Rehabilitating Correlations, Avoiding Inversion, and Extracting Roots

Nick Higham
School of Mathematics
The University of Manchester

Nicholas.J.Higham@manchester.ac.uk
@nhigham, nickhigham.wordpress.com
http://www.maths.manchester.ac.uk/~higham/talks

The Actuarial Profession
March 20, 2013, London
Precision
- Single
- Double
- Quad
- Variable

Accuracy
- Estimates
- Guarantees

Efficiency
- Speed
- Parallelism
- Energy efficiency
Outline

1. Rehabilitating Correlations
2. Matrix Inversion
3. Matrix Roots
An $n \times n$ symmetric matrix $A$ is a correlation matrix if

- It has ones on the diagonal.
- All its eigenvalues are nonnegative.
An $n \times n$ symmetric matrix $A$ is a correlation matrix if

- It has ones on the diagonal.
- All its eigenvalues are nonnegative.

Is this a correlation matrix?

$$
\begin{bmatrix}
1 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 1
\end{bmatrix}
$$
Correlation Matrix

An $n \times n$ symmetric matrix $A$ is a correlation matrix if

- It has ones on the diagonal.
- All its eigenvalues are nonnegative.

Is this a correlation matrix?

$$
\begin{bmatrix}
1 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 1
\end{bmatrix}
$$

Spectrum: $-0.4142, 1.0000, 2.4142$. 
“Given a real symmetric matrix $A$ which is almost a correlation matrix . . .

- What is the best approximating (in Frobenius norm?) correlation matrix?
- Is it unique?
- Can we compute it?

Typically we are working with $1400 \times 1400$ at the moment, but this will probably grow to $6500 \times 6500$.\)
How to Proceed

× Make ad hoc modifications to matrix: e.g., shift negative e’vals up to zero then diagonally scale.

√ Plug the gaps in the missing data, then compute an exact correlation matrix.

√ Compute the **nearest correlation matrix** in the weighted Frobenius norm \( \|A\|_2^2 = \sum_{i,j} w_i w_j a_{ij}^2 \).

*Given approx correlation matrix A find correlation matrix C to minimize \( \|A - C\| \).*

- Constraint set is a closed, convex set, so unique minimizer.
Derived theory and algorithm:


Extensions in:

Alternating Projections Algorithm


- Easy to implement.
- Guaranteed convergence, at a linear rate.
- Can add further constraints/projections.
Unexpected Applications

Some recent papers:

- **Applying stochastic small-scale damage functions to German winter storms** (2012)

- **Estimating variance components and predicting breeding values for eventing disciplines and grades in sport horses** (2012)

- **Characterisation of tool marks on cartridge cases by combining multiple images** (2012)

- **Experiments in reconstructing twentieth-century sea levels** (2011)
Computing the nearest correlation matrix

Rick Wicklin | NOVEMBER 28, 2012

Frequently someone will post a question to the SAS Support Community that says something like this:

I am trying to do [statistical task] and SAS issues an error and reports that my correlation matrix is not positive definite. What is going on and how can I complete [the task]?

The statistical task varies, but one place where this problem occurs is in simulating multivariate normal data. I have previously written about why an estimated matrix of pairwise correlations is not always a valid correlation matrix. This article discusses what to do about it. The material in this article is taken from my forthcoming book, Simulating Data with SAS.

- Applies Newton to **dual** (unconstrained) of \( \min \frac{1}{2} \| A - X \|_F^2 \) problem.
- **Globally** and **quadratically** convergent.

