‘From Statistics to Data Mining’
Computer Lab Session n° 5: Linear Algebra (2/2)

Master 1 COSI / CPS^2
Saint-Étienne, France

Fabrice Muhlenbach

Laboratoire Hubert Curien, UMR CNRS 5516
Université Jean Monnet de Saint-Étienne
18 rue du Professeur Benoît Lauras
42000 SAINT-ÉTIENNE, FRANCE
https://perso.univ-st-etienne.fr/muhlfabr/

Outcome

The objective of this lab is to become familiar with R functions for working with linear algebra, especially for computing eigenvalues and eigenvectors.

1 Eigenvalues and Eigenvectors

As seen during the previous lab session, Hilbert matrices are often studied in numerical linear algebra because they are easy to construct but have surprising properties. We will start with the 3×3 Hilbert matrix:

```
H3 <- matrix(c(1, 1/2, 1/3, 1/2, 1/3, 1/4, 1/3, 1/4, 1/5), nrow=3)
H3
```

Eigenvalues and eigenvectors can be computed using the function eigen(). For example,

```
> eigen(H3)
$values
[1] 1.40831893 0.12232707 0.00268734

$vectors
          [,1]       [,2]       [,3]
[1,] 0.8270449  0.5474484  0.1276593
[2,] 0.4598639 -0.5282902 -0.7137469
[3,] 0.3232984 -0.6490067  0.6886715
```

To see what this output means, let x_1 denote the first column of the $vectors output, i.e. $[0.827 \;\; 0.459 \;\; 0.323]^T$. This is the first eigenvector, and it corresponds to the eigenvalue 1.408.

Thus, $H_3 x_1 = 1.408 \; x_1$.

Denoting the second and third columns of $vectors by x_2 and x_3, we have $H_3 x_2 = 0.122 \; x_2$, and $H_3 x_3 = 0.00268 \; x_3$.

Exercises

1. Calculate the matrix $H = X(X^T X)^{-1} X^T$, where X is defined by:

```
X <- matrix(c(1, 2, 3, 1, 4, 9), ncol=2)
```

2. Calculate the eigenvalues and eigenvectors of H.
3. Calculate the trace of the matrix H, and compare with the sum of the eigenvalues.
4. Calculate the determinant of the matrix H, and compare with the product of the eigenvalues.
5. Verify that the columns of X and $I - H$ are eigenvectors of H, here $HX = X$ and $H(I - H) = 0$.

2 Advanced Topics

2.1 The Singular Value Decomposition of a Matrix

The singular value decomposition of a square matrix A consists of three square matrices, U, D, and V. The matrix D is a diagonal matrix. The relation among these matrices is $A = UDV^T$.

The matrices U and V are said to be *orthogonal*, which means that $U^{-1} = U^T$ and $V^{-1} = V^T$.

The singular value decomposition of a matrix is often used to obtain accurate solutions to linear systems of equations.

The elements of D are called the *singular values* of A. Note that $A^T A = V D^2 V^{-1}$. This is a "similarity transformation" which tells us that the squares of the singular values of A are the eigenvalues of $A^T A$.

The singular value decomposition can be obtained using the function svd(). For example, the singular value decomposition of the 3×3 Hilbert matrix H_3 is

```
> H3.svd <- svd(H3)
> H3.svd
$d
[1] 1.40831893 0.12232707 0.00268734

$u
           [,1]       [,2]       [,3]
[1,] -0.8270449  0.5474484  0.1276593
[2,] -0.4598639 -0.5282902 -0.7137469
[3,] -0.3232984 -0.6490067  0.6886715

$v
           [,1]       [,2]       [,3]
[1,] -0.8270449  0.5474484  0.1276593
[2,] -0.4598639 -0.5282902 -0.7137469
```

```
[3 ,]  −0.3232984  −0.6490067   0.6886715
```

We can verify that these components can be multiplied in the appropriate way to reconstruct H_3:

```
> H3.svd$u %*% diag(H3.svd$d) %*% t(H3.svd$v)
          [,1]      [,2]      [,3]
[1 ,] 1.0000000 0.5000000 0.3333333
[2 ,] 0.5000000 0.3333333 0.2500000
[3 ,] 0.3333333 0.2500000 0.2000000
```

Because of the properties of the U, V and D matrices, the singular value decomposition provides a simple way to compute a matrix inverse.

