

Matrix Decomposition with Kronecker Weights

Jan de Leeuw

April 27, 2026

The familiar orthogonal decomposition of a matrix into row effect, column effects, and interactions is generalized to a matrix space with the Kronecker inner product.

Table of contents

1	Kronecker Weights	4
2	Matrix Decomposition	4
2.1	Subspaces	4
2.2	Projection	5
2.2.1	First	5
2.2.2	Second	5
2.2.3	Third	6
2.2.4	Fourth	6
2.3	Pythagoras	7
2.4	Special Case	8
3	Generalizations	9
3.1	Regression	9
3.1.1	Subspaces	9
3.1.2	Orthogonality	9
3.1.3	Projection	9
3.2	Singular R and/or C	10
3.3	Multidimensional Arrays	10

3.4	Look Ma, No Coordinates	11
3.5	ML for the Matrix Variate Normal	11
4	Example	13
5	Code	16
	References	17

Note: This is a working manuscript which will be expanded/updated frequently. All suggestions for improvement are welcome. All Rmd, tex, html, pdf, R, and C files are in the public domain. Attribution will be appreciated, but is not required. The files can be found at <https://github.com/deleeuw/decomp>

1 Kronecker Weights

Suppose R and C are positive semi-definite matrices of order n and m . They define an inner product on the space $\mathbb{R}^{n \times m}$ of $n \times m$ matrices as $\langle X, Y \rangle_{RC} = \text{tr } RXY'$ and a corresponding squared norm (pseudo-norm if either R or C is singular) as $\|X\|_{RC}^2 = \langle X, X \rangle_{RC}$.

Two important alternative representations of this inner product are

$$\langle X, Y \rangle_{RC} = \text{vec}(X)'(C \otimes R)\text{vec}(Y), \quad (1)$$

or, elementwise,

$$\langle X, Y \rangle_{RC} = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^n \sum_{\ell=1}^m r_{ik} c_{j\ell} x_{ij} y_{k\ell}. \quad (2)$$

Since we will only be taking about inner products of this form we will leave out the subscript RC from now on.

2 Matrix Decomposition

2.1 Subspaces

We use e_n and e_m for vectors with n and m elements equal to one. If the number of elements is clear from the context we simply use e .

Define four subspaces of $\mathbb{R}^{n \times m}$.

1. $\mathbb{L}_\mu$, constant matrices, of the form $\mu e_n e_m'$.
2. $\mathbb{L}_a$, matrices with all columns the same, of the form $a e_n e_m'$.
3. $\mathbb{L}_b$, matrices with all rows the same, of the form $e_n b'$.
4. $\mathbb{L}_h$, matrices orthogonal to the previous three spaces.

We will choose a, b, μ and $H \in \mathbb{L}_h$ such that these four subspaces are orthogonal in the Kronecker inner product. For this we need

$$\text{tr } R\mu e e' C e a' = \mu a' R e \times e' C e = 0, \quad (3a)$$

$$\text{tr } R\mu e e' C b e' = \mu b' C e \times e' R e = 0, \quad (3b)$$

$$\text{tr } R a e' C b e' = a' R e \times b' C e = 0, \quad (3c)$$

$$\text{tr } R\mu e e' C H' = \mu e' C H' R e = 0, \quad (3d)$$

$$\text{tr } R a e' C H' = e' C H' R a = 0, \quad (3e)$$

$$\text{tr } R e b' C H' = b' C H' R e = 0. \quad (3f)$$

It follows the four subspaces are orthogonal if $a'Re = b'Ce = 0$ and HCe and $H'Re$ are zero. Thus we redefine our subspaces as

1. $\mathbb{L}_\mu$: all matrices $\mu ee'$.
2. $\mathbb{L}_a$: all matrices ae' where $a'Re = 0$.
3. $\mathbb{L}_b$: all matrices eb' where $b'Ce = 0$.
4. $\mathbb{L}_h$: all H with $HCe = 0$ and $H'Re = 0$.

