetheless, it is worth noting that the covariance structure is the same between the columns of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively. That is, it is assumed that a priori the users (and also the items) can be represented with fixed latent covariance structures.

Furthermore, a normal-Wishart prior distributions have been assigned to the parameters for users $\boldsymbol{\Theta}_{\boldsymbol{\upsilon}} = \{\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}\}$ and for items $\boldsymbol{\Theta}_{\boldsymbol{\nu}} = \{\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}\}$, as follows:

$$\begin{aligned} p(\boldsymbol{\Theta}_{\boldsymbol{\upsilon}}|\boldsymbol{\Theta}_0) &= p(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}|\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}})p(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}) = \phi_p(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}|\boldsymbol{a}_{\boldsymbol{\upsilon}}, (b_{\boldsymbol{\upsilon}}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}})^{-1})\omega(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|\boldsymbol{C}_{\boldsymbol{\upsilon}}, d_{\boldsymbol{\upsilon}}) \\ p(\boldsymbol{\Theta}_{\boldsymbol{\nu}}|\boldsymbol{\Theta}_0) &= p(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{\Lambda}_{\boldsymbol{\nu}})p(\boldsymbol{\Lambda}_{\boldsymbol{\nu}}) = \phi_p(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{a}_{\boldsymbol{\nu}}, (b_{\boldsymbol{\nu}}\boldsymbol{\Lambda}_{\boldsymbol{\nu}})^{-1})\omega(\boldsymbol{\Lambda}_{\boldsymbol{\nu}}|\boldsymbol{C}_{\boldsymbol{\nu}}, d_{\boldsymbol{\nu}}), \end{aligned}$$

where ω represents the density function of a random matrix with a Wishart distribution with ψ_0 degrees of freedom and a scale matrix $\mathbf{W}_0$ $(p \times p)$. It can be expressed as $(\boldsymbol{\Lambda}|\mathbf{W}_0, \psi_0) \sim \mathcal{W}(\mathbf{W}_0, \psi_0)$. In this paper, $\boldsymbol{\Theta}_0$ represents the set of hyperparameters.

Moreover, the predictive distribution of the rating $\boldsymbol{\mathcal{R}}^* = [r_{ij}^*]$ for the i-th user rating the j-th item is obtained by marginalizing the model parameters:

$$p(r_{ij}^*|\boldsymbol{\mathcal{R}}, \boldsymbol{\Theta}_0) = \int p(r_{ij}^*|\boldsymbol{\upsilon}_i, \boldsymbol{\nu}_j)p(\boldsymbol{U}, \boldsymbol{V}|\boldsymbol{\mathcal{R}}, \boldsymbol{\Theta}_{\boldsymbol{\upsilon}}, \boldsymbol{\Theta}_{\boldsymbol{\nu}})p(\boldsymbol{\Theta}_{\boldsymbol{\upsilon}}, \boldsymbol{\Theta}_{\boldsymbol{\nu}}|\boldsymbol{\Theta}_0)d\{\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\Theta}_{\boldsymbol{\upsilon}}, \boldsymbol{\Theta}_{\boldsymbol{\nu}}\}. \quad (5)$$

Since the exact value of this predictive distribution is analytically intractable due to the complexity of the posterior, inferential approximations are needed. As a result, Salakhutdinov and Mnih [7] used Markov chain Monte Carlo (MCMC) methods, specifically Gibbs sampler. Basically, MCMC methods use Monte Carlo approximation for the predictive distribution in Equation (5), expressed as:

$$p(r_{ij}^*|\boldsymbol{\mathcal{R}}, \boldsymbol{\Theta}_0) \approx \frac{1}{K}\sum_{k=1}^{K} p(r_{ij}^*|\boldsymbol{\upsilon}_i^{(k)}, \boldsymbol{\nu}_j^{(k)}),$$

where samples $\{\boldsymbol{\upsilon}_i^{(k)}, \boldsymbol{\nu}_j^{(k)}\}$ are generated from the full conditional distributions of the current values of all variables (Gibbs sampler).

The choice of this approach is related to the use of normal distributions for the user and item factors, as their respective full conditional distributions, given the hyperparameter values $\boldsymbol{\Theta}_0$ and the matrix $\boldsymbol{\mathcal{R}}$, are also normal (see Appendix A for our general case):

- Full conditional: $(\boldsymbol{\upsilon}_i|\boldsymbol{\mathcal{R}}, \boldsymbol{V}, \boldsymbol{\Theta}_{\boldsymbol{\upsilon}}, \alpha) \sim \mathcal{N}(\boldsymbol{\mu}^*_{\boldsymbol{\upsilon}_i}, [\boldsymbol{\Lambda}^*_{\boldsymbol{\upsilon}_i}]^{-1})$, where

$$\boldsymbol{\Lambda}^*_{\boldsymbol{\upsilon}_i} = \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij}[\boldsymbol{\nu}_j \boldsymbol{\nu}_j^\top] \quad \text{and} \quad \boldsymbol{\mu}^*_{\boldsymbol{\upsilon}_i} = [\boldsymbol{\Lambda}^*_{\boldsymbol{\upsilon}_i}]^{-1} \left(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} \boldsymbol{\mu}_{\boldsymbol{\upsilon}} + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij}[\boldsymbol{\nu}_j r_{ij}] \right);$$

- Full conditional: $(\boldsymbol{\nu}_j|\boldsymbol{\mathcal{R}}, \boldsymbol{U}, \boldsymbol{\Theta}_{\boldsymbol{\nu}}, \alpha) \sim \mathcal{N}(\boldsymbol{\mu}^*_{\boldsymbol{\nu}_j}, [\boldsymbol{\Lambda}^*_{\boldsymbol{\nu}_j}]^{-1})$, where

$$\boldsymbol{\Lambda}^*_{\boldsymbol{\nu}_j} = \boldsymbol{\Lambda}_{\boldsymbol{\nu}} + \alpha \sum_{i=1}^{m} \mathbb{I}_{ij}[\boldsymbol{\upsilon}_i \boldsymbol{\upsilon}_i^\top] \quad \text{and} \quad \boldsymbol{\mu}^*_{\boldsymbol{\nu}_j} = [\boldsymbol{\Lambda}^*_{\boldsymbol{\nu}_j}]^{-1} \left(\boldsymbol{\Lambda}_{\boldsymbol{\nu}} \boldsymbol{\mu}_{\boldsymbol{\nu}} + \alpha \sum_{i=1}^{m} \mathbb{I}_{ij}[\boldsymbol{\upsilon}_i r_{ij}] \right);$$

- Full conditional: $(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|\boldsymbol{U}, \boldsymbol{\Theta}_0) \sim \mathcal{NW}(\boldsymbol{a}^*_{\boldsymbol{\upsilon}}, [b^*_{\boldsymbol{\upsilon}} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}]^{-1}, \boldsymbol{C}^*_{\boldsymbol{\upsilon}}, d^*_{\boldsymbol{\upsilon}})$, where

$$b^*_{\boldsymbol{\upsilon}} = b_{\boldsymbol{\upsilon}} + m, \quad d^*_{\boldsymbol{\upsilon}} = d_{\boldsymbol{\upsilon}} + m, \quad \boldsymbol{a}^*_{\boldsymbol{\upsilon}} = (m\overline{\boldsymbol{\upsilon}} + b_{\boldsymbol{\upsilon}} \boldsymbol{a}_{\boldsymbol{\upsilon}})/b^*_{\boldsymbol{\upsilon}}, \quad \overline{\boldsymbol{Q}}_{\boldsymbol{\upsilon}} = \sum_{i=1}^{m} (\boldsymbol{\upsilon}_i - \overline{\boldsymbol{\upsilon}})(\boldsymbol{\upsilon}_i - \overline{\boldsymbol{\upsilon}})^\top,$$

$$\overline{\boldsymbol{Q}}_a = m b_{\boldsymbol{\upsilon}} (\boldsymbol{a}_{\boldsymbol{\upsilon}} - \overline{\boldsymbol{\upsilon}})(\boldsymbol{a}_{\boldsymbol{\upsilon}} - \overline{\boldsymbol{\upsilon}})^\top / b^*_{\boldsymbol{\upsilon}}, \quad [\boldsymbol{C}^*_{\boldsymbol{\upsilon}}]^{-1} = \boldsymbol{C}^{-1}_{\boldsymbol{\upsilon}} + \overline{\boldsymbol{Q}}_{\boldsymbol{\upsilon}} + \overline{\boldsymbol{Q}}_a, \quad \overline{\boldsymbol{\upsilon}} = (1/m) \sum_{i=1}^{m} \boldsymbol{\upsilon}_i;$$

- Full conditional for: $p(\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}|\boldsymbol{V}, \boldsymbol{\Theta}_0) \sim \mathcal{NW}(\boldsymbol{a}^*_{\boldsymbol{\nu}}, [b^*_{\boldsymbol{\nu}} \boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}, \boldsymbol{C}^*_{\boldsymbol{\nu}}, d^*_{\boldsymbol{\nu}})$, where

$$b^*_{\boldsymbol{\nu}} = b_{\boldsymbol{\nu}} + n, \quad d^*_{\boldsymbol{\nu}} = d_{\boldsymbol{\nu}} + n, \quad \boldsymbol{a}^*_{\boldsymbol{\nu}} = (n\overline{\boldsymbol{\nu}} + b_{\boldsymbol{\nu}} \boldsymbol{a}_{\boldsymbol{\nu}})/b^*_{\boldsymbol{\nu}}, \quad \overline{\boldsymbol{Q}}_{\boldsymbol{\nu}} = \sum_{j=1}^{n} (\boldsymbol{\nu}_j - \overline{\boldsymbol{\nu}})(\boldsymbol{\nu}_j - \overline{\boldsymbol{\nu}})^\top,$$

$$\overline{\boldsymbol{Q}}_b = n b_{\boldsymbol{\nu}} (\boldsymbol{a}_{\boldsymbol{\nu}} - \overline{\boldsymbol{\nu}})(\boldsymbol{a}_{\boldsymbol{\nu}} - \overline{\boldsymbol{\nu}})^\top / b^*_{\boldsymbol{\nu}}, \quad [\boldsymbol{C}^*_{\boldsymbol{\nu}}]^{-1} = \boldsymbol{C}^{-1}_{\boldsymbol{\nu}} + \overline{\boldsymbol{Q}}_{\boldsymbol{\nu}} + \overline{\boldsymbol{Q}}_b, \quad \overline{\boldsymbol{\nu}} = (1/n) \sum_{j=1}^{n} \boldsymbol{\nu}_j.$$

Hyperparameter α

In the BMF methodology, the parameter α is treated as the precision error of the observations and it is fixed at the value of 2. Nevertheless, a prior can be applied to this parameter and included in the Gibbs sampler for estimation. It is assigned as $(\alpha|a_\alpha, b_\alpha) \sim \mathcal{G}(a_\alpha, b_\alpha)$, that is, a gamma distribution with mean a_α/b_α. Thus,

- Full conditional for: $(\alpha|\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\mathcal{R}}) \sim \mathcal{G}\,(a^\star_\alpha, b^\star_\alpha)$, where $a^\star_\alpha = \dfrac{N}{2} + a_\alpha$ with $N = \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbb{I}_{ij}$, and $b^\star_\alpha = \dfrac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbb{I}_{ij} (r_{ij} - \boldsymbol{\upsilon}_i^\top \boldsymbol{\nu}_j)^2 + b_\alpha$.