For example, $H_3^{-1} = VD^{-1}U^T$ and can be recalculated as

```
> H3.svd$v %*% diag(1/H3.svd$d) %*% t(H3.svd$u)
      [,1] [,2] [,3]
[1 ,]    9  −36   30
[2 ,]  −36  192 −180
[3 ,]   30 −180  180
```

2.2 R Function apply()

In statistical applications, it is sometimes necessary to apply the same function to each of the rows of a matrix, or to each of the columns. A for() loop could be used in R, but it is sometimes more efficient computationally to use the apply() function.

There are three arguments. The first specifies the matrix. The second specifies whether the operation is to be applied to rows (1) or columns (2). The third argument specifies the function that should be applied. A simple example is to compute the sum of the rows of H_3:

```
> apply(H3, 1, sum)
[1] 1.8333333 1.0833333 0.7833333
```

Exercises

1. Consider the following *circulant* matrix:

$$P = \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 \\ 0.4 & 0.1 & 0.2 & 0.3 \\ 0.3 & 0.4 & 0.1 & 0.2 \\ 0.2 & 0.3 & 0.4 & 0.1 \end{bmatrix}$$

   - P is an example of a stochastic matrix. Use the apply() function to verify that the row sums add to 1.
   - Compute P^n for $n = 2, 3, 5, 10$. Is a pattern emerging?

2. An insurance company has four types of policies, which we will label A, B, C, and D.
   - They have a total of 245 921 policies.
   - The annual income from each policy is 10 € for type A, 30 € for type B, 50 € for type C, and 100 € for type D.
   - The total annual income for all policies is 7 304 620 €.
   - The claims on these policies arise at different rates. The expected number of type A claims is 0.1 claims per year, type B 0.15 claims per year, type C 0.03 claims per year, and type D 0.5 claims per year.
   - The total expected number of claims for the company is 34 390.48 per year.
   - The expected size of the claims is different for each policy type. For type A, it is 50 €, for type B it is 180 €, for type C it is 1500 €, and for type D it is 250 €.
   - The expected total claim amount is 6 864 693 €. This is the sum over all policies of the expected size of claim times the expected number of claims in a year.

   Use R to answer the following questions:

   (a) Find the total number of each type of policy.

   (b) Find the total income and total expected claim size for each type of policy.

3 Recommended Readings

- Braun and Murdoch (2007), "A First Course in Statistical Programming with R,"
  Chapter 6 "Computational linear algebra."
- Ciarlet et al. (1989), "Introduction to numerical linear algebra and optimisation"
- Lay (2012), "Linear Algebra and Its Applications,"
  Chapter 2 "Matrix Algebra,"
  Chapter 3 "Determinants,"
  Chapter 5 "Eigenvalues and Eigenvectors" and
  Chapter 7 "Symmetric Matrices and Quadratic Forms."
- Venables et al. (2013), "An Introduction to R,"
  Chapter 5 "Arrays and matrices."
- Teetor (2011), "R Cookbook",
  Chapter 13 "Beyond Basic Numerics and Statistics."

References

Braun, W. J. and D. J. Murdoch (2007). *A First Course in Statistical Programming with R*. Cambridge University Press.

Ciarlet, P. G., B. Miara, and J.-M. Thomas (1989). *Introduction to numerical linear algebra and optimisation*. Cambridge texts in applied mathematics. Cambridge University Press.

Lay, D. C. (2012). *Linear Algebra and Its Applications* (4^{th} ed.). Addison-Wesley.

Teetor, P. (2011). *R Cookbook*. O'Reilly.

Venables, W. N., D. M. Smith, and the R Core Team (2013). An introduction to R –notes on R: A programming environment for data analysis and graphics.
URL `http://cran.r-project.org/doc/manuals/R-intro.html`.