2.2 Projection

Now project, in the Kronecker metric, a matrix X on each of these orthogonal subspaces.

2.2.1 First

First project on $\mathbb{L}_\mu$. We must minimize

$$\|X - \mu ee'\|^2 = \|X\|^2 - 2\mu e'RXCe + \mu^2 e'Re \times e'Ce, \quad (4)$$

over μ , with solution

$$\hat{\mu} = \frac{e'RXCe}{e'Re \times e'Ce}. \quad (5)$$

2.2.2 Second

Now project on $\mathbb{L}_a$. We have

$$\|X - ae'\|^2 = \|X\|^2 - 2a'RXCe + a'Ra \times e'Ce, \quad (6)$$

which must be minimized over all $a'Re = 0$. With a Lagrange multiplier λ the stationary equations are

$$RXCe - \lambda Re = e'Ce \times Ra. \quad (7)$$

Premultiply by e' gives

$$\hat{\lambda} = \frac{e'RXCe}{e'Re}, \quad (8)$$

and thus

$$e'Ce \times \hat{a} = XCe - e \frac{e'RXCe}{e'Re}, \quad (9)$$

or

$$\hat{a} = \left(I - \frac{ee'R}{e'Re} \right) X \left(\frac{Ce}{e'Ce} \right). \quad (10)$$

At this point it is convenient to define the projectors

$$P := \frac{ee'R}{e'Re}, \quad (11)$$

$$Q := \frac{Cee'}{e'Ce}. \quad (12)$$

The projection on $\mathbb{L}_a$ is

$$\hat{a}e' = (I - P)XQ. \quad (13)$$

In addition, from (5),

$$\hat{\mu}ee' = PXQ. \quad (14)$$

2.2.3 Third

Projecting on $\mathbb{L}_b$ is similar to projecting on $\mathbb{L}_a$, by symmetry. Thus

$$\hat{e}b' = PX(I - Q). \quad (15)$$

2.2.4 Fourth

If P and Q are arbitrary square matrices of orders n and m , not necessarily projectors, we have

$$X = PXQ + (I - P)XQ + PX(I - Q) + (I - P)X(I - Q). \quad (16)$$

For our decomposition it follows that if we define the “residual”

$$\hat{H} := X - \hat{\mu}ee' - \hat{a}e' - \hat{e}b', \quad (17)$$

then $\hat{H} = (I - P)X(I - Q)$. Because H in (17) is the residual after projecting on a subspace the orthogonality of the residual space to the row and column spaces is basically Pythagoras’ Theorem. Alternatively, however, we could have defined $\hat{H}$ directly as the projection of X on $\mathbb{L}_h$. This produces the same $\hat{H}$. Here follows a proof of this last statement.

The stationary equations for the projection, using two vectors ϕ and ψ of Lagrange multipliers, are

$$H = X - \phi e' - e \psi', \quad (18a)$$

$$HCe = 0, \quad (18b)$$

$$H'Re = 0. \quad (18c)$$

Postmultiply (18a) by Ce . Then we must have $0 = XCe - \phi e'Ce - e\psi'Ce$ and thus

$$\phi = \frac{XCe}{e'Ce} - \frac{ee'C}{e'Ce}\psi. \quad (19a)$$

In the same way

$$\psi = \frac{X'Re}{e'Re} - \frac{ee'R}{e'Re}\phi. \quad (19b)$$

Substitute (19b) into (19a). Then

$$\phi = \frac{XCe}{e'Ce} - \frac{ee'C}{e'Ce} \left\{ \frac{X'Re}{e'Re} - \frac{ee'R}{e'Re}\phi \right\}. \quad (20)$$

Collecting terms and using (9) and (11) gives $(I - P)\phi = \hat{a}$. In the same way $(I - Q)\psi = \hat{b}$. This implies that $\phi = \hat{a} + \alpha e$ and $\psi = \hat{b} + \beta e$ for some scalars α and β , so that

$$\hat{H} = X - \hat{a}e' - e\hat{b}' - (\alpha + \beta)ee'. \quad (21)$$

Premultiplying with $e'R$ and postmultiplying with Ce , and then using (18b) and (18c), gives that $\alpha + \beta = \hat{\mu}$. End of proof.