- H & Borsdorf (2010) improve efficiency and reliability:
  - use minres for Newton equation,
  - Jacobi preconditioner,
  - reliability improved by line search tweaks,
  - extra scaling step to ensure unit diagonal.
The Nearest Correlation Matrix

A correlation matrix is a symmetric matrix with unit diagonal and nonnegative eigenvalues. In 2000 I was approached by a London fund management company who wanted to find the nearest correlation matrix (NCM) in the Frobenius norm to an almost correlation matrix: a symmetric matrix having a significant number of (small) negative eigenvalues. This problem arises when the data from which the correlations are constructed is asynchronous or incomplete, or when models are stress-tested by artificially adjusting individual correlations. Solving the NCM problem (or obtaining a true correlation matrix some other way) is important in order to avoid subsequent calculations breaking down due to negative variances or volatilities, for example.
Original NCM Problem

- **Alternating Projections Method**
  - MATLAB: Craig Lucas, `near_cor`.
  - MATLAB: Erlend Ringstad, `validcorr`.
  - MATLAB: Nick Higham, `randcorr` (see below).
  - R: Jens Oehlschlaegel and R Matrix package authors, `nearPD`.

- **Newton Method**
  - MATLAB: Defeng Sun, `various codes`.
  - NAG Library (Fortran/SMP, C, NAG Toolbox for MATLAB, `.NET, Java, Excel, R`):
    - `go2aaf` (Fortran), `go2aac` (C), `go2aa.m` (MATLAB), `go2aa` (R),
    - `go2abf` (Fortran), `go2abc` (C), `go2ab.m` (MATLAB), `go2ab` (R): allows for weighted Frobenius norm based on two-side scaling and lower bounds on the eigenvalues.
    - `go2ajf` (Fortran), `go2ajf.m` (MATLAB) to appear in Mark 24: allows for componentwise weighted Frobenius norm and lower bounds on the eigenvalues.
## Alternating Projections vs Newton

<table>
<thead>
<tr>
<th>Matrix</th>
<th>n</th>
<th>tol</th>
<th>Code</th>
<th>Time (s)</th>
<th>Iters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Random</td>
<td>100</td>
<td>1e-10</td>
<td>g02aa</td>
<td>0.023</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>alt proj</td>
<td>0.052</td>
</tr>
<tr>
<td>2. Random</td>
<td>500</td>
<td>1e-10</td>
<td>g02aa</td>
<td>0.48</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>alt proj</td>
<td>3.01</td>
</tr>
<tr>
<td>3. Real-life</td>
<td>1399</td>
<td>1e-4</td>
<td>g02aa</td>
<td>6.8</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>alt proj</td>
<td>100.6</td>
</tr>
</tbody>
</table>
Performance of NAG Codes

The graph illustrates the time (in seconds) required for different matrix computations as a function of matrix size (n). The data is categorized into four groups: FL22, CL09, FL23, and "Coming soon (est)". The graph shows a significant improvement for larger matrix sizes, with a note indicating a 2 x Improvement.

Correlations  Matrix Inversion  Matrix Roots
Factor Model (1)

\[ \xi = X\eta + F\varepsilon, \quad \eta_i, \varepsilon_i \in N(0, 1), \]

where \( \text{var}(\xi_i) \equiv 1 \), \( F = \text{diag}(f_{ii}) \). Implies

\[ \sum_{j=1}^{k} x_{ij}^2 \leq 1, \quad i = 1: n. \]

- “Multifactor normal copula model”.
- Collateralized debt obligations (CDOs).
- Multivariate time series.
Yields correlation matrix of form

\[ C(X) = D + XX^T = D + \sum_{j=1}^{k} x_j x_j^T, \]

\[ D = \text{diag}(I - XX^T), \quad X = [x_1, \ldots, x_k]. \]

\( C(X) \) has \( k \) factor correlation matrix structure.

\[ C(X) = \begin{bmatrix}
1 & y_1^T y_2 & \cdots & y_1^T y_n \\
y_1^T y_2 & 1 & \cdots & \vdots \\
\vdots & \ddots & \ddots & y_{n-1}^T y_n \\
y_1^T y_n & \cdots & y_{n-1}^T y_n & 1
\end{bmatrix}, \quad y_i \in \mathbb{R}^k. \]
Factor Structure

Nearest correlation matrix \textbf{with factor structure}.

- \textit{Principal factors method} (Andersen et al., 2003) has no convergence theory and can converge to an incorrect answer.
Nearest correlation matrix with factor structure.

- *Principal factors method* (Andersen et al., 2003) has no convergence theory and can converge to an incorrect answer.

