

SOFTWARE

Open Access

mbend: an R package for bending non-positive-definite symmetric matrices to positive-definite



Mohammad Ali Nilforooshan 

Abstract

Background: R package mbend was developed for bending symmetric non-positive-definite matrices to positive-definite (PD). Bending is a procedure of transforming non-PD matrices to PD. The covariance matrices used in multi-trait best linear unbiased prediction (BLUP) should be PD. Two bending methods are implemented in mbend. The first is an unweighted bending with small positive values in a descending order replacing negative eigenvalues (LRS14), and the second method is a weighted (precision-based) bending with a custom small positive value (ϵ) replacing smaller eigenvalues (HJ03). Weighted bending is beneficial, as it relaxes low precision elements to change and it reduces or prohibits the change in high precision elements. Therefore, a weighted version of LRS14 was developed in mbend. In cases where the precision of matrix elements is unknown, the package provides an unweighted version of HJ03. Another unweighted bending method (DB88) was tested, by which all eigenvalues are changed (eigenvalues less than ϵ replaced with $100 \times \epsilon$), and it is originally designed for correlation matrices.

Results: Different bending procedures were conducted on a 5×5 covariance matrix (\mathbf{V}), \mathbf{V} converted to a correlation matrix (\mathbf{C}) and an ill-conditioned 1000×1000 genomic relationship matrix (\mathbf{G}). Considering weighted distance statistics between matrix elements before and after bending, weighting considerably improved the bending quality. For weighted and unweighted bending of \mathbf{V} and \mathbf{C} , HJ03-4 (HJ03, $\epsilon = 10^{-4}$) performed the best. HJ03-2 (HJ03, $\epsilon = 10^{-2}$) ranked better than LRS14 for \mathbf{V} , but not for \mathbf{C} . Though the differences were marginal, LRS14 performed the best for \mathbf{G} . DB88-4 (DB88, $\epsilon = 10^{-4}$) was used for unweighted bending and it ranked the last. This method could perform considerably better with a lower ϵ .

Conclusions: R package mbend provides necessary tools for transforming symmetric non-PD matrices to PD, using different methods and parameters. There were benefits in both weighted bending and small positive values in a descending order replacing negative eigenvalues. Thus, weighted LRS14 was implemented in mbend. Different bending methods might be preferable for different matrices, depending on the matrix type (covariance vs. correlation), number and the magnitude of negative eigenvalues, and the matrix size.

Keywords: Matrix, Positive-definite, Bending, Eigenvalue, R

Correspondence: mohammad.nilforooshan@lic.co.nz
Livestock Improvement Corporation, Private Bag 3016, Hamilton 3240, New Zealand



© The Author(s). 2020 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

Background

Even in their simplest form, multivariate animal models rely on genetic and residual variance-covariance matrices across traits [1]. These matrices are in the order of the number of traits in the model, and their inverses are incorporated in the mixed model equations. Inversion of a matrix is often done using Cholesky decomposition, which requires the matrix to be positive-definite (PD). For models including additional random effects (e.g., animal permanent environment, maternal genetic, and maternal permanent environment), additional covariance matrices and their inverses are also required. To date, restricted (or residual) maximum likelihood (REML) [2] is the preferred method for estimating the variance components associated with the model. REML estimates are always PD, but the starting matrices need to be PD. Elements of these matrices are usually from different sources of information and the resulting matrices are, therefore, likely to be non-PD [3].

Another complexity is the estimation of covariance matrices for several traits at a time. This complexity increases by both the number of traits, and the number of random effects. In many situations, there might not be enough data points to support accurate inferences about all the variance components, simultaneously. Therefore, when many traits are included, variance components are usually estimated for subsets of traits [4]. The assembly of these small matrices to a large matrix, together with best guesses for missing elements (from literature or phenotypic covariances and heritabilities) can be non-PD.

The procedure of “bending”, which involves modifying eigenvalues of a non-PD matrix, was first introduced by Hayes and Hill [5]. Latter studies in the field of animal breeding and genetics presented different bending procedures. Jorjani et al. [4] introduced weighted bending, where a weight matrix is provided for the matrix to be bent. Meyer and Kirkpatrick [6] developed a bending procedure, based on penalized maximum likelihood and reducing bias in the estimates of canonical heritabilities (the eigenvalues of $\mathbf{P}^{-1}\mathbf{G}$, where \mathbf{P} and \mathbf{G} are the phenotypic and genetic covariance matrices, respectively). Also, Schaeffer [3] introduced a bending procedure, which involves changing negative eigenvalues to small positive values and reconstructing the matrix. A bending method for correlation matrices (also called smoothing), used in the field of psychology, involves changing all eigenvalues [7]. All these bending methods aim to produce a PD matrix, least deviated from the original non-PD matrix.

The aim of this study was to introduce R package mbend [8], which is a free and open source tool for bending symmetric non-PD matrices to PD, based on

eigenvalue modification of the non-PD matrix. Comparison of different methods were provided, using example covariance and correlation matrices, as well as an artificial ill-conditioned genomic relationship matrix (\mathbf{G}). R package mbend covers methods of Jorjani et al. [4] and Schaeffer [3], as well as extensions to these methods.

