

Crash Course in Matrix Algebra

Sven Otto

October 9, 2023

Table of contents

Welcome	4
Accompanying R scripts	4
Comments	4
1 Basic definitions	5
1.1 Scalar, vector, and matrix	5
1.2 Some specific matrices	6
1.3 Transposition	7
1.4 Matrices in R	7
2 Sums and Products	9
2.1 Matrix summation	9
2.2 Scalar-matrix multiplication	9
2.3 Element-by-element operations in R	10
2.4 Vector-vector multiplication	11
2.4.1 Inner product	11
2.4.2 Outer product	11
2.4.3 Vector multiplication in R	12
2.5 Matrix-matrix multiplication	12
3 Rank and inverse	15
3.1 Linear combination	15
3.2 Column rank	15
3.3 Nonsingular matrix	17
3.4 Determinant	17
3.5 Inverse matrix	18
4 Advanced concepts	20
4.1 Trace	20
4.2 Idempotent matrix	21
4.3 Eigendecomposition	21
4.3.1 Eigenvalues	21
4.3.2 Eigenvectors	22
4.3.3 Spectral decomposition	22
4.3.4 Eigendecomposition in R	22
4.4 Definite matrix	23

4.5	Cholesky decomposition	24
4.6	Vectorization	25
4.7	Kronecker product	25
4.8	Vector and matrix norm	26
5	Matrix calculus	27
5.1	Gradient	27
5.2	Hessian	27
5.3	Optimization	28
6	Problems	29
	Problem 1	29
	Problem 2	29
	Problem 3	29
	Problem 4	30
	Problem 5	30
	Problem 6	30
6.1	Solutions	30

Welcome

Due to the multivariate character of many econometric topics, matrix algebra is a commonly used tool in modern econometrics. It provides a powerful and efficient framework for representing and manipulating systems of linear equations. This short lecture note series provides a brief introduction to the most relevant matrix algebra concepts for econometricians and their implementation in R.

To learn R or refresh your skills, please check out my tutorial [Getting Started With R](#).

Accompanying R scripts

All R codes of the different sections can be found here:

- [matrix-sec1.R](#)
- [matrix-sec2.R](#)
- [matrix-sec3.R](#)
- [matrix-sec4.R](#)

Comments

Feedback is welcome. If you notice any typos or issues, please report them on [GitHub](#) or email me at sven.otto@uni-koeln.de.

1 Basic definitions

Let's start with some basic definitions and specific examples.

1.1 Scalar, vector, and matrix

A **scalar** a is a single real number. We write $a \in \mathbb{R}$.

A **vector** \mathbf{a} of length k is a $k \times 1$ list of real numbers

$$\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix}.$$

By default, when we refer to a vector, we always mean a column vector. We write $\mathbf{a} \in \mathbb{R}^k$. The value a_i is called i -th entry or i -th component of \mathbf{a} . A scalar is a vector of length 1. A row vector of length k is written as $\mathbf{b} = (b_1, \dots, b_k)$.

A **matrix** \mathbf{A} of order $k \times m$ is a rectangular array of real numbers

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{km} \end{pmatrix}$$

with k rows and m columns. We write $\mathbf{A} \in \mathbb{R}^{k \times m}$. The value a_{ij} is called (i, j) -th entry or (i, j) -th component of \mathbf{A} . We also use the notation $(\mathbf{A})_{i,j}$ to denote the (i, j) -th entry. A vector of length k is a $k \times 1$ matrix. A row vector of length k is a $1 \times k$ matrix. A scalar is a matrix of order 1×1 .

We may describe a matrix \mathbf{A} by its column or row vectors as

$$\mathbf{A} = (\mathbf{a}_1 \quad \mathbf{a}_2 \quad \dots \quad \mathbf{a}_m) = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix},$$

where

$$\mathbf{a}_i = \begin{pmatrix} a_{1i} \\ \vdots \\ a_{ki} \end{pmatrix}$$

is the i -th column of \mathbf{A} and $\boldsymbol{\alpha}_i = (a_{i1}, \dots, a_{im})$ is the i -th row.

1.2 Some specific matrices

A matrix is called **square matrix** if the numbers of rows and columns coincide (i.e., $k = m$).

$$\mathbf{B} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

is a square matrix. A square matrix is called **diagonal matrix** if all off-diagonal elements are zero.

$$\mathbf{C} = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}$$

is a diagonal matrix. We also write $\mathbf{C} = \text{diag}(1, 4)$. A square matrix is called **upper triangular** if all elements below the main diagonal are zero, and **lower triangular** if all elements above the main diagonal are zero. Examples of an upper triangular matrix \mathbf{D} and a lower triangular matrix \mathbf{E} are

$$\mathbf{D} = \begin{pmatrix} 1 & 2 \\ 0 & 4 \end{pmatrix}, \quad \mathbf{E} = \begin{pmatrix} 1 & 0 \\ 3 & 4 \end{pmatrix}.$$

The $k \times k$ diagonal matrix

$$\mathbf{I}_k = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} = \text{diag}(1, \dots, 1)$$

is called **identity matrix** of order k . The $k \times m$ matrix

$$\mathbf{0}_{k \times m} = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{pmatrix}$$

is called **zero matrix**. The **zero vector** of length k is

$$\mathbf{0}_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}.$$

If the order becomes clear from the context, we omit the indices and write \mathbf{I} for the identity matrix and $\mathbf{0}$ for the zero matrix or zero vector.