2.3. *Bayesian heteroscedastic matrix factorization*

In the BMF procedure, the factors for users (and items) can be correlated, but the variance for all users (and all items) is the same. This suggests that there are no discrepancies considered between user and item ratings. For example, it is not considered that one user might rate within a scale of 1 to 2, while a second user might rate within a scale of 2 to 5. Similarly, for items, a highly rated item might be classified only from 4 onwards, while a poorly rated item might be limited to a scale from 1 to 3.

To address this more generally, we propose a Bayesian heteroscedastic matrix factorization model (BHMF), which extends the BMF by adding a parameter $\lambda_{\boldsymbol{\upsilon}_i}$ for

each user and $\lambda_{\boldsymbol{\nu}_j}$ for each item. This new model is novel and it can presented as follows:

$$\begin{aligned} p(\boldsymbol{\mathcal{R}}|\boldsymbol{U},\boldsymbol{V},\alpha) &= \prod_{i=1}^{m}\prod_{j=1}^{n}\left[\phi_1(r_{ij}|\boldsymbol{\upsilon}_i^\top\boldsymbol{\nu}_j,\alpha^{-1})\right]^{\mathbb{I}_{ij}}, \\ p(\boldsymbol{U}|\boldsymbol{\mu}_{\boldsymbol{\upsilon}},\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}},\boldsymbol{\lambda}_{\boldsymbol{\upsilon}}) &= \prod_{i=1}^{m}\phi_p(\boldsymbol{\upsilon}_i|\boldsymbol{\mu}_{\boldsymbol{\upsilon}},[\lambda_{\boldsymbol{\upsilon}_i}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}]^{-1}), \\ p(\boldsymbol{V}|\boldsymbol{\mu}_{\boldsymbol{\nu}},\boldsymbol{\Lambda}_{\boldsymbol{\nu}},\boldsymbol{\lambda}_{\boldsymbol{\nu}}) &= \prod_{j=1}^{n}\phi_p(\boldsymbol{\nu}_j|\boldsymbol{\mu}_{\boldsymbol{\nu}},[\lambda_{\boldsymbol{\nu}_j}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}), \\ p(\alpha|a_\alpha,b_\alpha) &= \gamma(\alpha|a_\alpha,b_\alpha), \end{aligned}$$

where $\boldsymbol{\lambda}_{\boldsymbol{\upsilon}} = (\lambda_{\boldsymbol{\upsilon}_1},\ldots,\lambda_{\boldsymbol{\upsilon}_m})$ and $\boldsymbol{\lambda}_{\boldsymbol{\nu}} = (\lambda_{\boldsymbol{\nu}_1},\ldots,\lambda_{\boldsymbol{\nu}_n})$, whilst $\gamma(.|a_\alpha,b_\alpha)$ denotes de probability density funcion of a gamma random variable with mean a_α/b_α.

Additionally, a gamma prior distribution is assigned to the user parameters, $\lambda_{\boldsymbol{\upsilon}_i}$, and item parameters, $\lambda_{\boldsymbol{\nu}_j}$, as well as to the parameters $\kappa_{\boldsymbol{\upsilon}}$ and $\kappa_{\boldsymbol{\nu}}$, which refer to the rate and shape parameters of the gamma distribution for $\lambda_{\boldsymbol{\upsilon}_i}$ and $\lambda_{\boldsymbol{\nu}_j}$, respectively:

$$p(\boldsymbol{\lambda}_{\boldsymbol{\upsilon}}|\kappa_{\boldsymbol{\upsilon}}) = \prod_{i=1}^{m}p(\lambda_{\boldsymbol{\upsilon}_i}|\kappa_{\boldsymbol{\upsilon}}) = \prod_{i=1}^{m}\left[\gamma(\lambda_{\boldsymbol{\upsilon}_i}|\frac{\kappa_{\boldsymbol{\upsilon}}}{2},\frac{\kappa_{\boldsymbol{\upsilon}}}{2})\right], \quad p(\kappa_{\boldsymbol{\upsilon}}) = \gamma(\kappa_{\boldsymbol{\upsilon}}|a_{\kappa_{\boldsymbol{\upsilon}}},b_{\kappa_{\boldsymbol{\upsilon}}}),$$

$$p(\boldsymbol{\lambda}_{\boldsymbol{\nu}}|\kappa_{\boldsymbol{\nu}}) = \prod_{j=1}^{n}p(\lambda_{\boldsymbol{\nu}_j}|\kappa_{\boldsymbol{\nu}}) = \prod_{j=1}^{n}\left[\gamma(\lambda_{\boldsymbol{\nu}_j}|\frac{\kappa_{\boldsymbol{\nu}}}{2},\frac{\kappa_{\boldsymbol{\nu}}}{2})\right], \quad p(\kappa_{\boldsymbol{\nu}}) = \gamma(\kappa_{\boldsymbol{\nu}}|a_{\kappa_{\boldsymbol{\nu}}},b_{\kappa_{\boldsymbol{\nu}}}).$$

An important highlight is that modeling in this way considers $\lambda_{\boldsymbol{\upsilon}_i}$ and $\lambda_{\boldsymbol{\nu}_j}$ as auxiliary variables in the scale mixture of the normal distribution with the gamma distribution to obtain the marginal Student's t-distribution.

The normal-Wishart prior distributions for the user parameters $\boldsymbol{\Theta}_{\boldsymbol{\upsilon}} = \{\boldsymbol{\mu}_{\boldsymbol{\upsilon}},\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}\}$ and item parameters $\boldsymbol{\Theta}_{\boldsymbol{\nu}} = \{\boldsymbol{\mu}_{\boldsymbol{\nu}},\boldsymbol{\Lambda}_{\boldsymbol{\nu}}\}$ have been updated to

$$\begin{aligned} p(\boldsymbol{\Theta}_{\boldsymbol{\upsilon}}|\boldsymbol{\Theta}_0) &= p(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}|\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}})p(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}) = \phi_p(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}|\boldsymbol{a}_{\boldsymbol{\upsilon}},[b_{\boldsymbol{\upsilon}}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}]^{-1})\omega(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|\boldsymbol{C}_{\boldsymbol{\upsilon}},d_{\boldsymbol{\upsilon}}) \\ p(\boldsymbol{\Theta}_{\boldsymbol{\nu}}|\boldsymbol{\Theta}_0) &= p(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{\Lambda}_{\boldsymbol{\nu}})p(\boldsymbol{\Lambda}_{\boldsymbol{\nu}}) = \phi_p(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{a}_{\boldsymbol{\nu}},[b_{\boldsymbol{\nu}}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1})\omega(\boldsymbol{\Lambda}_{\boldsymbol{\nu}}|\boldsymbol{C}_{\boldsymbol{\nu}},d_{\boldsymbol{\nu}}). \end{aligned}$$

Moreover, the predictive distribution of the rating $\boldsymbol{\mathcal{R}}^* = [r_{ij}^*]$ for the i-th user who evaluated item j is obtained by marginalizing the model parameters:

$$\begin{aligned} p(r_{ij}^*|\boldsymbol{\mathcal{R}},\boldsymbol{\Theta}_0) &= \int p(r_{ij}^*|\boldsymbol{\upsilon}_i,\boldsymbol{\nu}_j,\alpha)p(\boldsymbol{U},\boldsymbol{V}|\boldsymbol{\mathcal{R}},\boldsymbol{\Theta}_{\boldsymbol{\upsilon}},\boldsymbol{\Theta}_{\boldsymbol{\nu}},\boldsymbol{\lambda}_{\boldsymbol{\upsilon}},\boldsymbol{\lambda}_{\boldsymbol{\nu}}) \\ &\times\ p(\boldsymbol{\Theta}_{\boldsymbol{\upsilon}},\boldsymbol{\Theta}_{\boldsymbol{\nu}}|\boldsymbol{\Theta}_0)p(\alpha)p(\boldsymbol{\lambda}_{\boldsymbol{\upsilon}},\boldsymbol{\lambda}_{\boldsymbol{\nu}})d\{\boldsymbol{U},\boldsymbol{V},\boldsymbol{\Theta}_{\boldsymbol{\upsilon}},\boldsymbol{\Theta}_{\boldsymbol{\nu}},\alpha,\boldsymbol{\lambda}_{\boldsymbol{\upsilon}},\boldsymbol{\lambda}_{\boldsymbol{\nu}}\}. \end{aligned}$$

Finally, the full conditionals for the BHMF model are described below (see Appendix A):

- Full conditional: $(\alpha|\boldsymbol{\mathcal{R}},\boldsymbol{U},\boldsymbol{V},a_\alpha,b_\alpha) \sim \mathcal{G}(a_\alpha^\star,b_\alpha^\star)$, where $a_\alpha^\star = \frac{N}{2} + a_\alpha$ with $N = \sum_{i=1}^{m}\sum_{j=1}^{n}\mathbb{I}_{ij}$, and $b_\alpha^\star = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbb{I}_{ij}(r_{ij} - \boldsymbol{\upsilon}_i^\top\boldsymbol{\nu}_j)^2 + b_\alpha$;
- Full conditional: $(\boldsymbol{\upsilon}_i|\boldsymbol{\mathcal{R}},\boldsymbol{V},\boldsymbol{\Theta}_{\boldsymbol{\upsilon}},\alpha) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^*,[\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^*]^{-1})$, where

$$\boldsymbol{\Lambda}^*_{\boldsymbol{\upsilon}_i} = \lambda_{\boldsymbol{\upsilon}_i}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} + \alpha\sum_{j=1}^{n}\mathbb{I}_{ij}[\boldsymbol{\nu}_j\boldsymbol{\nu}_j^\top] \quad \text{and} \quad \boldsymbol{\mu}^*_{\boldsymbol{\upsilon}_i} = \left[\boldsymbol{\Lambda}^*_{\boldsymbol{\upsilon}_i}\right]^{-1}\left(\lambda_{\boldsymbol{\upsilon}_i}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}\boldsymbol{\mu}_{\boldsymbol{\upsilon}} + \alpha\sum_{j=1}^{n}\mathbb{I}_{ij}[r_{ij}\boldsymbol{\nu}_j]\right);$$

- Full conditional: $(\boldsymbol{\nu}_j|\boldsymbol{\mathcal{R}}, \boldsymbol{U}, \boldsymbol{\Theta}_{\boldsymbol{\nu}}, \alpha) \sim \mathcal{N}(\boldsymbol{\mu}^*_{\boldsymbol{\nu}_j}, [\boldsymbol{\Lambda}^*_{\boldsymbol{\nu}_j}]^{-1})$, where

$$\boldsymbol{\Lambda}^*_{\boldsymbol{\nu}_j} = \lambda_{\boldsymbol{\nu}_j}\boldsymbol{\Lambda}_{\boldsymbol{\nu}} + \alpha\sum_{i=1}^{m}\mathbb{I}_{ij}[\boldsymbol{\upsilon}_i\boldsymbol{\upsilon}_i^\top], \quad \text{and} \quad \boldsymbol{\mu}^*_{\boldsymbol{\nu}_j} = \left[\boldsymbol{\Lambda}^*_{\boldsymbol{\nu}_j}\right]^{-1}\left(\lambda_{\boldsymbol{\nu}_j}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}\boldsymbol{\mu}_{\boldsymbol{\nu}} + \alpha\sum_{i=1}^{m}\mathbb{I}_{ij}[r_{ij}\boldsymbol{\upsilon}_i]\right);$$