Implementation

R package mbend [8] is written in R, and it is available on CRAN repository (<https://cran.r-project.org>) and can be installed, using the command `install.packages("mbend")`. In this study, the functionality of this package was presented using the same 5×5 non-PD matrix used by Jorjani et al. [4] and Schaeffer [3].

$$\mathbf{V} = \begin{bmatrix} 100 & 95 & 80 & 40 & 40 \\ & 100 & 95 & 80 & 40 \\ & & 100 & 95 & 80 \\ & & & 100 & 95 \\ Sym. & & & & 100 \end{bmatrix}$$

To study bending on a correlation matrix, $\mathbf{C} = \mathbf{V}/100$ was used. Jorjani et al. [4] used a matrix of weighting factors (\mathbf{W}) as the reciprocal of the number of animals involved in the estimation of variance components. The same matrix is also used in this study for weighted bending.

$$\mathbf{W} = 1/ \begin{bmatrix} 1000 & 500 & 20 & 50 & 200 \\ & 1000 & 500 & 5 & 50 \\ & & 1000 & 20 & 20 \\ & & & 1000 & 200 \\ Sym. & & & & 1000 \end{bmatrix}$$

For further comparisons, an ill-conditioned \mathbf{G} matrix was constructed using random genotypes on 5000 SNP and 1000 animals, without any quality control checks, and 10 duplicated genotypes, which resulted in a \mathbf{G} with 5 negative eigenvalues (ranging between $-224e-17$ to $-3.5e-17$) and 357 eigenvalues between 0 and 1. Matrix \mathbf{G} was constructed using method 1 of VanRaden [9]. The code for constructing the \mathbf{G} matrix together with all the (R) code used in this study are provided in the data repository.

R package mbend [8] was used throughout this study. It provides different methods for weighted and unweighted bending of symmetric non-PD matrices to PD. Two bending methods of Jorjani et al. [4] and Schaeffer [3] and extensions to them are implemented in R package mbend [8].

Method of Jorjani et al. [4]

This method (HJ03) is an iterative weighting procedure that converts a non-PD covariance matrix to a PD

matrix at convergence. It considers different precision associated with different elements of the covariance matrix. Therefore, minimising the change in matrix elements with high accuracy, through a weight matrix. Given the non-PD covariance matrix V , the weight matrix W , and a small positive real number (e.g., $\epsilon = 10^{-4}$), the procedure is as follows:

1. Decompose V_n to $U_n D_n U_n'$, where U_n and D_n are the matrix of eigenvectors and diagonal matrix of eigenvalues, and n is the iteration number.
2. Replace eigenvalues less than ϵ with ϵ in D_n to get Δ_n .
3. $V_{n+1} = V_n - [V_n - U_n \Delta_n U_n'] \circ W$, where \circ is the Hadamard function.
4. Repeat until V_{n+1} is PD.

The smaller the w_{ij} (element in W), the higher the relative certainty about v_{ij} (element in V). Accordingly, to retain an element of the matrix fixed during bending, that element would receive a weight of zero. In comparison with the weighted procedure, in the unweighted procedure, $V_{n+1} = U_n \Delta_n U_n'$. If no weight matrix is provided, the program performs an unweighted bending. Also, if the matrix is already PD, the program returns a message that “No action was required. The matrix is positive-definite”.

Jorjani et al. [4] extended their weighted bending method for covariance matrices to correlation matrices. R package mbend took a different approach for correlation matrices. First, it automatically recognises correlation matrices by checking all diagonal values against 1. Second, it treats correlation matrices as covariance matrices that should keep their diagonal elements constant during bending, by setting $w_{ii} = 0$. To avoid this restriction for a non-correlation matrix with all diagonal elements of 1 (e.g., a phenotypic matrix with variables standardised for phenotypic variances of 1), simply the matrix is multiplied by a positive constant, and then the resulting bent matrix is divided by that constant. A different approach that could be used for correlation matrices was treating them as covariance matrices and transforming the bent matrix (\hat{V}) to $T\hat{V}T'$, where $T^2 = \text{diag}(1/\text{diag}(\hat{V}))$.

Method of Schaeffer [3]

This method (LRS14) is an unweighted bending procedure, which means that different matrix elements are of the same precision. Negative eigenvalues are replaced with small positive values that are in a descending order (unlike equal values for HJ03). In this method, each of the m negative eigenvalues (λ_i) is replaced with $\rho(s - \lambda_i)^2 / (100s^2 + 1)$, where ρ is the smallest positive

eigenvalue and $s = 2 \sum_{i=1}^m \lambda_i$. In R package mbend, a weighted bending derivate of LRS14 is implemented by combining this method with HJ03 [8].

Method of Bock et al. [7]

This method (DB88) is used for bending correlation matrices [7]. In this method, eigenvalues smaller than ϵ are replaced with $100 \times \epsilon$. Also, unlike Jorjani et al. [4] and Schaeffer [3], eigenvalues greater than ϵ are changed to sum the number of those eigenvalues. The logic behind it might be that the sum of eigenvalues in a PD correlation matrix is equal to the size (trace) of the matrix. This method is implemented in function `cor.smooth` of R package psych (`psych::cor.smooth`) [10]. As this method is designed for bending correlation matrices, covariance matrices are first transformed to correlation matrices, after bending, the resulting matrix is transformed back to a covariance matrix (i.e., $T\hat{C}T'$, where $T^2 = \text{diag}(\text{diag}(V))$). That means, only the off-diagonal elements of the matrix change by bending.