Define $\mathbb{L}_h$ as the subspace of all matrices of this form, or equivalently the matrices with $HCe = 0$ and $e'RH = 0$.

Because $(I - Q)Ce = 0$ it follows that

$$\text{tr } R\hat{H}C(\hat{\mu}ee') = 0 \quad (22a)$$

and

$$\text{tr } R\hat{H}C(e\hat{b}') = 0 \quad (22b)$$

and

$$\text{tr } R\hat{a}e'CH' = 0 \quad (22c)$$

Thus $\mathbb{L}_h$ is also orthogonal in the Kronecker inner product to the other three subspaces.

2.3 Pythagoras

It follows that in the Kronecker norm

$$\|X\|^2 = \|PXQ\|^2 + \|(I - P)XQ\|^2 + \|PX(I - Q)\|^2 + \|(I - P)X(I - Q)\|^2, \quad (23)$$

and thus we not only have a decomposition of the data X , but also of its squared Kronecker norm.

This is important when fitting Y to X , with various restrictions on Y . Suppose, for example, we require $Y = \mu ee' + ae' + eb' + H$, with the orthogonality or identifiability constraints $a'Re = b'Ce = 0$ and $HCe = 0$ and $H'Re = 0$, and possibly with additional separate constraints on μ , a , b , and H . We may require, for example, that $\text{rank}(H) \leq p$, with $p < \min(m-1, n-1)$ and that a and b are increasing. We compute the projections first and then fit Y . From (23) We can write (23) as

$$\begin{aligned} \|X - Y\|^2 &= (\hat{\mu} - \mu)^2 (e'Re)^2 (e'Ce)^2 + \\ &\quad \{(\hat{a} - a)' R(\hat{a} - a)\}^2 (e'Ce)^2 + \{(\hat{b} - b)' C(\hat{b} - b)\}^2 (e'Re)^2 + \|(\hat{H} - H)\|^2. \end{aligned} \quad (24)$$

we can minimize each of the four components separately.

In the example with monotonicity constraints on a and b and rank constraints on H that means using monotone regression to minimize $(\hat{a} - a)' R(\hat{a} - a)$ and $(\hat{b} - b)' C(\hat{b} - b)$ over increasing a and b and minimizing $\text{tr} R(\hat{H} - H)C(\hat{H} - H)'$ with $H = AB'$ over the $n \times p$ matrices A and the $m \times p$ matrices B . For this last problem the stationary equations for A and B are (ignoring the orthogonality constraints of H for the time being)

$$R\hat{H}CB = RA(B'CB), \quad (25a)$$

$$C\hat{H}'RA = CB(A'RA). \quad (25b)$$

If we define $\tilde{B} := C^{\frac{1}{2}}B$ and $\tilde{A} := R^{\frac{1}{2}}A$ then

$$\{R^{-\frac{1}{2}}\hat{H}C^{-\frac{1}{2}}\} \tilde{B} = \tilde{A}(\tilde{B}'\tilde{B}), \quad (26a)$$

$$\{C^{-\frac{1}{2}}\hat{H}'R^{-\frac{1}{2}}\} \tilde{A} = \tilde{B}(\tilde{A}'\tilde{A}), \quad (26b)$$

showing that $\tilde{A}$ and $\tilde{B}$ can be computed from the singular value decomposition of $R^{-\frac{1}{2}}\hat{H}C^{-\frac{1}{2}}$. Since $\tilde{H}Ce = 0$ we see that $C^{\frac{1}{2}}e$ is a singular vector with singular value zero, and the other singular vectors in $\tilde{B}$ can be chosen orthogonal to $C^{\frac{1}{2}}e$. Thus $\tilde{B}'C^{\frac{1}{2}}e = B'Ce = 0$, and because $H = AB'$ we have $HCe = 0$. In the same way $H'Re = 0$, and the orthogonality/identifiability conditions on H are satisfied.