- Full conditional: $(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|\boldsymbol{U}, \boldsymbol{\Theta}_0) \sim \mathcal{NW}(\boldsymbol{a}^*_{\boldsymbol{\upsilon}}, [b^*_{\boldsymbol{\upsilon}}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}]^{-1}, \boldsymbol{C}^*_{\boldsymbol{\upsilon}}, d^*_{\boldsymbol{\upsilon}})$,where

$$b^*_{\boldsymbol{\upsilon}} = b_{\boldsymbol{\upsilon}} + m\bar{\lambda}_{\boldsymbol{\upsilon}}, \quad d^*_{\boldsymbol{\upsilon}} = d_{\boldsymbol{\upsilon}} + m, \quad \boldsymbol{a}^*_{\boldsymbol{\upsilon}} = \frac{1}{b^*_{\boldsymbol{\upsilon}}}(m\bar{\lambda}_{\boldsymbol{\upsilon}}\tilde{\boldsymbol{\upsilon}} + b_{\boldsymbol{\upsilon}}\boldsymbol{a}_{\boldsymbol{\upsilon}}),$$

$$\tilde{\boldsymbol{Q}}_{\boldsymbol{\upsilon}} = \sum_{i=1}^{m}\lambda_{\boldsymbol{\upsilon}_i}(\boldsymbol{\upsilon}_i - \tilde{\boldsymbol{\upsilon}})(\boldsymbol{\upsilon}_i - \tilde{\boldsymbol{\upsilon}})^\top, \quad \tilde{\boldsymbol{Q}}_{\boldsymbol{a}_{\boldsymbol{\upsilon}}} = \frac{m\bar{\lambda}_{\boldsymbol{\upsilon}}b_{\boldsymbol{\upsilon}}}{b^*_{\boldsymbol{\upsilon}}}(\boldsymbol{a}_{\boldsymbol{\upsilon}} - \tilde{\boldsymbol{\upsilon}})(\boldsymbol{a}_{\boldsymbol{\upsilon}} - \tilde{\boldsymbol{\upsilon}})^\top,$$

$$[\boldsymbol{C}^*_{\boldsymbol{\upsilon}}]^{-1} = \boldsymbol{C}^{-1}_{\boldsymbol{\upsilon}} + \tilde{\boldsymbol{Q}}_{\boldsymbol{\upsilon}} + \tilde{\boldsymbol{Q}}_{\boldsymbol{a}_{\boldsymbol{\upsilon}}}, \quad \bar{\lambda}_{\boldsymbol{\upsilon}} = \frac{1}{m}\sum_{i=1}^{m}\lambda_{\boldsymbol{\upsilon}_i}, \quad \tilde{\boldsymbol{\upsilon}} = \frac{1}{m\bar{\lambda}_{\boldsymbol{\upsilon}}}\sum_{i=1}^{m}\lambda_{\boldsymbol{\upsilon}_i}\boldsymbol{\upsilon}_i.$$

- Full conditional: $(\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}|\boldsymbol{V}, \boldsymbol{\Theta}_0) \sim \mathcal{NW}(\boldsymbol{a}^*_{\boldsymbol{\nu}}, [b^*_{\boldsymbol{\nu}}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}, \boldsymbol{C}^*_{\boldsymbol{\nu}}, d^*_{\boldsymbol{\nu}})$, where

$$b^*_{\boldsymbol{\nu}} = b_{\boldsymbol{\nu}} + n\bar{\lambda}_{\boldsymbol{\nu}}, \quad d^*_{\boldsymbol{\nu}} = d_{\boldsymbol{\nu}} + n, \quad \boldsymbol{a}^*_{\boldsymbol{\nu}} = \frac{1}{b^*_{\boldsymbol{\nu}}}(n\bar{\lambda}_{\boldsymbol{\nu}}\tilde{\boldsymbol{\nu}} + b_{\boldsymbol{\nu}}\boldsymbol{a}_{\boldsymbol{\nu}}),$$

$$\tilde{\boldsymbol{Q}}_{\boldsymbol{\nu}} = \sum_{j=1}^{n}\lambda_{\boldsymbol{\nu}_j}(\boldsymbol{\nu}_j - \tilde{\boldsymbol{\nu}})(\boldsymbol{\nu}_j - \tilde{\boldsymbol{\nu}})^\top, \quad \tilde{\boldsymbol{Q}}_{\boldsymbol{a}_{\boldsymbol{\nu}}} = \frac{n\bar{\lambda}_{\boldsymbol{\nu}}b_{\boldsymbol{\nu}}}{b^*_{\boldsymbol{\nu}}}(\boldsymbol{a}_{\boldsymbol{\nu}} - \tilde{\boldsymbol{\nu}})(\boldsymbol{a}_{\boldsymbol{\nu}} - \tilde{\boldsymbol{\nu}})^\top,$$

$$[\boldsymbol{C}^*_{\boldsymbol{\nu}}]^{-1} = \boldsymbol{C}^{-1}_{\boldsymbol{\nu}} + \tilde{\boldsymbol{Q}}_{\boldsymbol{\nu}} + \tilde{\boldsymbol{Q}}_{\boldsymbol{a}_{\boldsymbol{\nu}}}, \quad \bar{\lambda}_{\boldsymbol{\nu}} = \frac{1}{n}\sum_{j=1}^{n}\lambda_{\boldsymbol{\nu}_j}, \quad \tilde{\boldsymbol{\nu}} = \frac{1}{n\bar{\lambda}_{\boldsymbol{\nu}}}\sum_{j=1}^{n}\lambda_{\boldsymbol{\nu}_j}\boldsymbol{\nu}_j.$$

Just as with the prior distributions, the full conditional distributions of $\lambda_{\boldsymbol{\upsilon}_i}$ and $\lambda_{\boldsymbol{\nu}_j}$ (conditional on the set $\mathcal{F}$ comprising everything else) are also gamma distributions (see Appendix A):

- Full conditional: $(\lambda_{\boldsymbol{\upsilon}_i}|\mathcal{F}) \sim \mathcal{G}\left(\frac{D_{\boldsymbol{\upsilon}} + \kappa_{\boldsymbol{\upsilon}}}{2}, \frac{1}{2}\left[\kappa_{\boldsymbol{\upsilon}} + (\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})^T\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}(\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})\right]\right)$, where $D_{\boldsymbol{\upsilon}}$ is the dimension of the matrix $\boldsymbol{C}_{\boldsymbol{\upsilon}}$.
- Full conditional: $(\lambda_{\boldsymbol{\nu}_j}|\mathcal{F}) \sim \mathcal{G}\left(\frac{D_{\boldsymbol{\nu}} + \kappa_{\boldsymbol{\nu}}}{2}, \frac{1}{2}\left[\kappa_{\boldsymbol{\nu}} + (\boldsymbol{\nu}_j - \boldsymbol{\mu}_{\boldsymbol{\nu}})^T\boldsymbol{\Lambda}_{\boldsymbol{\nu}}(\boldsymbol{\nu}_j - \boldsymbol{\mu}_{\boldsymbol{\nu}})\right]\right)$, where $D_{\boldsymbol{\nu}}$ is the dimension of the matrix $\boldsymbol{C}_{\boldsymbol{\nu}}$.

Finally, we have the full conditional distributions of $\kappa_{\boldsymbol{\upsilon}}$ and $\kappa_{\boldsymbol{\nu}}$. In this specific case, it is not possible to identify any known distribution (see Appendix A). Therefore,

- Full conditional:

$$p(\kappa_{\boldsymbol{\upsilon}}|\mathcal{F}) \propto \exp\left\{-\frac{\kappa_{\boldsymbol{\upsilon}}}{2}\left[\sum_{i=1}^{m}(\lambda_{\boldsymbol{\upsilon}_i} - \ln(\lambda_{\boldsymbol{\upsilon}_i})) - m\ln\left(\frac{\kappa_{\boldsymbol{\upsilon}}}{2}\right)\right] - m\ln\Gamma\left(\frac{\kappa_{\boldsymbol{\upsilon}}}{2}\right)\right\}p(\kappa_{\boldsymbol{\upsilon}});$$

- Full conditional:

$$p(\kappa_{\boldsymbol{\nu}}|\mathcal{F}) \propto \exp\left\{-\frac{\kappa_{\boldsymbol{\nu}}}{2}\left[\sum_{j=1}^{n}(\lambda_{\boldsymbol{\nu},j} - \ln(\lambda_{\boldsymbol{\nu}_j})) - n\ln\left(\frac{\kappa_{\boldsymbol{\nu}}}{2}\right)\right] - n\ln\Gamma\left(\frac{\kappa_{\boldsymbol{\nu}}}{2}\right)\right\} p(\kappa_{\boldsymbol{\nu}}).$$

Gibbs sampler, more generally, includes modeling the degrees of freedom $\kappa_{\boldsymbol{\upsilon}}$ and $\kappa_{\boldsymbol{\nu}}$. The complete conditionals do not have a known form but are easy to sample using other numerical methods, such as Metropolis-Hastings, which should be used to generate values for these conditionals. Nonetheless, the goal is simply to model heteroscedasticity and obtain a lower mean squared prediction error without the algorithm taking too long. Thus, at this stage, the degrees of freedom were fixed at a number between 3 and 10, which will be specified in the applications.

3. Applications

In this section, analyses for two datasets are presented: the Netflix and the MovieLens, due to their importance in the field. The standard values for the parameter p, the latent dimension, tested across all procedures are $5, 10, 20$, and 30. Generally, a higher value of p would lead to better predictions at the cost of increased computational expense. Additionally, all models were trained for 200 iterations or epochs, even though convergence may have been reached earlier in some cases. The programming language used was Python[1]. The maximum processing time for Netflix data was 8 hours for PMF (for all values of p) and approximately 36 hours for each value of p for BMF and BHMF. For MovieLens data, the processing time was 4.5 hours for PMF and around 16 hours for the total processing time of all tested situations for BMF and BHMF.

3.1. *Netflix*

The Netflix is a service for streaming videos of movies, TV series, etc., via the internet. The training data, collected from Netflix, consists of a collection of over 100 million ratings, ranging on a scale from 1 to 5, from a random sample of 480,189 anonymous users and 17,700 movies between October 1998 and December 2005. Additionally, approximately 1.5 million ratings were provided as supplemental material for model preparation. The goal is to predict new ratings for about 3 million users on the same set of movies based on this database. The provided training sample is subdivided into two categories: training data and test data (75% for training and 25% for testing) for evaluating model efficiency. The challenge lies in obtaining good results with this data, as it is extremely sparse, with approximately 99% of the entries consisting of zeros.