- Algorithm based on *spectral projected gradient method* (Borsdorf, H & Raydan, 2010).
  - Respects the constraints, exploits their convexity, and converges to a feasible stationary point.
  - NAG routine *g02aef* (Mark 23, 2012).
Variations

- Specific rank or bound on correlations.
- Block structure.
- Bounds on individual correlations.
- Other requirements?
Outline

1. Rehabilitating Correlations
2. Matrix Inversion
3. Matrix Roots
Avoiding Inversion

**Fundamental Tenet of Numerical Analysis**

Don’t invert matrices.
**Avoiding Inversion**

### Fundamental Tenet of Numerical Analysis

Don’t invert matrices.

- **Ax = b**: use Gaussian elimination (with pivoting), not \( x = A^{-1}b \).
- **\( x^T A^{-1} y = x^T (A^{-1} y) \).**
- **\( (A^{-1})_{ii} = e_i^T A^{-1} e_i \).**
- **\( \text{trace}(A^{-1}) \approx m^{-1} \sum_{k=1}^{m} v_k^T A^{-1} v_k \),**
  \( v_k \sim \text{uniform}\{-1, 1\} \). (Bekas et al., 2007)
How to Invert (1)

- Use a condition estimator to warn when $A$ is nearly singular.

```matlab
>> A = hilb(16);
>> X = inv(A);
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate.
RCOND = 9.721674e-19.
```
To compute variance–covariance matrix \((X^T X)^{-1}\) of least squares estimator, use QR factorization \(X = QR\) \(((X^T X)^{-1} = R^{-1} R^{-T})\) and do not explicitly form \(X^T X\).

Quality of computed inverse \(\hat{Y} \approx A^{-1}\) measured by

\[
\frac{\|\hat{Y} A - I\|}{\|\hat{Y}\| \|A\|} \quad \text{or} \quad \frac{\|A\hat{Y} - I\|}{\|\hat{Y}\| \|A\|}
\]

but not both.
Computing the Sample Variance

Sample variance of \( x_1, \ldots, x_n \):

\[
s^2_n = \frac{1}{n - 1} \sum_{i=1}^{n} (x_i - \bar{x})^2, \quad \text{where} \quad \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i. \quad (1)
\]

Can compute using one-pass formula:

\[
s^2_n = \frac{1}{n - 1} \left( \sum_{i=1}^{n} x_i^2 - \frac{1}{n} \left( \sum_{i=1}^{n} x_i \right)^2 \right). \quad (2)
\]
Computing the Sample Variance

Sample variance of $x_1, \ldots, x_n$:

$$s_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2, \text{ where } \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i. \quad (1)$$

Can compute using one-pass formula:

$$s_n^2 = \frac{1}{n-1} \left( \sum_{i=1}^{n} x_i^2 - \frac{1}{n} \left( \sum_{i=1}^{n} x_i \right)^2 \right). \quad (2)$$

For $x = (10000, 10001, 10002)$ using 8-digit arithmetic, (1) gives: 1.0, (2) gives 0.0.

(2) can even give negative results in floating point arithmetic!
Note:
The sample standard deviation $\sigma_{n-1}$ is defined as
\[ \sqrt{\frac{\sum x^2 - (\sum x)^2}{n-1}} \]

Nota:
La desviación estándar de muestra $\sigma_{n-1}$ se define como
\[ \sqrt{\frac{\sum x^2 - (\sum x)^2}{n-1}} \]

the population standard deviation $\sigma_n$ is defined as
la desviación estándar de población $\sigma_n$ se define como
\[ \sqrt{\frac{\sum x^2 - (\sum x)^2}{n}} \]
Spreadsheets in the Cloud

Standard deviation of $x = [n, \; n + 1, \; n + 2]^T$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>Exact</th>
<th>Google Sheet</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^7$</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$10^8$</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Spreadsheets in the Cloud

Standard deviation of \( x = [n, n + 1, n + 2]^T \).

<table>
<thead>
<tr>
<th>( n )</th>
<th>Exact</th>
<th>Google Sheet</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 10^7 )</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( 10^8 )</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Nick Higham

How Accurate Are Spreadsheets in the Cloud?