Deviation and correlation

R package mbend returns the following statistics:

1. Minimum deviation ($\hat{V}-V$), together with matrix indices of the element
2. Maximum deviation, together with matrix indices of the element
3. Average deviation of the upper triangle elements
4. Average absolute deviation of the upper triangle elements (AAD)
5. Weighted AAD
6. Correlation between the upper triangle elements
7. Weighted correlation between the upper triangle elements
8. Root of mean squared deviation of the upper triangle elements (RMSD)
9. Weighted RMSD

$$AAD = \frac{\sum_{i=1}^n \sum_{j=1, j \geq i}^n |\hat{v}_{ij} - v_{ij}|}{n(n+1)/2}$$

$$Weighted\ AAD_{(w_{ij} > 0)} = \frac{\sum_{i=1}^n \sum_{j=1, j \geq i}^n |(\hat{v}_{ij} - v_{ij}) / w_{ij}|}{\sum_{i=1}^n \sum_{j=1, j \geq i}^n (1/w_{ij})}$$

$$RMSD = \sqrt{\frac{\sum_{i=1}^n \sum_{j=1, j \geq i}^n (\hat{v}_{ij} - v_{ij})^2}{n(n+1)/2}}$$

$$Weighted\ RMSD_{(w_{ij} > 0)} = \sqrt{\frac{\sum_{i=1}^n \sum_{j=1, j \geq i}^n ((\hat{v}_{ij} - v_{ij}) / w_{ij})^2}{\sum_{i=1}^n \sum_{j=1, j \geq i}^n (1/w_{ij}^2)}}$$

where, n is the size of the matrix, and w_{ij} , v_{ij} and \hat{v}_{ij} are the elements of \mathbf{W} , \mathbf{V} and $\hat{\mathbf{V}}$, respectively. The weighted statistics (weighted correlation, weighted AAD and weighted RMSD) are calculated for matrix elements with $w_{ij} > 0$. For correlation matrices, upper triangle elements did not include diagonal elements. Thus,

$$\begin{aligned} \text{AAD} &= \frac{\sum_{i=1}^{n-1} \sum_{j=2, j \geq i}^n |\hat{v}_{ij} - v_{ij}|}{n(n-1)/2}, \\ \text{Weighted AAD}_{(w_{ij} > 0)} &= \frac{\sum_{i=1}^{n-1} \sum_{j=2, j \geq i}^n |(\hat{v}_{ij} - v_{ij})/w_{ij}|}{\sum_{i=1}^{n-1} \sum_{j=2, j \geq i}^n (1/w_{ij})}, \\ \text{RMSD} &= \sqrt{\frac{\sum_{i=1}^{n-1} \sum_{j=2, j \geq i}^n (\hat{v}_{ij} - v_{ij})^2}{n(n-1)/2}}, \\ \text{Weighted RMSD}_{(w_{ij} > 0)} &= \sqrt{\frac{\sum_{i=1}^{n-1} \sum_{j=2, j \geq i}^n ((\hat{v}_{ij} - v_{ij})/w_{ij})^2}{\sum_{i=1}^{n-1} \sum_{j=2, j \geq i}^n (1/w_{ij}^2)}}. \end{aligned}$$

Function “bend”

R package mbend has a function called bend that is used with the syntax: bend (inmat, wtmat, reciprocal=FALSE, max.iter=10,000, small.positive=0.0001, method=“hj”). If any of the last 4 arguments are missing, the function will use the default value (FALSE, 10000, 0.0001, and “hj”, respectively). Arguments inmat and wtmat are for the matrix to be bent and the weight matrix. If wtmat is missing, an unweighted bending is performed. Argument reciprocal takes TRUE or FALSE as input, and if TRUE, reciprocals of \mathbf{W} elements are used. Where $w_{ij}=0$, it would remain 0. This argument is ignored if no weight matrix is provided to the function. The maximum number of iterations is defined by max.iter. The argument small.positive is used for HJ03 and ignored for LRS14. It is a user-defined small positive value (ϵ) replacing smaller eigenvalues in \mathbf{D} [4]. Argument method takes “hj” or “lrs” for HJ03 and LRS14, respectively.

The function returns $\hat{\mathbf{V}}$, eigenvalues of \mathbf{V} and $\hat{\mathbf{V}}$, and the statistics described in “Deviation and correlation”, all in a single list. Where weighted bending is applied, the number of upper triangle elements with $w_{ij} > 0$ (w_gt_0) is reported, as the weighted statistics rely on these observations. An example of weighted bending a correlation matrix, using bend function of mbend package is provided in the Additional file 1.