In PMF, we used the same approach as Salakhutdinov and Mnih [8]. In BMF, we set $\boldsymbol{a}_{\boldsymbol{\upsilon}} = \boldsymbol{a}_{\boldsymbol{\nu}} = \mathbf{0}$, $b_{\boldsymbol{\upsilon}} = b_{\boldsymbol{\nu}} = 2$, $d_{\boldsymbol{\upsilon}} = d_{\boldsymbol{\nu}} = p$, $\boldsymbol{C}_{\boldsymbol{\upsilon}} = \boldsymbol{C}_{\boldsymbol{\nu}}$ as the identity matrix, and $\alpha = 2$. In our BHMF, we set $\boldsymbol{a}_{\boldsymbol{\upsilon}} = \boldsymbol{a}_{\boldsymbol{\nu}} = \mathbf{0}$, $b_{\boldsymbol{\upsilon}} = b_{\boldsymbol{\nu}} = 2$, $d_{\boldsymbol{\upsilon}} = d_{\boldsymbol{\nu}} = p$, and $\boldsymbol{C}_{\boldsymbol{\upsilon}} = \boldsymbol{C}_{\boldsymbol{\nu}}$ set as the identity matrix. The hyperparameters for the prior distribution of α are defined as $a_\alpha = b_\alpha = 0.1$. Moreover, we fix $\kappa_{\boldsymbol{\upsilon}} = \kappa_{\boldsymbol{\nu}} = 5$.

Table 1 presents the root mean squared error (RMSE) results of the PMF, BMF, and our BHMF models for the Netflix data.

[1] https://www.python.org

Table 1. The RMSE obtained from PMF, BMF and our BHMF.

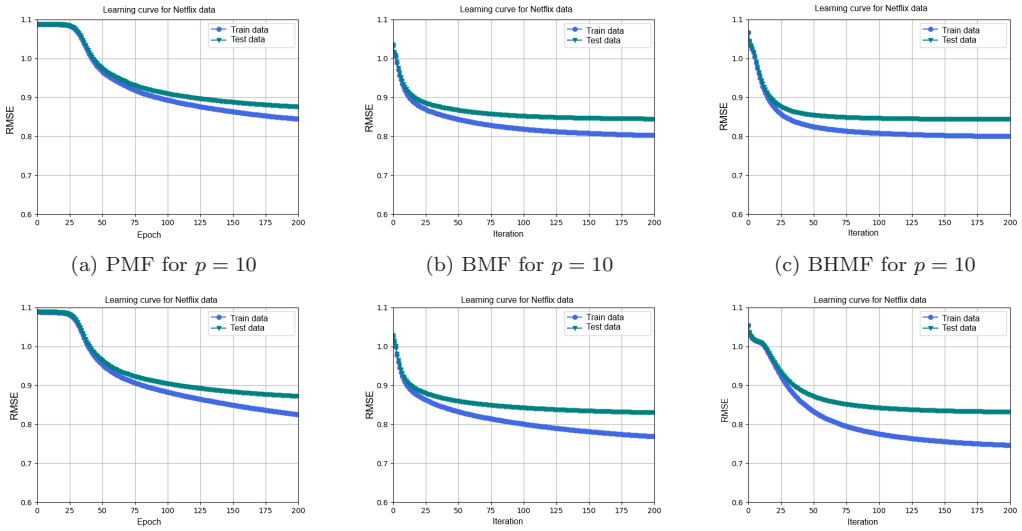
p	RMSE					
	Train data			Test data		
	PMF	BMF	BHMF	PMF	BMF	BHMF
5	0.8644	0.8344	0.8348	0.8864	0.8630	0.8630
10	0.8446	0.8026	0.7999	0.8755	0.8442	0.8431
20	0.8335	0.7774	0.7797	0.8732	0.8329	0.8356
30	0.8251	0.7688	0.7466	0.8719	0.8297	0.8315

Table 1 shows that increasing p reduces the mean squared error for all three models (PMF, BMF, and BHMF), confirming that a higher p leads to a lower RMSE. The BMF and BHMF models produced very similar results, with differences only in the third decimal place of the RMSE. Both models are thus considered suitable for Netflix data.

For training data, BHMF performed similarly to BMF across all p values, with the best result at $p = 30$. This was expected because BHMF is more robust (due to its use of the Student t-distribution with 5 degrees of freedom, which can better handle data discrepancies) compared to BMF. However, BHMF's theoretical variance is higher than BMF's, so BMF was expected to have a slightly better RMSE for test data, though the differences may be minor. Thus, BHMF might offer better RMSE adjustments at the possible cost of increased computational time.

The BMF and BHMF models outperformed PMF, which requires extensive hyper-parameter tuning and showed less improvement with higher p values. Consequently, BMF and BHMF at $p = 30$ provided the best results for both training and test data, while PMF at $p = 5$ had the worst performance, with RMSE not falling below 0.86.

Figure 1 displays the learning curves of the PMF, BMF and BHMF models on Netflix data with two p values: 10 and 30. The PMF model effectively reduced RMSE for both training and test data, though it took over 25 epochs to start decreasing



(a) PMF for $p = 10$ (b) BMF for $p = 10$ (c) BHMF for $p = 10$

(d) PMF for $p = 30$ (e) BMF for $p = 30$ (f) BHMF for $p = 30$

Figure 1. Learning curves of PMF, BMF, and BHMF for Netflix data at different values of p.

and remained above 0.80 even after 200 epochs. Extended training could potentially improve results. BMF also reduced RMSE for both datasets, with a rapid initial decrease in RMSE during the first 25 iterations. However, further reductions in test data were less significant, indicating that additional training might not be necessary. The BHMF model reduced RMSE similarly to BMF, with a high initial reduction but a slower decline as p increased. Further training had minimal impact on test data, suggesting limited benefit from extending iterations.

3.2. *MovieLens*

The GroupLens[2] research group collected and provided MovieLens datasets to aid film recommendation research. The data, which include over 10 million ratings on a 1 to 5 scale for 10,681 movies by 71,567 users, are very sparse (about 98.7% of the training set entries are zeros). The same training-test split provided by MovieLens was used.

For PMF, the best results were achieved with a mini-batch size of 100,000, a learning rate of $\epsilon = 5$, a gradient moment of 0.90, and a regularization parameter of $L_2 = 0.005$. For BMF, we used the following settings: $\boldsymbol{a_\upsilon} = \boldsymbol{a_\nu} = \mathbf{0}$, $b_\upsilon = b_\nu = 2$, $d_\upsilon = d_\nu = p$, $\boldsymbol{C_\upsilon} = \boldsymbol{C_\nu}$ as the identity matrix, and $\alpha = 2$. In BHMF, we again set $\boldsymbol{a_\upsilon} = \boldsymbol{a_\nu} = \mathbf{0}$, $b_\upsilon = b_\nu = 2$, $d_\upsilon = d_\nu = p$, and $\boldsymbol{C_\upsilon} = \boldsymbol{C_\nu}$ as the identity matrix. The prior distribution hyperparameters for α were set as $a_\alpha = b_\alpha = 0.1$, and we fixed $\kappa_\upsilon = \kappa_\nu = 5$. Initial values for $\boldsymbol{U}$ and $\boldsymbol{V}$ were set to 0.1, multiplied by random samples from a uniform distribution in $(0, 1)$.

Table 2 indicates that increasing p reduces the root mean squared error (RMSE) for all models (PMF, BMF, BHMF). Among these, BHMF achieved the best results on MovieLens data, especially with $p = 30$, showing RMSE below 0.80 for training and well below 0.90 for test data. This suggests BHMF is more effective at capturing data variability compared to PMF and BMF. On the other hand, PMF did not reach RMSE values below 0.79 (train data) despite various hyperparameter tests, and increasing p did not justify the computational cost. BMF performed between PMF and BHMF, making it a good candidate for further method comparisons.

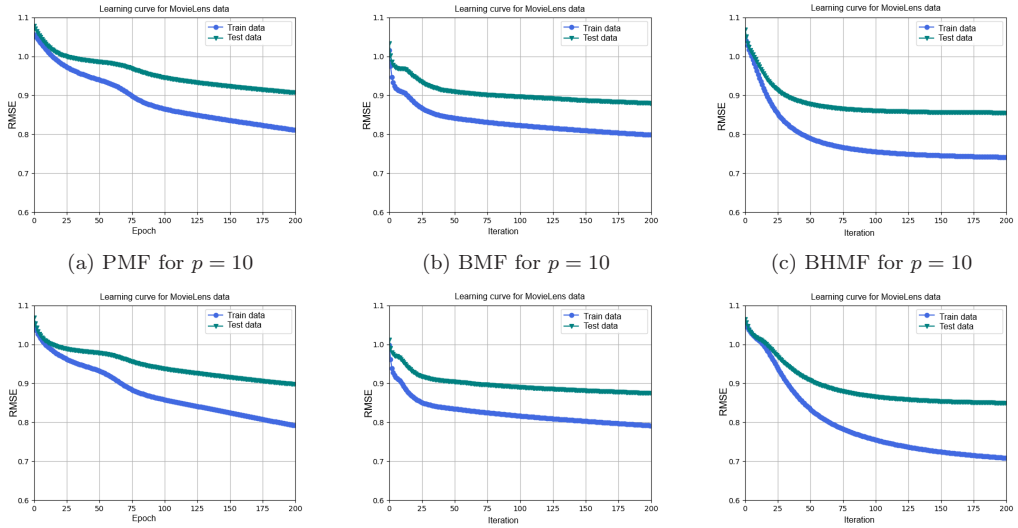
Table 2. The RMSE obtained from PMF, BMF and our BHMF.

	RMSE					
p	Train data			Test data		
	PMF	BMF	BHMF	PMF	BMF	BHMF
5	0.8221	0.8045	0.7757	0.9137	0.8839	0.8680
10	0.8112	0.7986	0.7413	0.9068	0.8796	0.8550
20	0.7999	0.7945	0.7201	0.9016	0.8785	0.8508
30	0.7922	0.7916	0.7087	0.8976	0.8743	0.8494

Figure 2 shows the learning curve of the PMF, BMF and BHMF models on MovieLens data with different p values. For PMF, it illustrates that increasing p generally reduces RMSE for both training and test data, with a consistent decline after the 25th epoch and a high initial drop in RMSE, unlike Netflix data. RMSE continues to decrease even after 200 epochs, suggesting potential for further improvement with more iterations. Similar to PMF, BMF reduces RMSE across all p values, with a rapid decline in the first 25 iterations, followed by a slower decrease. RMSE continues to drop after 200 iterations, indicating that additional iterations might improve

[2]https://grouplens.org

results. The BHMF model also shows a reduction in RMSE for all p values, with a significant drop in the first 25 iterations. However, as p increases, the rate of RMSE reduction slows. Even after 200 iterations, RMSE continues to decrease, albeit more gradually, suggesting that extending training might be unnecessary beyond this point. This pattern is consistent across both MovieLens and Netflix data.



(a) PMF for $p = 10$ (b) BMF for $p = 10$ (c) BHMF for $p = 10$

(d) PMF for $p = 30$ (e) BMF for $p = 30$ (f) BHMF for $p = 30$

Figure 2. Learning curves of PMF, BMF, and BHMF for MovieLens data at different values of p.