For a vector \( x \) with \( n \) elements the sample variance is \( s_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \), where the sample mean is \( \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \). An alternative formula often given in textbooks is \( s_n^2 = \frac{1}{n-1} \left( \sum_{i=1}^{n} x_i^2 - \frac{1}{n} \left( \sum_{i=1}^{n} x_i \right)^2 \right) \). This second formula has the advantage that it can be computed with just one pass through the data, whereas the first formula requires two passes. However, the one-pass formula can suffer damaging subtractive cancellation, making it numerically unstable. When I wrote my book *Accuracy and Stability of Numerical Algorithms* I found that several pocket calculators appeared to use the one-pass formula.
Correlations Matrix Inversion Matrix Roots

Spreadsheets in the Cloud

Standard deviation of $x = [n, n + 1, n + 2]^T$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>Exact</th>
<th>Google Sheet</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^7$</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$10^8$</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Nick Higham

How Accurate Are Spreadsheets in the Cloud?

For a vector $x$ with $n$ elements the sample variance is $s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2$, where the sample mean is $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$. An alternative formula often given in textbooks is $s^2 = \frac{1}{n} \left( \sum_{i=1}^{n} x_i^2 - \frac{1}{n} \left( \sum_{i=1}^{n} x_i \right)^2 \right)$. This second formula has the advantage that it can be computed with just one pass through the data, whereas the first formula requires two passes. However, the one-pass formula can suffer damaging subtractive cancellation, making it numerically unstable. When I wrote my book Accuracy and Stability of Numerical Algorithms I found that several pocket calculators appeared to use the one-pass formula.

McCullough & Yalta (2013)
Outline

1. Rehabilitating Correlations
2. Matrix Inversion
3. Matrix Roots
Term “matrix” coined in 1850 by James Joseph Sylvester, FRS (1814–1897).

Matrix algebra developed by Arthur Cayley, FRS (1821–1895).
Memoir on the Theory of Matrices (1858).
Cayley and Sylvester on Matrix Functions

- Cayley considered matrix square roots in his 1858 memoir.


- Sylvester (1883) gave first definition of $f(A)$ for general $f$.

Let vectors $v_{2011}$, $v_{2010}$ represent risks, credit ratings or stock prices in 2011 and 2010.

Assume a Markov model $v_{2011} = P v_{2010}$, where $P$ is a transition probability matrix.

$P^{1/2}$ enables predictions to be made at 6-monthly intervals.
Matrix Roots in Markov Models

- Let vectors $v_{2011}$, $v_{2010}$ represent risks, credit ratings or stock prices in 2011 and 2010.
- Assume a Markov model $v_{2011} = P v_{2010}$, where $P$ is a transition probability matrix.
- $P^{1/2}$ enables predictions to be made at 6-monthly intervals.

$P^{1/2}$ is matrix $X$ such that $X^2 = P$. What are $P^{2/3}$, $P^{0.9}$?

$$P^s = \exp(s \log P).$$

**Problem:** $\log P$, $P^{1/k}$ may have wrong sign patterns $\Rightarrow$ “regularize”.
Chronic Disease Example

- Estimated 6-month transition matrix.
- Four AIDS-free states and 1 AIDS state.
- 2077 observations (Charitos et al., 2008).

\[
P = \begin{bmatrix}
0.8149 & 0.0738 & 0.0586 & 0.0407 & 0.0120 \\
0.5622 & 0.1752 & 0.1314 & 0.1169 & 0.0143 \\
0.3606 & 0.1860 & 0.1521 & 0.2198 & 0.0815 \\
0.1676 & 0.0636 & 0.1444 & 0.4652 & 0.1592 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}.
\]

Want to estimate the 1-month transition matrix.

\[
\Lambda(P) = \{1, 0.9644, 0.4980, 0.1493, -0.0043\}.
\]

- Lin (2011, for survey of regularization methods.)
MATLAB: Arbitrary Powers

>> A = [1 1e-8; 0 1]
A =
    1.0000e+000   1.0000e-008
    0   1.0000e+000

>> A^0.1
ans =
    1     0
    0     1

>> expm(0.1*logm(A))
ans =
    1.0000e+000   1.0000e-009
    0   1.0000e+000

New backward-error based **inverse scaling and squaring alg** for matrix logarithm (Al-Mohy, H & Relton, 2012)—faster and more accurate.