Results

The covariance matrix (V)

Matrix \mathbf{V} is a non-PD covariance matrix. The following command in R shows eigenvalues from the eigendecomposition of \mathbf{V} .

```
> round (eigen (V) $values, 2)
```

```
[1] 399.48 98.52 23.65 -3.12 -18.52.
```

Table 1 shows deviations and correlations between \mathbf{V} and $\hat{\mathbf{V}}$ for unweighted bending, using HJ03–2 (HJ03, $\epsilon = 10^{-2}$), HJ03–4 (HJ03, $\epsilon = 10^{-4}$), LRS14 and DB88–4 (DB88, $\epsilon = 10^{-4}$). The unweighted HJ03 is like the iterative spectral method [11], which is a simplified form of altering spectral method [12]. The difference between the unweighted HJ03 and the iterative spectral method is that, in the iterative spectral method, negative eigenvalues are replaced with the small positive value, but in HJ03, eigenvalues smaller than the small positive value are replaced with the small positive value. An unweighted HJ03 is equivalent to HJ03 with $\mathbf{W} = \mathbf{11}'$. Though some differences were small, the methods can be ranked from HJ03–4 being the best performer to HJ03–2, LRS14 and DB88–4. All the processes converged in 1 iteration.

Table 2 shows deviations and correlations between \mathbf{V} and $\hat{\mathbf{V}}$ for weighted bending, using HJ03–2, HJ03–4 and LRS14. The weighted bending procedures produced the same weighted correlation coefficients (0.9955) between the upper triangle elements of \mathbf{V} and $\hat{\mathbf{V}}$. As expected, HJ03–4 performed better than HJ03–2. LRS14 produced the closest mean of deviation to zero. However, it performed worse than HJ03–2 (considering the range of deviations, weighted AAD, weighted RMSD, and the number of iterations to convergence). LRS14 took the maximum (787) number of iterations to converge. However, it was not a concern, as the convergence was made in less than 0.2 s, due to the small size of \mathbf{V} .

The program provides weighted statistics for weighted bending. Those statistics were manually calculated for unweighted bending of \mathbf{V} and \mathbf{C} matrices. The code is available in the data repository, and the results are provided in Table S1. Comparing Tables 1, 2 and S1 shows lower weighted AAD and weighted

Table 1 Deviation and correlation between the upper triangle elements of $V_{(5 \times 5)}$ (the covariance matrix) and its unweighted bent matrix

Statistics	HJ03–2 ^a	HJ03–4 ^b	LRS14 ^c	DB88–4 ^d
Min (dev.)	−5.9320	−5.9296	−5.9370	−10.9546
Max (dev.)	6.5016	6.4973	6.5418	2.6427
Mean (dev.)	0.7241	0.7235	0.7330	−2.9890
AAD	3.3754	3.3727	3.4087	3.6937
Correlation	0.9856	0.9856	0.9855	0.9833
RMSD	3.9300	3.9275	3.9542	5.0784
Number of iterations	1	1	1	1

dev. = bend(\mathbf{V}) – \mathbf{V} ; AAD = average absolute deviation; RMSD = root of mean squared deviation; ^a Method of Jorjani et al. [4] with $\epsilon = 10^{-2}$; ^b Method of Jorjani et al. [4] with $\epsilon = 10^{-4}$; ^c Method of Schaeffer [3]; ^d Method of Bock et al. [7] with $\epsilon = 10^{-4}$

Table 2 Deviation and correlation between the upper triangle elements of $V_{(5 \times 5)}$ (the covariance matrix) and its weighted (using $W_{(5 \times 5)}$) bent matrix

Statistics	HJ03-2 ^a	HJ03-4 ^b	LRS14 ^c
Min (dev.)	-20.0068	-20.0008	-19.9923
Max (dev.)	5.8467	5.8456	5.8638
Mean (dev.)	-1.7167	-1.7161	-1.7158
AAD	3.6262	3.6253	3.6338
Weighted AAD	0.6102	0.6100	0.6122
Correlation	0.9623	0.9623	0.9622
Weighted correlation	0.9955	0.9955	0.9955
RMSD	6.3706	6.3687	6.3748
Weighted RMSD	0.5328	0.5327	0.5347
Number of iterations	248	428	787

dev. = bend(\mathbf{V}) - \mathbf{V} ; AAD = average absolute deviation; RMSD = root of mean squared deviation; ^a Method of Jorjani et al. [4] with $\epsilon = 10^{-2}$; ^b Method of Jorjani et al. [4] with $\epsilon = 10^{-4}$; ^c Method of Schaeffer [3]

RMSD, and higher weighted correlation for weighted bending compared to unweighted bending, at the cost of greater deviations, AAD and RMSD, and lower correlation coefficients.

The correlation matrix (C)

Table 3 shows deviations and correlations between \mathbf{C} and $\hat{\mathbf{C}}$ for unweighted bending, for different methods. The differences between results from different methods were marginal. Overall, the methods can be ranked from HJ03-4 being the best performer to LRS14, HJ03-2 and DB88-4. The criteria for this ranking were the mean of deviations, AAD and RMSD. Likely, DB88 with $\epsilon < 10^{-4}$ could perform as good as HJ03-4 and LRS14, because in this method, eigenvalues smaller than ϵ are replaced with $100 \times \epsilon$.