4. Concluding remarks

This paper developed a method for estimating user ratings of movies with millions of observations. It used latent factor techniques, collaborative filtering, and a Bayesian approach in recommendation systems. The method was applied to real data from Netflix and MovieLens to test its effectiveness. Model performance was evaluated based on RMSE for both training and test data. The data were divided into 75% for training and 25% for testing for Netflix, and MovieLens data were split into two samples with exactly 10 test ratings per user.

The Bayesian heteroscedastic matrix factorization (BHMF) model extended the Bayesian matrix factorization (BMF) to account for individual user (or item) variation. Probabilistic matrix factorization (PMF) and BMF generally performed worse in this study, with BHMF sometimes yielding similar or slightly worse results compared to BMF, especially for $p = 20$ and $p = 30$ on Netflix data. For test data, PMF did not achieve RMSE values below 0.87 for Netflix or 0.89 for MovieLens in any configuration, unlike BMF and BHMF. A significant limitation was the high computational cost, as increasing p increases the computation time due to the larger data volume.

The proposed methodology demonstrated high accuracy in estimating user ratings for both Netflix and MovieLens, despite the computational cost. Future work could explore improvements to BHMF, use numerical methods for parameter sampling, investigate alternative distributions, apply different priors, and test other datasets and iteration counts.

References

[1] Aggarwal, C., 2016. Recommender Systems: The Textbook, Springer. doi: 10.1007/978-3-319-29659-3

[2] Canny, J., 2002. Collaborative filtering with privacy via factor analysis. SIGIR '02 Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, 238–245. doi: 10.1145/564376.564419

[3] Devooght, R., Kourtellis, N., Mantrach, A. (2015). Dynamic matrix factorization with priors on unknown values. KDD '15 Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 189–198. doi: 10.1145/2783258.2783346

[4] Gemulla, R., Nijkamp, E., Haas, P. J., Sismanis, Y. 2011. Large-scale matrix factorization with distributed stochastic gradient descent. KDD '11 Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, 69–77. doi: 10.1145/2020408.2020426

[5] Huffel, S., Vandewalle, J., Haegemans, A., 1987. An efficient and reliable algorithm for computing the singular subspace of a matrix, associated with its smallest singular values. Journal of Computational and Applied Mathematics, 19 (3), 313–330. doi: 10.1016/0377-0427(87)90201-9

[6] Kabbur, S., Ning, X., Karypis, G. 2013. FISM: Factored item similarity models for top-N recommender systems. KDD '13 Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, 659–667. doi: 10.1145/2487575.2487589

[7] Salakhutdinov, R. and Mnih, A., 2008. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. ICML '08 Proceedings of the 25th International Conference on Machine Learning, 25, 880–887. doi: 10.1145/1390156.1390267

[8] Salakhutdinov, R. and Mnih, A., 2008. Probabilistic matrix factorization. Advances in Neural Information Processing Systems 20, Canada, 2008. http://papers.nips.cc/paper/3208-probabilistic-matrix-factorization.pdf

[9] Shen, H., Huang, J. Z., 2008. Sparse principal component analysis via regularized low rank matrix approximation. Journal of Multivariate Analysis, 99 (6), 1015–1034. doi: 10.1016/j.jmva.2007.06.007

[10] Shen, J., Cheng, Z., Yang, M., Han, B., Li, S., 2019. Style-oriented personalized landmark recommendation. IEEE Transactions on Industrial Electronics, 66 (12), 9768–9776. doi: 10.1109/TIE.2019.2910043

[11] Xian, Z., Li, Q., Li, G., Li, L. 2017. New collaborative filtering algorithms based on SVD++ and differential privacy. Mathematical Problems in Engineering, 2017 (1), 1–14. doi: 10.1155/2017/1975719.

Appendix A. The BHMF model and its full conditional distributions

The BHMF model, with $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$, is given by:

$$
\begin{aligned}
(r_{ij}|\boldsymbol{\upsilon}_i, \boldsymbol{\nu}_j, \alpha) &\sim \mathcal{N}(\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j, \alpha^{-1}), \\
(\boldsymbol{\upsilon}_i|\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}, \lambda_{\boldsymbol{\upsilon}_i}) &\sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, [\lambda_{\boldsymbol{\upsilon}_i}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}]^{-1}), \quad (\lambda_{\boldsymbol{\upsilon}_i}|\kappa_{\boldsymbol{\upsilon}}) \sim \mathcal{G}\left(\frac{\kappa_{\boldsymbol{\upsilon}}}{2}, \frac{\kappa_{\boldsymbol{\upsilon}}}{2}\right), \quad \kappa_{\boldsymbol{\upsilon}} \sim \mathcal{G}(a_{\kappa_{\boldsymbol{\upsilon}}}, b_{\kappa_{\boldsymbol{\upsilon}}}), \\
(\boldsymbol{\nu}_j|\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}, \lambda_{\boldsymbol{\nu}_j}) &\sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\nu}}, [\lambda_{\boldsymbol{\nu}_j}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}), \quad (\lambda_{\boldsymbol{\nu}_j}|\kappa_{\boldsymbol{\nu}}) \sim \mathcal{G}\left(\frac{\kappa_{\boldsymbol{\nu}}}{2}, \frac{\kappa_{\boldsymbol{\nu}}}{2}\right), \quad \kappa_{\boldsymbol{\nu}} \sim \mathcal{G}(a_{\kappa_{\boldsymbol{\nu}}}, b_{\kappa_{\boldsymbol{\nu}}}), \\
(\boldsymbol{\mu}_{\boldsymbol{\upsilon}}|\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}) &\sim \mathcal{N}(\boldsymbol{a}_{\boldsymbol{\upsilon}}, [b_{\boldsymbol{\upsilon}}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}]^{-1}), \quad \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} \sim \mathcal{W}(\boldsymbol{C}_{\boldsymbol{\upsilon}}, d_{\boldsymbol{\upsilon}}), \\
(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{\Lambda}_{\boldsymbol{\nu}}) &\sim \mathcal{N}(\boldsymbol{a}_{\boldsymbol{\nu}}, [b_{\boldsymbol{\nu}}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}), \quad \boldsymbol{\Lambda}_{\boldsymbol{\nu}} \sim \mathcal{W}(\boldsymbol{C}_{\boldsymbol{\nu}}, d_{\boldsymbol{\nu}}), \quad \text{and} \\
\alpha &\sim \mathcal{G}(a_\alpha, b_\alpha).
\end{aligned}
$$

It follows that

$$\begin{aligned}
p(\boldsymbol{\mathcal{R}}|\boldsymbol{U},\boldsymbol{V},\alpha) &= \prod_{i=1}^{m}\prod_{j=1}^{n}\left[\phi_1(r_{ij}|\boldsymbol{v}_i^T\boldsymbol{\nu}_j,\alpha^{-1})\right]^{\mathbb{I}_{ij}} \\
p(\boldsymbol{U}|\boldsymbol{\mu}_{\boldsymbol{v}},\boldsymbol{\Lambda}_{\boldsymbol{v}},\boldsymbol{\lambda}_{\boldsymbol{v}}) &= \prod_{i=1}^{m}\phi_p(\boldsymbol{v}_i|\boldsymbol{\mu}_{\boldsymbol{v}},[\lambda_{\boldsymbol{v}_i}\boldsymbol{\Lambda}_{\boldsymbol{v}}]^{-1}) \\
p(\boldsymbol{\lambda}_{\boldsymbol{v}}|\kappa_{\boldsymbol{v}}) &= \prod_{i=1}^{m}p(\lambda_{\boldsymbol{v}_i}|\kappa_{\boldsymbol{v}}) \\
p(\boldsymbol{V}|\boldsymbol{\mu}_{\boldsymbol{\nu}},\boldsymbol{\Lambda}_{\boldsymbol{\nu}},\boldsymbol{\lambda}_{\boldsymbol{\nu}}) &= \prod_{j=1}^{n}\phi_p(\boldsymbol{\nu}_j|\boldsymbol{\mu}_{\boldsymbol{\nu}},[\lambda_{\boldsymbol{\nu}}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}) \\
p(\boldsymbol{\lambda}_{\boldsymbol{\nu}}|\kappa_{\boldsymbol{\nu}}) &= \prod_{j=1}^{n}p(\lambda_{\boldsymbol{\nu}_j}|\kappa_{\boldsymbol{v}}).
\end{aligned}$$

Moreover,

$$\begin{aligned}
p(\boldsymbol{\mu}_{\boldsymbol{v}},\boldsymbol{\Lambda}_{\boldsymbol{v}}) &= p(\boldsymbol{\mu}_{\boldsymbol{v}}|\boldsymbol{\Lambda}_{\boldsymbol{v}})p(\boldsymbol{\Lambda}_{\boldsymbol{v}}) = \phi_p(\boldsymbol{\mu}_{\boldsymbol{v}}|\boldsymbol{a}_{\boldsymbol{v}},[b_{\boldsymbol{v}}\boldsymbol{\Lambda}_{\boldsymbol{v}}]^{-1})\omega(\boldsymbol{\Lambda}_{\boldsymbol{v}}|\boldsymbol{C}_{\boldsymbol{v}},d_{\boldsymbol{v}}) \\
p(\boldsymbol{\mu}_{\boldsymbol{\nu}},\boldsymbol{\Lambda}_{\boldsymbol{\nu}}) &= p(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{\Lambda}_{\boldsymbol{\nu}})p(\boldsymbol{\Lambda}_{\boldsymbol{\nu}}) = \phi_p(\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{a}_{\boldsymbol{\nu}},[b_{\boldsymbol{\nu}}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1})\omega(\boldsymbol{\Lambda}_{\boldsymbol{\nu}}|\boldsymbol{C}_{\boldsymbol{\nu}},d_{\boldsymbol{\nu}}).
\end{aligned}$$

From now on, consider $N = \sum_{i=1}^{m}\sum_{j=1}^{n}\mathbb{I}_{ij}$ to be the effective sample size (counting only the observed values).