2.4 Special Case

The classical special case of our results has $R = I$ and $C = I$. Then P and Q are symmetric projectors, with $P = ee'/n$ and $Q = ee'/m$. Vector $\hat{a}$ are the centered row means, $\hat{b}$ are the centered column means, $\hat{\mu}$ is the grand mean, and matrix $\hat{H}$ is the doubly-centered X . Orthogonality of the subspaces forms the basis for Factorial Analysis of Variance.

3 Generalizations

The developments in this paper can be generalized in several ways.

3.1 Regression

The vectors e_n and e_m in our formulas can be replaced by any known vectors u and v with n and m elements. In fact they can be replaced by known $n \times r$ and $m \times r$ matrices U and V of rank r . Assume without loss of generality that $U'RU = V'CV = I$.

3.1.1 Subspaces

1. $\mathbb{L}_M$: UMV' with M of dimension $r \times r$,
2. $\mathbb{L}_A$: UA with A of dimension $r \times m$,
3. $\mathbb{L}_B$: $B'V'$ with B of dimension $r \times n$,
4. $\mathbb{L}_H$: the residual space.

Thus the decomposition we are looking for is

$$X = UMV' + UA + B'V' + H, \quad (27)$$

which is a form of regressing X on U and V . Or, alternatively, the residuals H are X “corrected for” U and V .

3.1.2 Orthogonality

1. $\mathbb{L}_M \perp \mathbb{L}_A$ if $\text{tr } U' RUMV'CA' = 0$ thus if $ACV = 0$
2. $\mathbb{L}_M \perp \mathbb{L}_B$ if $\text{tr } BRUMV'CV = 0$ thus if $BRU = 0$
3. $ACV = 0$ and $BRU = 0$ already imply that $\mathbb{L}_A \perp \mathbb{L}_B$.

3.1.3 Projection

We define the two projectors $P := UU'R$ and $Q := CVV'$.

1. Projection on $\mathbb{L}_M$ using $\hat{M} = U'RXCV$ is PXQ .
2. Projection on $\mathbb{L}_A$ using $\hat{A} = U'RX(I - Q)$ is $PX(I - Q)$.
3. Projection on $\mathbb{L}_B$ using $\hat{B}' = (I - P)XC'$ is $(I - P)XQ$.
4. Projection on $\mathbb{L}_H$ is $(I - P)X(I - Q)$.

3.2 Singular R and/or C

If either R or C is singular then some of the analysis in the previous sections must be modified. Singularity is one situation in which replacing the vectors e_n and e_m by different vectors u and v may come in handy.

First consider a special, and especially serious, case of singularity. For the decomposition

$$X = \mu ee' + ae' + eb' + H \quad (28)$$

suppose $Ce = 0$. The projections on $\mathbb{L}_\mu$ and $\mathbb{L}_a$ are both zero, so that we are basically looking at the projection on $\mathbb{L}_b$ only, with no need to require orthogonality. The projection is $\hat{e}b' = PX$ and thus the residual is $\hat{H} = (I - P)X$. In the even more unfortunate case that both $Ce = 0$ and $Re = 0$ all three projections are zero and $\hat{H} = X$. Nothing lost, nothing gained,

A singularity $Cz = 0$ or $Rz = 0$ for some $z \neq e$ is less serious. The projections and the decomposition can be carried out as usual, but the projections may not be unique. A possible approach is to use eigen-decompositions $R = K\Phi K'$ and $C = L\Psi L'$, with Φ and Ψ positive definite, diagonal and of orders $\text{rank}(R)$ and $\text{rank}(C)$. Then

$$\langle X, Y \rangle_{RC} = \langle K'XL, K'YL \rangle_{\Phi\Psi}, \quad (29)$$

and we can decompose $\tilde{X} := K'XL$ uniquely as

$$\tilde{X} = \mu uv' + \tilde{a}v' + u\tilde{b}' + \tilde{H}, \quad (30)$$

with $u := K'e$, $v := L'e$, $\tilde{a} := K'a$, $\tilde{b}' := L'b$, and $\tilde{H} := K'HL$. After that transform this unique solution back to the non-unique solution for a , b , and H .