Table 3 Deviation and correlation between the upper triangle (excluding diagonal) elements of $C_{(5 \times 5)}$ (the correlation matrix) and its unweighted bent matrix

Statistics	HJ03-2 ^a	HJ03-4 ^b	LRS14 ^c	DB88-4 ^d
Min (dev.)	-0.0826	-0.0797	-0.0788	-0.1095
Max (dev.)	0.0725	0.0702	0.0717	0.0264
Mean (dev.)	-0.0200	-0.0194	-0.0204	-0.0448
AAD	0.0490	0.0475	0.0491	0.0554
Correlation	0.9841	0.9853	0.9854	0.9896
RMSD	0.0537	0.0520	0.0530	0.0622
Number of iterations	4	13	39	1

dev. = bend(\mathbf{V}) - \mathbf{V} ; AAD = average absolute deviation; RMSD = root of mean squared deviation; ^a Method of Jorjani et al. [4] with $\epsilon = 10^{-2}$; ^b Method of Jorjani et al. [4] with $\epsilon = 10^{-4}$; ^c Method of Schaeffer [3]; ^d Method of Bock et al. [7] with $\epsilon = 10^{-4}$

Table 4 Deviation and correlation between the upper triangle (excluding diagonal) elements of $C_{(5 \times 5)}$ (the correlation matrix) and its weighted (using $W_{(5 \times 5)}$) bent matrix

Statistics	HJ03-2 ^a	HJ03-4 ^b	LRS14 ^c
Min (dev.)	-0.2056	-0.1995	-0.1994
Max (dev.)	0.0644	0.0630	0.0632
Mean (dev.)	-0.0293	-0.0284	-0.0284
AAD	0.0569	0.0554	0.0555
Weighted AAD	0.0146	0.0142	0.0142
Correlation	0.9428	0.9463	0.9462
Weighted correlation	0.9939	0.9943	0.9942
RMSD	0.0828	0.0803	0.0804
Weighted RMSD	0.0110	0.0107	0.0108
Number of iterations	88	286	694

dev. = bend(\mathbf{V}) - \mathbf{V} ; AAD = average absolute deviation; RMSD = root of mean squared deviation; ^a Method of Jorjani et al. [4] with $\epsilon = 10^{-2}$; ^b Method of Jorjani et al. [4] with $\epsilon = 10^{-4}$; ^c Method of Schaeffer [3]

Table 4 shows deviations and correlations between \mathbf{C} and $\hat{\mathbf{C}}$ for weighted bending, using HJ03-2, HJ03-4 and LRS14. HJ03-4 and LRS14 performed almost equally better than HJ03-2. It took LRS14 a considerably greater number of iterations to converge, which might be a challenge for large matrices.

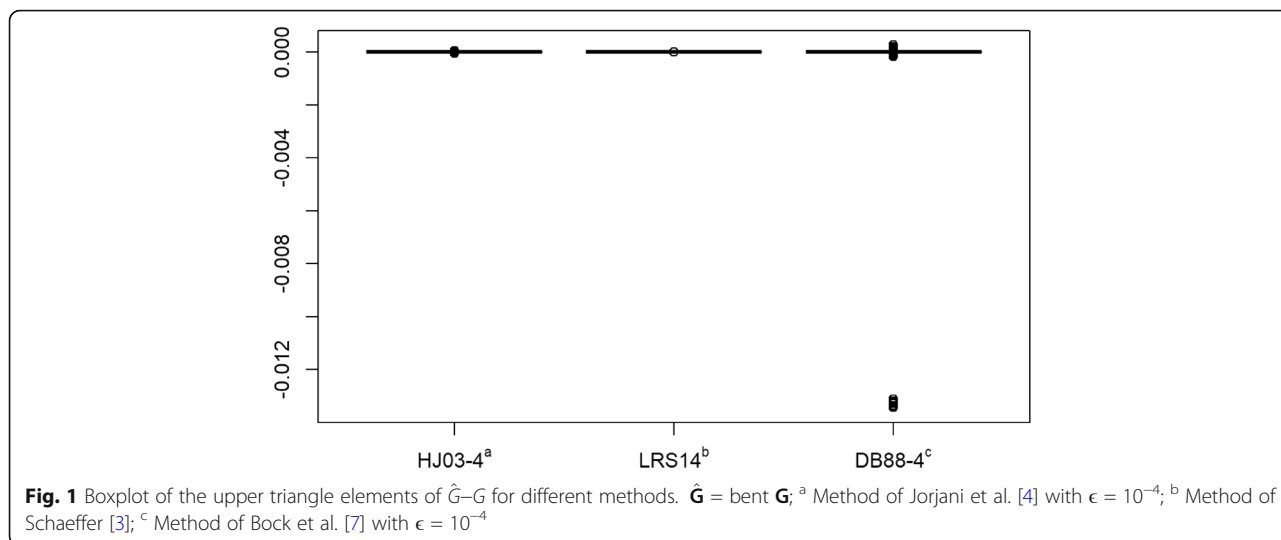
Like comparing Tables 1, 2 and S1, comparing Tables 3, 4 and S1 shows lower weighted AAD and weighted RMSD, and higher weighted correlation for weighted compared to unweighted bending, at the cost of greater deviations, AAD and RMSD, and lower correlation coefficients.