A.1. *Full conditional distribution of* $\boldsymbol{\alpha}$

$$\begin{aligned}
p(\alpha|\mathcal{F}) &\propto p(\boldsymbol{\mathcal{R}}|\boldsymbol{U},\boldsymbol{V},\alpha)p(\alpha|a_\alpha,b_\alpha) \\
&\propto \alpha^{\frac{N}{2}+a_\alpha-1}\exp\left\{-\frac{\alpha}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbb{I}_{ij}(r_{ij}-\boldsymbol{v}_i^T\boldsymbol{\nu}_j)^2 - b_\alpha\alpha\right\} \\
&\propto \alpha^{(\frac{N}{2}+a_\alpha)-1}\exp\left\{-\alpha\left[\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbb{I}_{ij}(r_{ij}-\boldsymbol{v}_i^T\boldsymbol{\nu}_j)^2 + b_\alpha\right]\right\},
\end{aligned}$$

with $\mathcal{F} = (\boldsymbol{U},\boldsymbol{V},\boldsymbol{\mathcal{R}})$. Thus,

$$(\alpha|\mathcal{F}) \sim \mathcal{G}\left(\frac{N}{2}+a_\alpha, \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbb{I}_{ij}(r_{ij}-\boldsymbol{v}_i^T\boldsymbol{\nu}_j)^2 + b_\alpha\right).$$

A.2. *Full conditional distribution of* $\boldsymbol{v_i}$

$$\begin{aligned}
p(\boldsymbol{v}_i|\mathcal{F}) &\propto p(\boldsymbol{\mathcal{R}}|\boldsymbol{U},\boldsymbol{V},\alpha)p(\boldsymbol{v}_i|\boldsymbol{\mu}_{\boldsymbol{v}},\boldsymbol{\Lambda}_{\boldsymbol{v}},\lambda_{\boldsymbol{v}_i}) \\
&\propto \prod_{j=1}^{n}\left[\phi_1(r_{ij}|\boldsymbol{v}_i^T\boldsymbol{\nu}_j,\alpha^{-1})\right]^{\mathbb{I}_{ij}} p(\boldsymbol{v}_i|\boldsymbol{\mu}_{\boldsymbol{v}},\boldsymbol{\Lambda}_{\boldsymbol{v}},\lambda_{\boldsymbol{v}_i}) \\
&\propto \exp\left\{-\frac{\alpha}{2}\sum_{j=1}^{n}\mathbb{I}_{ij}(r_{ij}-\boldsymbol{v}_i^T\boldsymbol{\nu}_j)^2 - \frac{1}{2}(\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})^T[\lambda_{\boldsymbol{v}_i}\boldsymbol{\Lambda}_{\boldsymbol{v}}](\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})\right\},
\end{aligned}$$

with $\mathcal{F} = (\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\mathcal{R}}, \boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}, \lambda_{\boldsymbol{\upsilon}_i})$. Now, we have that

$$\begin{aligned}
SQ_{\boldsymbol{\upsilon}_i} &= \alpha \sum_{j=1}^{n} \mathbb{I}_{ij}(r_{ij} - \boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)^2 + (\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})^T [\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}](\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}}) \\
&= \alpha \sum_{j=1}^{n} \mathbb{I}_{ij} r_{ij}^2 - 2\alpha \sum_{j=1}^{n} \mathbb{I}_{ij} r_{ij} \boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij} (\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)^2 \\
&+ \boldsymbol{\upsilon}_i^T [\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}] \boldsymbol{\upsilon}_i - 2\boldsymbol{\upsilon}_i [\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}] \boldsymbol{\mu}_{\boldsymbol{\upsilon}} + \boldsymbol{\mu}_{\boldsymbol{\upsilon}}^T [\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}] \boldsymbol{\mu}_{\boldsymbol{\upsilon}} \\
&= c_1 + \boldsymbol{\upsilon}_i^T \left[\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij} \boldsymbol{\nu}_j \boldsymbol{\nu}_j^T \right] \boldsymbol{\upsilon}_i \\
&- 2\boldsymbol{\upsilon}_i^T \left[\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} \boldsymbol{\mu}_{\boldsymbol{\upsilon}} + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij} r_{ij} \boldsymbol{\nu}_j \right] \\
&= c_1 + \boldsymbol{\upsilon}_i^T \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} \boldsymbol{\upsilon}_i - 2\boldsymbol{\upsilon}_i^T \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} \boldsymbol{\mu}_{\boldsymbol{\upsilon}}^{\star} \\
&= c_2 + \boldsymbol{\upsilon}_i^T \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} \boldsymbol{\upsilon}_i - 2\boldsymbol{\upsilon}_i^T \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} \boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star} + \boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star T} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} \boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star} \\
&= c_2 + (\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star})^T \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} (\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star}),
\end{aligned}$$

where c_1 and c_2 are constants, whilst

$$\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} = \lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij} \boldsymbol{\nu}_j \boldsymbol{\nu}_j^T \quad \text{and} \quad \boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star} = \left[\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star} \right]^{-1} \left[\lambda_{\boldsymbol{\upsilon}_i} \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}} \boldsymbol{\mu}_{\boldsymbol{\upsilon}} + \alpha \sum_{j=1}^{n} \mathbb{I}_{ij} r_{ij} \boldsymbol{\nu}_j \right].$$

Moreover, note that

$$\begin{aligned}
\sum_{j=1}^{n} \mathbb{I}_{ij} (\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)^2 &= \sum_{j=1}^{n} \mathbb{I}_{ij} (\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)(\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j) = \sum_{j=1}^{n} \mathbb{I}_{ij} (\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)(\boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)^T \\
&= \sum_{j=1}^{n} \mathbb{I}_{ij} \boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j \boldsymbol{\nu}_j^T \boldsymbol{u}_i = \boldsymbol{\upsilon}_i^T \left[\sum_{j=1}^{n} \mathbb{I}_{ij} \boldsymbol{\nu}_j \boldsymbol{\nu}_j^T \right] \boldsymbol{\upsilon}_i.
\end{aligned}$$

It follows that $(\boldsymbol{\upsilon}_i | \mathcal{F}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\upsilon}_i}^{\star}, [\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}_i}^{\star}]^{-1})$.

A.3. *Full conditional distribution of $\boldsymbol{\nu}_j$*

$$\begin{aligned}
p(\boldsymbol{\nu}_j | \mathcal{F}) &\propto p(\boldsymbol{\mathcal{R}} | \boldsymbol{U}, \boldsymbol{V}, \alpha) p(\boldsymbol{\nu}_j | \boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}, \lambda_{\boldsymbol{\nu}_j}) \\
&\propto \prod_{i=1}^{m} \left[\phi_1(r_{ij} | \boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j, \alpha^{-1}) \right]^{\mathbb{I}_{ij}} p(\boldsymbol{\nu}_j | \boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}, \lambda_{\boldsymbol{\nu}_j}) \\
&\propto \exp \left\{ -\frac{\alpha}{2} \sum_{i=1}^{m} \mathbb{I}_{ij} (r_{ij} - \boldsymbol{\upsilon}_i^T \boldsymbol{\nu}_j)^2 - \frac{1}{2} (\boldsymbol{\nu}_j - \boldsymbol{\mu}_{\boldsymbol{\nu}})^T [\lambda_{\boldsymbol{\nu}_j} \boldsymbol{\Lambda}_{\boldsymbol{\nu}}] (\boldsymbol{\nu}_j - \boldsymbol{\mu}_{\boldsymbol{\nu}}) \right\},
\end{aligned}$$

with $\mathcal{F} = (\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\mathcal{R}}, \boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}}, \lambda_{\boldsymbol{\nu}_j})$. By developing the expression above in the same manner as the full conditional distribution of $\boldsymbol{\upsilon}_i$, we obtain

$$\boldsymbol{\Lambda}_{\boldsymbol{\nu}_j}^{\star} = \lambda_{\boldsymbol{\nu}_j} \boldsymbol{\Lambda}_{\boldsymbol{\nu}} + \alpha \sum_{i=1}^{m} \mathbb{I}_{ij} \boldsymbol{\upsilon}_i \boldsymbol{\upsilon}_i^T \quad \text{and} \quad \boldsymbol{\mu}_{\boldsymbol{\nu}_j}^{\star} = \left[\boldsymbol{\Lambda}_{\boldsymbol{\nu}_j}^{\star} \right]^{-1} \left[\lambda_{\boldsymbol{\nu}_j} \boldsymbol{\Lambda}_{\boldsymbol{\nu}} \boldsymbol{\mu}_{\boldsymbol{\nu}} + \alpha \sum_{i=1}^{m} \mathbb{I}_{ij} r_{ij} \boldsymbol{\upsilon}_i \right],$$

such that $(\boldsymbol{\nu}_j | \mathcal{F}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\nu}_j}^{\star}, [\boldsymbol{\Lambda}_{\boldsymbol{\nu}_j}^{\star}]^{-1})$.

A.4. *Full conditional distribution of* $(\boldsymbol{\mu}_{\boldsymbol{v}}, \boldsymbol{\Lambda}_{\boldsymbol{v}})$

$$
\begin{aligned}
p(\boldsymbol{\mu}_{\boldsymbol{v}}, \boldsymbol{\Lambda}_{\boldsymbol{v}}|\mathcal{F}) &\propto p(\boldsymbol{U}|\boldsymbol{\mu}_{\boldsymbol{v}}, \boldsymbol{\Lambda}_{\boldsymbol{v}}, \boldsymbol{\lambda}_{\boldsymbol{v}})p(\boldsymbol{\mu}_{\boldsymbol{v}}|\boldsymbol{\Lambda}_{\boldsymbol{v}})p(\boldsymbol{\Lambda}_{\boldsymbol{v}}) \\
&\propto \prod_{i=1}^{m}\left[|\boldsymbol{\Lambda}_{\boldsymbol{v}}|^{1/2}\exp\left\{-\frac{1}{2}(\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})^T[\lambda_{\boldsymbol{v}_i}\boldsymbol{\Lambda}_{\boldsymbol{v}}](\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})\right\}\right] \\
&\times |\boldsymbol{\Lambda}_{\boldsymbol{v}}|^{1/2}\exp\left\{-\frac{1}{2}(\boldsymbol{\mu}_{\boldsymbol{v}}-\boldsymbol{a}_{\boldsymbol{v}})^T(b_{\boldsymbol{v}}\boldsymbol{\Lambda}_{\boldsymbol{v}})(\boldsymbol{\mu}_{\boldsymbol{v}}-\boldsymbol{a}_{\boldsymbol{v}})\right\} \\
&\times |\boldsymbol{\Lambda}_{\boldsymbol{v}}|^{(d_{\boldsymbol{v}}-D_{\boldsymbol{v}}-1)/2}\exp\left\{-\frac{1}{2}\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{\Lambda}_{\boldsymbol{v}})\right\} \\
&= |\boldsymbol{\Lambda}_{\boldsymbol{v}}|^{m/2+1/2+(d_{\boldsymbol{v}}-D_{\boldsymbol{v}}-1)/2} \\
&\times \exp\left\{-\frac{1}{2}\sum\nolimits_{i=1}^{m}(\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})^T[\lambda_{\boldsymbol{v}_i}\boldsymbol{\Lambda}_{\boldsymbol{v}}](\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})\right\} \\
&\times \exp\left\{-\frac{1}{2}(\boldsymbol{\mu}_{\boldsymbol{v}}-\boldsymbol{a}_{\boldsymbol{v}})^T(b_{\boldsymbol{v}}\boldsymbol{\Lambda}_{\boldsymbol{v}})(\boldsymbol{\mu}_{\boldsymbol{v}}-\boldsymbol{a}_{\boldsymbol{v}})-\frac{1}{2}\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{\Lambda}_{\boldsymbol{v}})\right\},
\end{aligned}
$$

with $\mathcal{F} = (\boldsymbol{U}, \boldsymbol{\lambda}_{\boldsymbol{v}})$. Next, we split a few sums of squares and then combine them to determine the conditional distribution.