The even more singular situation in which R and C are rectangular is discussed in De Leeuw (1984) and Zha (1991).

3.3 Multidimensional Arrays

Our analysis for matrices also applies to multidimensional arrays. If the array has p “dimensions” then there are 2^p components. The squared Kronecker inner product does not have a simple matrix representation like $\text{tr} RXCX'$ any more. But, in case of $n_1 \times n_2 \times n_3$ arrays X and Y with positive semi-definite weight matrices R , C , and S of orders n_1 , n_2 and n_3 , we still have

$$\langle X, Y \rangle = \text{vec}(X)'(S \otimes C \otimes R)\text{vec}(Y) \quad (31)$$

and, elementwise,

$$\langle X, Y \rangle = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} \sum_{i'=1}^{n_1} \sum_{j'=1}^{n_2} \sum_{k'=1}^{n_3} r_{ii'} c_{jj'} s_{kk'} x_{ijk} y_{i'j'k'} \quad (32)$$

The eight subspaces are of the form $x_{ijk} = \mu$, $x_{ijk} = a_i$, $x_{ijk} = b_j$, $x_{ijk} = t_j$, $x_{ijk} = \gamma_{ij}$, $x_{ijk} = \delta_{ik}$, $x_{ijk} = \epsilon_{jk}$, and a final “residual” space. The orthogonality of the subspaces and the formulas for the projections can be worked out in the same way as for the two-dimensional case. I may do this in some other paper. Or I may not.

3.4 Look Ma, No Coordinates

Our approach to matrix space also applies to any finite sequence of orthogonal subspaces in any finite-dimensional inner product space. Suppose $\Pi_1, \dots, \Pi_t$ are projectors (linear idempotent operators) on orthogonal subspaces $\mathbb{L}_1, \dots, \mathbb{L}_t$, and $\circ$ is composition. Thus $\Pi_s = \Pi_s \circ \Pi_s$ for all s and $\Pi_s \circ \Pi_t = 0$ for all $s \neq t$. If $\Pi_r = I - (\Pi_1 + \dots + \Pi_t)$ then $\Pi_s \circ \Pi_r = 0$ for all s . Thus the residual space is orthogonal to all $\mathbb{L}_s$. The decomposition is

$$X = \Pi_1(X) + \Pi_2(X) + \dots + \Pi_r(X). \quad (33)$$

In our examples from the previous sections we have $\Pi_1(X) = PXQ$, $\Pi_2(X) = PX(I - Q)$, $\Pi_3(X) = (I - P)XQ$, and $\Pi_4(X) = (I - P)X(I - Q)$.

3.5 ML for the Matrix Variate Normal

If the $n \times m$ matrix X has the matrix variate normal (MVN) distribution with mean parameter Y and dispersion parameters R and C then the negative log likelihood is

$$\mathcal{L}(Y, R, C) = m \det(R) + n \det(C) + \text{tr } R^{-1}(X - Y)C^{-1}(X - Y)'. \quad (34)$$

There is a huge literature on the MVN distribution and its applications in environmental statistics, genetics, three-way analysis, growth curve analysis, and time series analysis. We refer to Chapter 2 in Gupta and Nagar (2000), to the edited volume of Sakata (2016), to the unpublished report of De Leeuw and Kukuljević (2009), to the MVN visualisation paper of Pocuca et al. (1923), to the recent clustering application by Berrettini et al. (2025), and to the R package matrixNormal (Hargarten (2026)). These publications, and their references, should give an idea of the scope of the MVN distribution and the separable covariances of the Kronecker inner product.