The ill-conditioned genomic relationship matrix (G)

The \mathbf{G} matrix was bent using HJ03-4, LRS14 and DB88-4. Assuming all genotypes of the same quality, unweighted bending was carried out. The distributions of the upper triangle elements of $\hat{\mathbf{G}}-\mathbf{G}$ are shown in Fig. 1, where $\hat{\mathbf{G}}$ is the bent \mathbf{G} . All the three methods performed well providing minimal deviations. DB88-4 showed larger deviations with 10 elements showing deviations between -0.0131 and -0.0135. LRS14 performed the best. The AAD values were $8.5e-15$, $1e-7$ and $1.47e-5$, and the RMSD values were $11e-15$, $4e-7$ and $6.17e-5$ for LRS14, HJ03-4 and DB88-4, respectively. It took HJ03-4, LRS14 and DB88-4, 6 s, 47 s and 3 s time to derive $\hat{\mathbf{G}}$, in 1, 8 and 1 iterations, respectively.

Discussion

The covariance (\mathbf{V}), the correlation (\mathbf{C}) and the genomic relationship (\mathbf{G}) matrices were successfully bent using different methods. In situations where the precision of different matrix elements is known (e.g., the number of



observations involved or the standard errors), weighted bending is highly recommended. It may cost an overall larger deviation between the original and the bent matrices, but minimising the deviations or preserving the elements with higher precision. The extension of LRS14 to a weighted bending worked perfectly and proved to be better than LRS14. An improvement made to HJ03 was changing W to $W/\max(W)$, internally [8]. This reduced the number of iterations to convergence, by changing $\max(W)$ to 1. This practice is recommended as elements of $[11' - W]$ and W are positive and convex combinations of each other (i.e., $V_{n+1} = V_n - [V_n - U_n \Delta_n U_n'] \circ W = V_n \circ [11' - W] + [U_n \Delta_n U_n'] \circ W$).

The unweighted bending of the covariance matrices (V and G) took an iteration to converge, except LRS14 for G , which took 8 iterations to converge. For the unweighted bending of the correlation matrix, except DB88-4, the other methods converged in more than 1 iteration. The reason was that unweighted bending for correlation matrices is equivalent to a weighted bending (except for DB88-4) with off-diagonal weights equal to 1, and diagonal weights equal to 0. As a result, correlation coefficients between the original and the bent matrix decreased from Table 1 to Table 3 (covariance to correlation), but it increased for DB88-4 from 0.9833 to 0.9896, because this method is mainly designed for bending correlation matrices. Probably for the same reason, it did not perform as good as HJ03-4 and LRS14 for bending G (Fig. 1). As a result of DB88 being designed for correlation matrices, when it comes to covariance matrices, the diagonal elements are not allowed to change, which puts more force on the off-diagonal elements to change. Contrary to weighted bending of the covariance matrix, weighted bending of the correlation matrix converged in fewer number of iterations.

The most important statistics to judge among different bending methods are absolute distance measures such as AAD and RMSD. Correlation coefficients between the original and the bent matrix are not important as such but showing the direction of changes corresponding to the value of elements in the original matrix. Where weighted bending is involved, weighted AAD and weighted RMSD should be considered.

Although, LRS14 is not based on any known statistical properties [3], it performed close to HJ03-2 (slightly better for C and slightly worse for V) and it was the best performer for G . In the first iteration of bending C by LRS14, the eigenvalues -0.0312 and -0.1852 were replaced with 0.0019 and 0.0007 , respectively. Therefore, the benefit over HJ03-2 may come from a combination of these values being in a descending (rather than equal) order and less than 10^{-2} . In the first iteration of bending V by LRS14, the eigenvalues -3.1229 and -18.5235 were replaced with 0.2036 and 0.0774 . Given these values are greater than 10^{-2} (compared to HJ03-2), the benefit in replacing negative eigenvalues with small positive values in a decreasing order becomes evident. It would be interesting to see how LRS14 performs with increasing the denominator ($100s^2 + 1$).

Other methods

There are many other bending methods used in other fields, such as psychology, economics, finance and engineering. Most of those are designed for bending correlation matrices. Therefore, applying them to covariance matrices would result in unchanged diagonal elements and consequently further changes in off-diagonal elements. Several of those methods are explained by Marée [11], and Lorenzo-Seva and Ferrando [13]. As examples, here, a few methods are explained briefly.