Part I

Let $\bar{\lambda}_{\boldsymbol{v}} = \frac{1}{m}\sum_{i=1}^{m}\lambda_{\boldsymbol{v}_i}$ and $\tilde{\boldsymbol{v}} = \frac{\sum_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i}{\sum_{i=1}^{m}\lambda_{\boldsymbol{v}_i}} = \frac{1}{m\bar{\lambda}_{\boldsymbol{v}}}\sum_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i$. Then,

$$
\begin{aligned}
\mathrm{SQ}_{\boldsymbol{\mu},1} &= \sum\nolimits_{i=1}^{m}(\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}})^T[\lambda_{\boldsymbol{v}_i}\boldsymbol{\Lambda}_{\boldsymbol{v}}](\boldsymbol{v}_i-\boldsymbol{\mu}_{\boldsymbol{v}}) \\
&= \mathrm{tr}\left(\sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}(\boldsymbol{v}_i\boldsymbol{v}_i^T-\boldsymbol{v}_i\boldsymbol{\mu}_{\boldsymbol{v}}^T-\boldsymbol{\mu}_{\boldsymbol{v}}\boldsymbol{v}_i^T+\boldsymbol{\mu}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}^T)\boldsymbol{\Lambda}_{\boldsymbol{v}}\right) \\
&= \mathrm{tr}\left(\left[m\bar{\lambda}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}^T-m\bar{\lambda}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T+\sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T\right]\boldsymbol{\Lambda}_{\boldsymbol{v}}\right) \\
&= \mathrm{tr}\left(\left[m\bar{\lambda}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}^T-m\bar{\lambda}_{\boldsymbol{v}}\boldsymbol{\mu}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T+m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T\right.\right. \\
&- \left.\left.m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T+\sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T\right]\boldsymbol{\Lambda}_{\boldsymbol{v}}\right) \\
&= \mathrm{tr}\left(\left[m\bar{\lambda}_{\boldsymbol{v}}(\boldsymbol{\mu}_{\boldsymbol{v}}-\tilde{\boldsymbol{v}})(\boldsymbol{\mu}_{\boldsymbol{v}}-\tilde{\boldsymbol{v}})^T+\sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T\right]\boldsymbol{\Lambda}_{\boldsymbol{v}}\right) \\
&= \mathrm{tr}\left(m\bar{\lambda}_{\boldsymbol{v}}(\boldsymbol{\mu}_{\boldsymbol{v}}-\tilde{\boldsymbol{v}})(\boldsymbol{\mu}_{\boldsymbol{v}}-\tilde{\boldsymbol{v}})^T\boldsymbol{\Lambda}_{\boldsymbol{v}}\right)+\mathrm{tr}\left(\left[\sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T\right]\boldsymbol{\Lambda}_{\boldsymbol{v}}\right) \\
&= (\boldsymbol{\mu}_{\boldsymbol{v}}-\tilde{\boldsymbol{v}})^T[m\bar{\lambda}_{\boldsymbol{v}}\boldsymbol{\Lambda}_{\boldsymbol{v}}](\boldsymbol{\mu}_{\boldsymbol{v}}-\tilde{\boldsymbol{v}})+\mathrm{tr}(\tilde{\mathrm{S}}_{\boldsymbol{v}}\boldsymbol{\Lambda}_{\boldsymbol{v}}),
\end{aligned}
$$

where

$$
\begin{aligned}
\tilde{\mathrm{S}}_{\boldsymbol{v}} &= \sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}(\boldsymbol{v}_i-\tilde{\boldsymbol{v}})(\boldsymbol{v}_i-\tilde{\boldsymbol{v}})^T \\
&= \sum\nolimits_{i=1}^{m}(\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T-\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\tilde{\boldsymbol{v}}^T-\lambda_{\boldsymbol{v}_i}\tilde{\boldsymbol{v}}\boldsymbol{v}_i^T+\lambda_{\boldsymbol{v}_i}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T) \\
&= \sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T+m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T \\
&= \sum\nolimits_{i=1}^{m}\lambda_{\boldsymbol{v}_i}\boldsymbol{v}_i\boldsymbol{v}_i^T-m\bar{\lambda}_{\boldsymbol{v}}\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^T.
\end{aligned}
$$

Part II

$$\begin{aligned}
\mathrm{SQ}_{\boldsymbol{\mu},2} &= (\boldsymbol{\mu}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})(\boldsymbol{\mu}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})+(\boldsymbol{\mu}_{\upsilon}-\boldsymbol{a}_{\upsilon})^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})(\boldsymbol{\mu}_{\upsilon}-\boldsymbol{a}_{\upsilon})\\
&= \boldsymbol{\mu}_{\upsilon}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{\mu}_{\upsilon}-2\boldsymbol{\mu}_{\upsilon}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}+\tilde{\boldsymbol{\upsilon}}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}\\
&+ \boldsymbol{\mu}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}-2\boldsymbol{\mu}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}+\boldsymbol{a}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}\\
&= \boldsymbol{\mu}_{\upsilon}^T((m\bar{\lambda}_{\upsilon}+b_{\upsilon})\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{\mu}_{\upsilon}-2\boldsymbol{\mu}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}(m\bar{\lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+b_{\upsilon}\boldsymbol{a}_{\upsilon})\\
&+ \tilde{\boldsymbol{\upsilon}}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}+\boldsymbol{a}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}\\
&= \boldsymbol{\mu}_{\upsilon}^T(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{\mu}_{\upsilon}-2\boldsymbol{\mu}_{\upsilon}^T(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}^{\star}+\boldsymbol{a}_{\upsilon}^{\star T}(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}^{\star}\\
&+ \tilde{\boldsymbol{\upsilon}}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}+\boldsymbol{a}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}-\boldsymbol{a}_{\upsilon}^{\star T}(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}^{\star}\\
&= (\boldsymbol{\mu}_{\upsilon}-\boldsymbol{a}_{\upsilon}^{\star})^T(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})(\boldsymbol{\mu}_{\upsilon}-\boldsymbol{a}_{\upsilon}^{\star})+\tilde{\boldsymbol{\upsilon}}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}\\
&+ \boldsymbol{a}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}-\boldsymbol{a}_{\upsilon}^{\star T}(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}^{\star},
\end{aligned}$$

where

$$b_{\upsilon}^{\star}=b_{\upsilon}+m\bar{\lambda}_{\upsilon}\quad\text{and}\quad\boldsymbol{a}_{\upsilon}^{\star}=\frac{1}{b_{\upsilon}^{\star}}(m\bar{\lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+b_{\upsilon}\boldsymbol{a}_{\upsilon}).$$

Thus, $(\boldsymbol{\mu}_{\upsilon}|\boldsymbol{\Lambda}_{\upsilon},\mathcal{F})\sim\mathcal{N}(\boldsymbol{a}_{\upsilon}^{\star},[b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon}]^{-1})$.

Part III

$$\begin{aligned}
\mathrm{SQ}_{\boldsymbol{\Lambda}} &= \tilde{\boldsymbol{\upsilon}}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}+\boldsymbol{a}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}-\boldsymbol{a}_{\upsilon}^{\star T}(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}^{\star}\\
&= \tilde{\boldsymbol{\upsilon}}^T(m\bar{\lambda}_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\tilde{\boldsymbol{\upsilon}}+\boldsymbol{a}_{\upsilon}^T(b_{\upsilon}\boldsymbol{\Lambda}_{\upsilon})\boldsymbol{a}_{\upsilon}-\frac{(m\bar{\lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+b_{\upsilon}\boldsymbol{a}_{\upsilon})^T}{b_{\upsilon}^{\star}}(b_{\upsilon}^{\star}\boldsymbol{\Lambda}_{\upsilon})\frac{(m\bar{\lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+b_{\upsilon}\boldsymbol{a}_{\upsilon})^T}{b_{\upsilon}^{\star}}\\
&= \frac{1}{b_{\upsilon}^{\star}}\left[\tilde{\boldsymbol{\upsilon}}^T\boldsymbol{\Lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}m\bar{\lambda}_{\upsilon}b_{\upsilon}^{\star}+\boldsymbol{a}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}\boldsymbol{a}_{\upsilon}b_{\upsilon}b_{\upsilon}^{\star}-(m\bar{\lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+b_{\upsilon}\boldsymbol{a}_{\upsilon})^T\boldsymbol{\Lambda}_{\upsilon}(m\bar{\lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+b_{\upsilon}\boldsymbol{a}_{\upsilon})\right]\\
&= \frac{1}{b_{\upsilon}^{\star}}\left[\tilde{\boldsymbol{\upsilon}}^T\boldsymbol{\Lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}(m^2\bar{\lambda}_{\upsilon}^2+m\bar{\lambda}_{\upsilon}b_{\upsilon})+\boldsymbol{a}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}\boldsymbol{a}_{\upsilon}(m\bar{\lambda}_{\upsilon}b_{\upsilon}+b_{\upsilon}^2)\right.\\
&- \left.m^2\bar{\lambda}_{\upsilon}^2\tilde{\boldsymbol{\upsilon}}^T\boldsymbol{\Lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}-2m\bar{\lambda}_{\upsilon}b_{\upsilon}\boldsymbol{a}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}-b_{\upsilon}^2\boldsymbol{a}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}\boldsymbol{a}_{\upsilon}\right]\\
&= \frac{m\bar{\lambda}_{\upsilon}b_{\upsilon}}{b_{\upsilon}^{\star}}\left[\boldsymbol{a}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}\boldsymbol{a}_{\upsilon}-2\boldsymbol{a}_{\upsilon}^T\boldsymbol{\Lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}+\tilde{\boldsymbol{\upsilon}}^T\boldsymbol{\Lambda}_{\upsilon}\tilde{\boldsymbol{\upsilon}}\right]\\
&= \frac{m\bar{\lambda}_{\upsilon}b_{\upsilon}}{b_{\upsilon}^{\star}}(\boldsymbol{a}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})^T\boldsymbol{\Lambda}_{\upsilon}(\boldsymbol{a}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})\\
&= \mathrm{tr}\left(\frac{m\bar{\lambda}_{\upsilon}b_{\upsilon}}{b_{\upsilon}^{\star}}(\boldsymbol{a}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})(\boldsymbol{a}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})^T\boldsymbol{\Lambda}_{\upsilon}\right)=\mathrm{tr}\left(\tilde{\mathrm{S}}_{\boldsymbol{a}}\boldsymbol{\Lambda}_{\upsilon}\right),
\end{aligned}$$

where

$$\tilde{\mathrm{S}}_{\boldsymbol{a}}=\frac{m\bar{\lambda}_{\upsilon}b_{\upsilon}}{b_{\upsilon}^{\star}}(\boldsymbol{a}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})(\boldsymbol{a}_{\upsilon}-\tilde{\boldsymbol{\upsilon}})^T.$$

Part IV

Now, we have that

$$\begin{aligned} p(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|\mathcal{F}) &\propto |\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|^{(d_{\boldsymbol{\upsilon}}+m-D_{\boldsymbol{\upsilon}}-1)/2} \exp\left\{-\frac{1}{2}\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{\upsilon}}^{-1}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}) - \frac{1}{2}\mathrm{tr}(\tilde{\mathrm{S}}_{\boldsymbol{\upsilon}}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}) - \frac{1}{2}\mathrm{tr}(\tilde{\mathrm{S}}_{\boldsymbol{a}}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}})\right\} \\ &= |\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|^{(d_{\boldsymbol{\upsilon}}^{\star}-D_{\boldsymbol{\upsilon}}-1)/2} \exp\left\{-\frac{1}{2}\mathrm{tr}([\boldsymbol{C}_{\boldsymbol{\upsilon}}^{\star}]^{-1}\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}})\right\}, \end{aligned}$$

where

$$d_{\boldsymbol{\upsilon}}^{\star} = d_{\boldsymbol{\upsilon}} + m \quad \text{and} \quad [\boldsymbol{C}_{\boldsymbol{\upsilon}}^{\star}]^{-1} = \boldsymbol{C}_{\boldsymbol{\upsilon}}^{-1} + \tilde{\mathrm{S}}_{\boldsymbol{\upsilon}} + \tilde{\mathrm{S}}_{\boldsymbol{a}}.$$

Thus, $(\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}|\mathcal{F}) \sim \mathcal{W}(\boldsymbol{C}_{\boldsymbol{\upsilon}}^{\star}, d_{\boldsymbol{\upsilon}}^{\star})$.