Alternative Newton-based algorithms available for $A^{1/q}$ with $q$ an integer, e.g., for

$$X_{k+1} = \frac{1}{q} [(q + 1)X_k - X_k^{q+1} A], \quad X_0 = A,$$

$X_k \rightarrow A^{-1/q}$. 
Knowledge Transfer Partnership

- University of Manchester and NAG (2010–2013) funded by EPSRC, NAG and TSB.
- Developing suite of NAG Library codes for matrix functions.
- Extensive set of new codes included in Mark 23 (2012), Mark 24 (2013).
- Improvements to existing state of the art: faster and more accurate.

*My work also supported by €2M ERC Advanced Grant.*
The Matrix Square Root, Blocking and Parallelism

NAG recently embarked on a ‘Knowledge Transfer Partnership’ with the University of Manchester to introduce matrix function capabilities into the NAG Library. As part of this collaboration, Nick Higham (University of Manchester), Rui Ralha (University of Minho, Portugal) and I have been investigating how blocking can be used to speed up the computation of matrix square roots.

There is plenty of interesting mathematical theory concerning matrix square roots, but for now we’ll just use the definition that a matrix $X$ is a square root of $A$ if $X^2 = A$. Matrix roots have applications in finance and population modelling, where transition matrices are used to describe the evolution of a system from over a certain time interval, $t$. The square root of a transition matrix can be used to describe the evolution for the interval $t/2$. The matrix square root also forms a key part of the algorithms used to compute other matrix functions.

To find a square root of a matrix, we start by computing a Schur decomposition. The square root $U$ of the resulting upper triangular matrix $T$ can then be found via a simple recurrence over the elements $U_{ij}$ and $T_{ij}$:

\[
U_{ii} = \sqrt{T_{ii}},
\]

\[
U_{ii}U_{ij} + U_{ij}U_{jj} = T_{ij} - \sum_{k=i+1}^{j-1} U_{ik}U_{kj}.
\]
Matrix Functions in Parallel

Last year I wrote a blog post about NAG’s work on parallelising the computation of the matrix square root. More recently, as part of our Matrix Functions Knowledge Transfer Partnership with the University of Manchester, we’ve been investigating parallel implementations of the Schur-Parlett algorithm [1].

Most algorithms for computing functions of matrices are tailored for a specific function, such as the matrix exponential or the matrix square root. The Schur-Parlett algorithm is much more general; it will work for any “well behaved” function (this general term can be given a more mathematically precise meaning). For a function such as

\[ f(A) = e^A + \sin 2A - \cosh 4A, \]
Fast moving developments in numerical linear algebra algorithms.

Numerical reliability is essential.

Partnership with NAG enables rapid inclusion of our algorithms in the NAG Library.

Keen to hear about your matrix problems.


N. J. Higham.

*Accuracy and Stability of Numerical Algorithms.*
ISBN 0-89871-521-0.
xxx+680 pp.

N. J. Higham.
Computing the nearest correlation matrix—A problem from finance.
N. J. Higham and L. Lin.
On $p$th roots of stochastic matrices.

N. J. Higham and L. Lin.
An improved Schur–Padé algorithm for fractional powers of a matrix and their Fréchet derivatives.
20 pp.
L. Lin.  
*Roots of Stochastic Matrices and Fractional Matrix Powers.*  
117 pp.  
MIMS EPrint 2011.9, Manchester Institute for Mathematical Sciences.

B. D. McCullough and A. T. Yalta.  
Spreadsheets in the cloud—not ready yet.  
*James Joseph Sylvester. Jewish Mathematician in a Victorian World.*
xiii+461 pp.

H.-D. Qi and D. Sun.  
A quadratically convergent Newton method for computing the nearest correlation matrix.  