Rebonato and Jäkel [14] used hypersphere decomposition methodology for creating a valid correlation matrix

for the use in risk management and option pricing. In this trigonometric-based method, $\hat{\mathbf{C}} = \mathbf{B}\mathbf{B}'$ and the row vectors of \mathbf{B} are coordinates of angles (θ_{ij}) lying on a unit hypersphere. The elements of \mathbf{B} are calculated as:

$$b_{ij} = \begin{cases} \cos\theta_{ij} & \text{for } j = 1 \\ \cos\theta_{ij} \cdot \prod_{k=1}^{j-1} \sin\theta_{ik} & \text{for } j = 2 \text{ to } n-1 \\ \prod_{k=1}^{j-1} \sin\theta_{ik} & \text{for } j = n \end{cases}$$

Rapisarda et al. [15] simplified this method by reducing \mathbf{B} to a lower triangle matrix. This method resembles deriving the Cholesky decomposition of a PD correlation matrix close to the non-PD correlation matrix. The elements of \mathbf{B} are calculated as:

$$b_{ij} = \begin{cases} 1 & \text{for } i = j = 1 \\ \cos\theta_{ij} & \text{for } i \geq 2, j = 1 \\ \prod_{k=1}^{j-1} \sin\theta_{ik} & \text{for } i = j, 2 \leq i \leq n \\ \cos\theta_{ij} \prod_{k=1}^{j-1} \sin\theta_{ik} & \text{for } 2 \leq j \leq i-1 \\ 0 & \text{for } i + 1 \leq j \leq n \end{cases}$$

Numpacharoen and Atsawarungruangkit [16] introduced a method for obtaining the theoretical bounds of correlation coefficients and an algorithm for permuting random correlation matrices within those boundaries.

Bentler and Yuan [17] developed a bending method via off-diagonal scaling of the matrix. The symmetric PD matrix is obtained as $\hat{\mathbf{V}} = \Delta(\mathbf{V} - \mathbf{D}_V)\Delta + \mathbf{D}_V$, where $\mathbf{D}_V = \text{diag}(\text{diag}(\mathbf{V}))$, Δ is a diagonal matrix such that $\mathbf{0} < \Delta^2 \text{diag}(\text{diag}(\mathbf{V} - \mathbf{D})) < \mathbf{D}_V$, and \mathbf{D} is a diagonal matrix such that $\mathbf{V} - \mathbf{D}$ is PD. \mathbf{D} can be a diagonal matrix of small negative values, or according to Bentler and Yuan [17], it can be obtained by minimum trace factor analysis [18, 19].

An alternative approach to bending is fitting a reduced rank factor-analytic model. Multi-trait BLUP can be reformulated by changing the covariance structure among n traits to the factor-analytic structure of n orthogonal factors [20]. In a reduced rank factor-analytic model, specific factors (not explaining the common variance) are absent, by setting the corresponding eigenvalues to zero [20].

Obviously, there should be some confidence around matrix elements. Neither a rank reduced factor-analytic model nor bending can solve the problem of major flaws in matrix elements. For both methodologies, the reference is the original matrix. In bending, only if uncertainty around some matrix elements causes non-PDness, the matrix is bent with as minimal as possible impact on the original matrix. Similarly, when it comes to \mathbf{G} matrix, prevention (i.e., quality control and discarding problematic genotypes) is better than cure (i.e., bending).

Conclusions

This study introduced a new R package for bending symmetric non-PD matrices to PD, with the flexibility of choosing between weighted and unweighted bending, two different bending methods, the smallest positive eigenvalue (for one of the methods), and direct or reciprocal use of the weight matrix elements for weighted bending. Together with the bent matrix, several bending performance statistics are provided by the program. Where precision of matrix elements is available, weighted bending is recommended. There was benefit in small positive values in a descending order replacing negative eigenvalues. This method can further benefit from the possibility of choosing the smallest positive eigenvalue, which can be a topic for future research. The differences between the performance of different methods were minor and the methods ranked differently for different matrices. Therefore, testing different methods, and ϵ values (for HJ03) are recommended. Bending methods may perform differently for different matrices, depending on whether a covariance or a correlation matrix is being bent, the number of negative eigenvalues and their magnitude relative to the smallest positive eigenvalue, and the size of the matrix (i.e., number of diagonal elements relative to all elements).

There are many bending methods available, and those have approached the problem in various ways. Some methods are preferable for correlation matrices. This study showed that eigendecomposition-based methods are simple, robust and computationally efficient. Finally, the application of bending and R package mbend is not limited to multi-trait BLUP, but also other multivariate mixed models, genetic selection indices [5], or any situation, where a symmetric non-PD matrix needs to be transformed to a PD matrix.

Availability and requirements

- **Project name:** R package mbend
- **Project home page:** <https://cran.r-project.org/package=mbend>
- **Operating system(s):** Platform independent
- **Programming language:** R
- **Other requirements:** None
- **License:** GPL-3
- **Any restrictions to use by non-academics:** None

Supplementary information

Supplementary information accompanies this paper at <https://doi.org/10.1186/s12863-020-00881-z>.

Additional file 1 Appendix. Bending a correlation matrix using function bend from R package mbend. **Table S1.** Weighted statistics

(using $W_{(5 \times 5)}$) between the upper triangle elements of $V_{(5 \times 5)}$ (the covariance matrix) and $C_{(5 \times 5)}$ (the correlation matrix) and their unweighted bent matrices.

Abbreviations

BLUP: Best linear unbiased prediction; HJ03: Method of Jorjani et al. (2003); HJ03–2: HJ03 with $\epsilon = 10^{-2}$; HJ03–4: HJ03 with $\epsilon = 10^{-4}$; LRS14: Method of Schaeffer (2014); DB88: Method of Bock et al. (1988); DB88–4: DB88 with $\epsilon = 10^{-4}$

Acknowledgements

The author is thankful from Dr. Hossein Jorjani for fruitful discussions that led to the development of R package mbend.