Minimizing (34) can be done using block relaxation (De Leeuw (1994)). We iteratively alternate minimization over R and C for fixed Y and minimization over Y for fixed R and C .

Minimization over R and C can be done with majorization, in this case in its EM incarnation (Dutilleul (1999), Manceur and Dutilleul (2013)). Minimization over Y for fixed R and C is the topic of this paper and of de Leeuw and Graffelman (2026). Application of our recent results to ML fitting of the MVN will have to wait for another paper.

4 Example

```
set.seed(12345)
xmat <- matrix(rnorm(12), 4, 3)
rmat <- crossprod(matrix(rnorm(40), 10, 4)) / 10
cmat <- crossprod(matrix(rnorm(30), 10, 3)) / 10
```

[1] "xmat"

	[,1]	[,2]	[,3]
[1,]	0.5855288	0.6058875	-0.2841597
[2,]	0.7094660	-1.8179560	-0.9193220
[3,]	-0.1093033	0.6300986	-0.1162478
[4,]	-0.4534972	-0.2761841	1.8173120

[1] "cmat"

	[,1]	[,2]	[,3]
[1,]	1.3744581	0.7610956	-0.2692483
[2,]	0.7610956	1.2428719	-0.1845269
[3,]	-0.2692483	-0.1845269	1.3504256

[1] "rmat"

	[,1]	[,2]	[,3]	[,4]
[1,]	0.6606540	0.6316750	0.2060831	0.4388240
[2,]	0.6316750	1.5141559	-0.7485539	0.5940988
[3,]	0.2060831	-0.7485539	2.0102613	-0.4442019
[4,]	0.4388240	0.5940988	-0.4442019	1.3318363

We now apply the `decompose()` function from the code section.

```
h<-decompose(xmat, rmat, cmat)
```

The four components are:

```
[[1]]
      [,1]      [,2]      [,3]
[1,] -0.01383449 -0.01383449 -0.01383449
[2,] -0.01383449 -0.01383449 -0.01383449
[3,] -0.01383449 -0.01383449 -0.01383449
[4,] -0.01383449 -0.01383449 -0.01383449
```

```
[[2]]
      [,1]      [,2]      [,3]
[1,] 0.2414392 -0.3254692 0.1578904
[2,] 0.2414392 -0.3254692 0.1578904
[3,] 0.2414392 -0.3254692 0.1578904
[4,] 0.2414392 -0.3254692 0.1578904
```

```
[[3]]
      [,1]      [,2]      [,3]
[1,] 0.43727229 0.43727229 0.43727229
[2,] -0.59892198 -0.59892198 -0.59892198
[3,] 0.19675185 0.19675185 0.19675185
[4,] 0.07507512 0.07507512 0.07507512
```

```
[[4]]
      [,1]      [,2]      [,3]
[1,] -0.07934821 0.50791885 -0.8654880
[2,] 1.08078326 -0.87973030 -0.4644560
[3,] -0.53365990 0.77265039 -0.4570556
[4,] -0.75617703 -0.01195553 1.5981810
```

The Kronecker inner products of the four components are:

	[,1]	[,2]	[,3]	[,4]
[1,]	6.027679e-03	5.637851e-18	-3.122502e-17	-6.938894e-18
[2,]	-1.127570e-17	8.540491e-01	-1.734723e-16	1.110223e-16
[3,]	-3.816392e-17	1.075529e-16	2.740579e+00	-1.110223e-16
[4,]	2.081668e-17	-5.551115e-17	-8.604228e-16	9.280312e+00

Check: the trace of the inner product matrix and the squared norm of the data:

[1] 12.88097 12.88097

5 Code

```
set.seed(12345)
xmat <- matrix(rnorm(12), 4, 3)
rmat <- crossprod(matrix(rnorm(40), 10, 4)) / 10
cmat <- crossprod(matrix(rnorm(30), 10, 3)) / 10

decompose <- function(xmat, rmat, cmat) {
  n <- nrow(xmat)
  m <- ncol(xmat)
  pmat <- matrix(colSums(rmat), n, n, byrow = TRUE) / sum(rmat)
  qmat <- matrix(rowSums(cmat), m, m) / sum(cmat)
  px <- pmat %*% xmat
  xq <- xmat %*% qmat
  dd <- rep(list(0), 4)
  dd[[1]] <- px %*% qmat
  dd[[2]] <- px - dd[[1]]
  dd[[3]] <- xq - dd[[1]]
  dd[[4]] <- xmat - dd[[1]] - dd[[2]] - dd[[3]]
  ip <- matrix(0, 4, 4)
  for (i in 1:4) {
    for (j in 1:4) {
      ip[i, j] <- kroneckerIP(dd[[i]], dd[[j]], rmat, cmat)
    }
  }
  ssq <- c(kroneckerIP(xmat, xmat, rmat, cmat), sum(diag(ip)))
  return(list(dd = dd, ip = ip, ssq = ssq))
}

kroneckerIP <- function(xmat, ymat, rmat, cmat) {
  return(sum(ymat * (rmat %*% xmat %*% cmat)))
}
```

References

- Berrettini, Marco, Giuliano Galimberti, and Cinzia Viroli. 2025. “Mean-Restricted Matrix-Variate Normals with an Application to Clustering.” *Statistics and Computing* 35:100: 1–20.
- De Leeuw, Jan. 1984. “Fixed-Rank Approximation with Singular Weight Matrices.” *Computational Statistics Quarterly* 1: 3–12. <https://jansweb.netlify.app/publication/deleeuw-a-84-b/deleeuw-a-84-b.pdf>.
- De Leeuw, Jan. 1994. “Block Relaxation Algorithms in Statistics.” In *Information Systems and Data Analysis*, edited by H. H. Bock, W. Lenski, and M. M. Richter. Springer Verlag. <https://jansweb.netlify.app/publication/deleeuw-c-94-c/deleeuw-c-94-c.pdf>.
- de Leeuw, Jan, and Jan Graffelman. 2026. “Matrix Approximation with Kronecker and Precision Weighting.” <https://jansweb.netlify.app/publication/deleeuw-graffelman-e-26-d/deleeuw-graffelman-e-26-d.pdf>.
- De Leeuw, Jan, and Irina Kukuyeva. 2009. “Fixed-Rank Approximation with Constraints.” <https://jansweb.netlify.app/publication/deleeuw-kukuyeva-u-09/deleeuw-kukuyeva-u-09.pdf>.
- Dutilleul, Pierre. 1999. “The MLE Algorithm for the Matrix Normal Distribution.” *Journal of Statistical Computation and Simulation* 64 (105–123).
- Gupta, Arjan K., and D. K. Nagar. 2000. *Matrix Variate Distributions*. Chapman & Hall.
- Hargarten, Paul M. 2026. *matrixNormal: The Matrix Normal Distribution*. <https://doi.org/10.32614/CRAN.package.matrixNormal>.
- Manceur, Ameer M., and Pierre Dutilleul. 2013. “Maximum Likelihood eEstimation for the Tensor Normal Distribution: Algorithm, Minimum Sample Size, and Empirical Bias and Dispersion.” *Journal of Computational and Applied Mathematics* 239: 37–49.
- Pocuca, Gallagher, Nikola, Katharine M. Clark, and Paul D. McNicholas. 1923. “Visual Assessment of Matrix-Variate Normality.” *Australian and New Zealand Journal of Statistics* 65 (2): 152–65.
- Sakata, Toshio, ed. 2016. *Applied Matrix and Tensor Variate Data Analysis*. Springer. <https://doi.org/10.1007/978-4-431-55387-8>.

Zha, Hongyuan. 1991. "The Product-Product Singular Value Decomposition of Matrix Triplets." *BIT* 31: 711–26.