A.5. *Full conditional distribution of* $(\boldsymbol{\mu}_{\boldsymbol{\nu}}, \boldsymbol{\Lambda}_{\boldsymbol{\nu}})$

Based on the calculations from the previous full conditional distribution, we have that

$$\begin{aligned} (\boldsymbol{\mu}_{\boldsymbol{\nu}}|\boldsymbol{\Lambda}_{\boldsymbol{\nu}}, \boldsymbol{V}, \boldsymbol{\lambda}_{\boldsymbol{\nu}}) &\sim \mathcal{N}(\boldsymbol{a}_{\boldsymbol{\nu}}^{\star}, [b_{\boldsymbol{\nu}}^{\star}\boldsymbol{\Lambda}_{\boldsymbol{\nu}}]^{-1}); \\ (\boldsymbol{\Lambda}_{\boldsymbol{\nu}}|\boldsymbol{V}, \boldsymbol{\lambda}_{\boldsymbol{\nu}}) &\sim \mathcal{W}(\boldsymbol{C}_{\boldsymbol{\nu}}^{\star}, d_{\boldsymbol{\nu}}^{\star}), \end{aligned}$$

where

$$\begin{aligned} b_{\boldsymbol{\nu}}^{\star} &= b_{\boldsymbol{\nu}} + n\bar{\lambda}_{\boldsymbol{\nu}}, \quad d_{\boldsymbol{\nu}}^{\star} = d_{\boldsymbol{\nu}} + n, \quad \boldsymbol{a}_{\boldsymbol{\nu}}^{\star} = \frac{1}{b_{\boldsymbol{\nu}}^{\star}}(n\bar{\lambda}_{\boldsymbol{\nu}}\tilde{\boldsymbol{\nu}} + b_{\boldsymbol{\nu}}\boldsymbol{a}_{\boldsymbol{\nu}}), \\ \tilde{\mathrm{Q}}_{\boldsymbol{\nu}} &= \sum\nolimits_{j=1}^{n} \lambda_{\boldsymbol{\nu}_j}(\boldsymbol{\nu}_j - \tilde{\boldsymbol{\nu}})(\boldsymbol{\nu}_j - \tilde{\boldsymbol{\nu}})^T, \\ \tilde{\mathrm{Q}}_{\boldsymbol{a}} &= \frac{n\bar{\lambda}_{\boldsymbol{\nu}} b_{\boldsymbol{\nu}}}{b_{\boldsymbol{\nu}}^{\star}}(\boldsymbol{a}_{\boldsymbol{\nu}} - \tilde{\boldsymbol{\nu}})(\boldsymbol{a}_{\boldsymbol{\nu}} - \tilde{\boldsymbol{\nu}})^T, \\ [\boldsymbol{C}_{\boldsymbol{\nu}}^{\star}]^{-1} &= \boldsymbol{C}_{\boldsymbol{\nu}}^{-1} + \tilde{\mathrm{Q}}_{\boldsymbol{\nu}} + \tilde{\mathrm{Q}}_{\boldsymbol{a}}. \end{aligned}$$

A.6. *Full conditional distribution of* $\boldsymbol{\lambda}_{\boldsymbol{\upsilon}_i}$

$$\begin{aligned} p(\lambda_{\boldsymbol{\upsilon}_i}|\mathcal{F}) &\propto p(\boldsymbol{\upsilon}_i|\boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}, \lambda_{\boldsymbol{\upsilon}_i}) p(\lambda_{\boldsymbol{\upsilon}_i}|\kappa_{\boldsymbol{\upsilon}}) \\ &\propto \lambda_{\boldsymbol{\upsilon}_i}^{(D_{\boldsymbol{\upsilon}}+\kappa_{\boldsymbol{\upsilon}})/2-1} \exp\left\{-\frac{\lambda_{\boldsymbol{\upsilon}_i}}{2}\left[\kappa_{\boldsymbol{\upsilon}} + (\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})^T\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}(\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})\right]\right\}, \end{aligned}$$

with $\mathcal{F} = (\boldsymbol{U}, \boldsymbol{\mu}_{\boldsymbol{\upsilon}}, \boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}, \kappa_{\boldsymbol{\upsilon}})$. Thus,

$$(\lambda_{\boldsymbol{\upsilon}_i}|\mathcal{F}) \sim \mathcal{G}\left(\frac{D_{\boldsymbol{\upsilon}} + \kappa_{\boldsymbol{\upsilon}}}{2}, \frac{1}{2}\left[\kappa_{\boldsymbol{\upsilon}} + (\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})^T\boldsymbol{\Lambda}_{\boldsymbol{\upsilon}}(\boldsymbol{\upsilon}_i - \boldsymbol{\mu}_{\boldsymbol{\upsilon}})\right]\right).$$

A.7. *Full conditional distribution of* $\boldsymbol{\lambda_{\nu_j}}$

$$\begin{aligned} p(\lambda_{\boldsymbol{\nu}_j}|\mathcal{F}) &\propto p(\boldsymbol{\nu}_j|\boldsymbol{\mu_\nu}, \boldsymbol{\Lambda_\nu}, \lambda_{\boldsymbol{\nu}_j}) p(\lambda_{\boldsymbol{\nu}_j}|\kappa_{\boldsymbol{\nu}}) \\ &\propto \lambda_{\boldsymbol{\nu}_j}^{(D_{\boldsymbol{\nu}}+\kappa_{\boldsymbol{\nu}})/2-1} \exp\left\{-\frac{\lambda_{\boldsymbol{\nu}_j}}{2}\left[\kappa_{\boldsymbol{\nu}} + (\boldsymbol{\nu}_j - \boldsymbol{\mu_\nu})^T \boldsymbol{\Lambda_\nu}(\boldsymbol{\nu}_j - \boldsymbol{\mu_\nu})\right]\right\}. \end{aligned}$$

with $\mathcal{F} = (\boldsymbol{V}, \boldsymbol{\mu_\nu}, \boldsymbol{\Lambda_\nu}, \kappa_{\boldsymbol{\nu}})$. Thus,

$$(\lambda_{\boldsymbol{\nu}_j}|\mathcal{F}) \sim \mathcal{G}\left(\frac{D_{\boldsymbol{\nu}} + \kappa_{\boldsymbol{\nu}}}{2}, \frac{1}{2}\left[\kappa_{\boldsymbol{\nu}} + (\boldsymbol{\nu}_j - \boldsymbol{\mu_\nu})^T \boldsymbol{\Lambda_\nu}(\boldsymbol{\nu}_j - \boldsymbol{\mu_\nu})\right]\right).$$

A.8. *Full conditional distribution of* $\boldsymbol{\kappa_\upsilon}$

$$\begin{aligned} p(\kappa_{\boldsymbol{\upsilon}}|\boldsymbol{\lambda_\upsilon}) &\propto p(\boldsymbol{\lambda_\upsilon}|\kappa_{\boldsymbol{\upsilon}}) p(\kappa_{\boldsymbol{\upsilon}}) = \left[\prod_{i=1}^{m} p(\boldsymbol{\lambda}_{\boldsymbol{\upsilon}_i}|\kappa_{\boldsymbol{\upsilon}})\right] p(\kappa_{\boldsymbol{\upsilon}}) \\ &\propto \frac{[\kappa_{\boldsymbol{\upsilon}}/2]^{m\kappa_{\boldsymbol{\upsilon}}/2}}{[\Gamma(\kappa_{\boldsymbol{\upsilon}}/2)]^m}\left[\prod_{i=1}^{m} \lambda_{\boldsymbol{\upsilon}_i}^{\kappa_{\boldsymbol{\upsilon}}/2}\right] \exp\left\{-\frac{\kappa_{\boldsymbol{\upsilon}}}{2} m\bar{\lambda}_{\boldsymbol{\upsilon}}\right\} p(\kappa_{\boldsymbol{\upsilon}}) \\ &= \exp\left\{-\frac{\kappa_{\boldsymbol{\upsilon}}}{2}\left[\sum_{i=1}^{m}(\lambda_{\boldsymbol{\upsilon}_i} - \ln(\lambda_{\boldsymbol{\upsilon}_i})) - m\ln\left(\frac{\kappa_{\boldsymbol{\upsilon}}}{2}\right)\right] - m\ln\Gamma\left(\frac{\kappa_{\boldsymbol{\upsilon}}}{2}\right)\right\} p(\kappa_{\boldsymbol{\upsilon}}) \end{aligned}$$

A.9. *Full conditional distribution of* $\boldsymbol{\kappa_\nu}$

$$\begin{aligned} p(\kappa_{\boldsymbol{\nu}}|\boldsymbol{\lambda_\nu}) &\propto p(\boldsymbol{\lambda_\nu}|\kappa_{\boldsymbol{\nu}}) p(\kappa_{\boldsymbol{\nu}}) = \left[\prod_{j=1}^{n} p(\boldsymbol{\lambda}_{\boldsymbol{\nu}_j}|\kappa_{\boldsymbol{\nu}})\right] p(\kappa_{\boldsymbol{\nu}}) \\ &\propto \frac{[\kappa_{\boldsymbol{\nu}}/2]^{n\kappa_{\boldsymbol{\nu}}/2}}{[\Gamma(\kappa_{\boldsymbol{\nu}}/2)]^n}\left[\prod_{j=1}^{n} \lambda_{\boldsymbol{\nu}_j}^{\kappa_{\boldsymbol{\nu}}/2}\right] \exp\left\{-\frac{\kappa_{\boldsymbol{\nu}}}{2} n\bar{\lambda}_{\boldsymbol{\nu}}\right\} p(\kappa_{\boldsymbol{\nu}}) \\ &= \exp\left\{-\frac{\kappa_{\boldsymbol{\nu}}}{2}\left[\sum_{j=1}^{n}(\lambda_{\boldsymbol{\nu}_j} - \ln(\lambda_{\boldsymbol{\nu}_j})) - n\ln\left(\frac{\kappa_{\boldsymbol{\nu}}}{2}\right)\right] - n\ln\Gamma\left(\frac{\kappa_{\boldsymbol{\nu}}}{2}\right)\right\} p(\kappa_{\boldsymbol{\nu}}). \end{aligned}$$