Author's contributions

MAN wrote the R package and the manuscript. He is the maintainer of the R package and the corresponding author of the article. MAN read and approved the final manuscript.

Funding

Publication fund was provided by Livestock Improvement Corporation, Hamilton, New Zealand.

Availability of data and materials

The datasets and code supporting the conclusions of this article are available in the Mendeley repository: <https://doi.org/10.17632/kf3d8v8939.1>.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The author declares that he has no competing interests.

Received: 16 March 2020 Accepted: 29 June 2020

Published online: 03 September 2020

References

- Schaeffer LR. Sire and cow evaluation under multiple trait models. *J Dairy Sci.* 1984;67(7):1567–80. [https://doi.org/10.3168/jds.S0022-0302\(84\)81479-4](https://doi.org/10.3168/jds.S0022-0302(84)81479-4).
- Patterson H, Thompson R. Recovery of inter-block information when block sizes are unequal. *Biometrika.* 1971;58(3):545–54. <https://doi.org/10.2307/2334389>.
- Schaeffer, L.R.: Making covariance matrices positive definite (2014). <http://animalbiosciences.uoguelph.ca/%7EElrs/ELARES/PDforce.pdf>.
- Jorjani H, Klei L, Emanuelson U. A simple method for weighted bending of genetic (co) variance matrices. *J Dairy Sci.* 2003;86(2):677–9. [https://doi.org/10.3168/jds.S0022-0302\(03\)73646-7](https://doi.org/10.3168/jds.S0022-0302(03)73646-7).
- Hayes JF, Hill WG. Modifications of estimates of parameters in the construction of genetic selection indices ("bending"). *Biometrics.* 1981;37(3):483–93. <https://doi.org/10.2307/2530561>.
- Meyer K, Kirkpatrick M. Better estimates of genetic covariance matrices by "bending" using penalized maximum likelihood. *Genetics.* 2010;185(3):1097–110. <https://doi.org/10.1534/genetics.109.113381>.
- Bock RD, Gibbons R, Muraki E. Full-information item factor analysis. *Appl Psychol Meas.* 1988;12(3):261–80. <https://doi.org/10.1177/014662168801200305>.
- Nilforooshan, M.A.: mbend: Matrix bending. R package version 1.3.0 (2020). <https://CRAN.R-project.org/package=mbend>.
- VanRaden PM. Efficient methods to compute genomic predictions. *J Dairy Sci.* 2008;91(11):4414–23. <https://doi.org/10.3168/jds.2007-0980>.
- Revelle, W.: Procedures for Psychological, Psychometric, and Personality Research. R package version 1.9.12.31 (2020). <https://CRAN.R-project.org/package=psych>.
- Marée SC. Correcting non positive definite correlation matrices. Netherlands: BSc thesis, Department of Applied Mathematics, Delft University of Technology; 2012. <http://resolver.tudelft.nl/uuid:2175c274-ab03-4fd5-85a9-228fe421c1dbf>.

- Higham N. Computing the nearest correlation matrix – a problem from finance. *Numer Anal.* 2001;22(3):329–43. <https://doi.org/10.1093/imanum/22.3.329>.
- Lorenzo-Seva U, Ferrando PG. Not positive definite correlation matrices in exploratory item factor analysis: causes, consequences and a proposed solution. *Struct Equ Model.* 2020;27:1–10. <https://doi.org/10.1080/10705511.2020.1735393>.
- Rebonato R, Jäckel P. The most general methodology for creating a valid correlation matrix for risk management and option pricing purposes. *J Risk.* 2000;2(2):17–27. <https://doi.org/10.21314/JOR.2000.023>.
- Rapisarda F, Brigo D, Mercurio F. Parameterizing correlations: a geometric interpretation. *IMA J Manag Math.* 2007;18(1):55–73. <https://doi.org/10.1093/imaman/dpl010>.
- Numpacharoen K, Atsawarungrangkit A. Generating correlation matrices based on the boundaries of their coefficients. *PLoS One.* 2012;7(11):e48902. <https://doi.org/10.1371/journal.pone.0048902>.
- Bentler PM, Yuan K. Positive definiteness via off-diagonal scaling of a symmetric indefinite matrix. *Psychometrika.* 2011;76(1):119–23. <https://doi.org/10.1007/s11336-010-9191-3>.
- Bentler PM. A lower bound method for the dimension-free measurement of internal consistency. *Soc Sci Res.* 1972;1(4):343–57. [https://doi.org/10.1016/0049-089X\(72\)90082-8](https://doi.org/10.1016/0049-089X(72)90082-8).
- Della Riccia G, Shapiro A. Minimum rank and minimum trace of covariance matrices. *Psychometrika.* 1982;47(4):443–8. <https://doi.org/10.1007/BF02293708>.
- Meyer K. Factor-analytic models for genotype \times environment type problems and structured covariance matrices. *Genet Sel Evol.* 2009;41:21. <https://doi.org/10.1186/1297-9686-41-21>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