1.3 Transposition

The **transpose** \mathbf{A}' of the matrix \mathbf{A} is obtained by flipping rows and columns on the main diagonal:

$$\mathbf{A}' = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{k1} \\ a_{12} & a_{22} & \cdots & a_{k2} \\ \vdots & \vdots & & \vdots \\ a_{1m} & a_{2m} & \cdots & a_{km} \end{pmatrix}.$$

If \mathbf{A} is a matrix of order $k \times m$, then \mathbf{A}' is a matrix of order $m \times k$. *Example:*

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{pmatrix} \Rightarrow \mathbf{A}' = \begin{pmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \end{pmatrix}$$

The definition implies that transposing twice produces the original matrix:

$$(\mathbf{A}')' = \mathbf{A}.$$

The transpose of a (column) vector is a row vector:

$$\mathbf{a}' = (a_1, \dots, a_k)$$

A **symmetric matrix** is a square matrix \mathbf{A} with $\mathbf{A}' = \mathbf{A}$. An example of a symmetric matrix is

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}.$$

1.4 Matrices in R

Let's define some matrices in R:

```
A = matrix(c(1,4,7,2,5,8), nrow = 3, ncol = 2)
A
```

```
      [,1] [,2]
[1,]    1    2
[2,]    4    5
[3,]    7    8
```

```
t(A) #transpose of A
```

```
      [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
```

```
A[3,2] #the (3,2)-entry of A
```

```
[1] 8
```

```
B = matrix(c(1,2,2,4), nrow = 2, ncol = 2) # another matrix
all(B == t(B)) #check whether B is symmetric
```

```
[1] TRUE
```

```
diag(c(1,4)) #diagonal matrix
```

```
      [,1] [,2]
[1,]    1    0
[2,]    0    4
```

```
diag(1, nrow = 3) #identity matrix
```

```
      [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

```
matrix(0, nrow=2, ncol=5) #matrix of zeros
```

```
      [,1] [,2] [,3] [,4] [,5]
[1,]    0    0    0    0    0
[2,]    0    0    0    0    0
```

```
dim(A) #number of rows and columns
```

```
[1] 3 2
```


2 Sums and Products

2.1 Matrix summation

Let \mathbf{A} and \mathbf{B} both be matrices of order $k \times m$. Their sum is defined componentwise:

$$\mathbf{A} + \mathbf{B} = \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1m} + b_{1m} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2m} + b_{2m} \\ \vdots & \vdots & & \vdots \\ a_{k1} + b_{k1} & a_{k2} + b_{k2} & \cdots & a_{km} + b_{km} \end{pmatrix}.$$

Only two matrices of the same order can be added. *Example:*

$$\mathbf{A} = \begin{pmatrix} 2 & 0 \\ 1 & 5 \\ 3 & 2 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -1 & 1 \\ 7 & 1 \\ -5 & 2 \end{pmatrix}, \quad \mathbf{A} + \mathbf{B} = \begin{pmatrix} 1 & 1 \\ 8 & 6 \\ -2 & 4 \end{pmatrix}.$$

The matrix summation satisfies the following rules:

- (i) $\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$ (commutativity)
- (ii) $(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$ (associativity)
- (iii) $\mathbf{A} + \mathbf{0} = \mathbf{A}$ (identity element)
- (iv) $(\mathbf{A} + \mathbf{B})' = \mathbf{A}' + \mathbf{B}'$ (transposition)

2.2 Scalar-matrix multiplication

The product of a $k \times m$ matrix \mathbf{A} with a scalar $\lambda \in \mathbb{R}$ is defined componentwise:

$$\lambda \mathbf{A} = \begin{pmatrix} \lambda a_{11} & \lambda a_{12} & \cdots & \lambda a_{1n} \\ \lambda a_{21} & \lambda a_{22} & \cdots & \lambda a_{2n} \\ \vdots & \vdots & & \vdots \\ \lambda a_{m1} & \lambda a_{m2} & \cdots & \lambda a_{mn} \end{pmatrix}.$$

Example:

$$\lambda = 2, \quad \mathbf{A} = \begin{pmatrix} 2 & 0 \\ 1 & 5 \\ 3 & 2 \end{pmatrix}, \quad \lambda \mathbf{A} = \begin{pmatrix} 4 & 0 \\ 2 & 10 \\ 6 & 4 \end{pmatrix}.$$

Scalar-matrix multiplication satisfies the distributivity law:

- (i) $\lambda(\mathbf{A} + \mathbf{B}) = \lambda\mathbf{A} + \lambda\mathbf{B}$
- (ii) $(\lambda + \mu)\mathbf{A} = \lambda\mathbf{A} + \mu\mathbf{A}$

2.3 Element-by-element operations in R

Basic arithmetic operations work on an element-by-element basis in R:

```
A = matrix(c(2,1,3,0,5,2), ncol=2)
B = matrix(c(-1,7,-5,1,1,2), ncol=2)
A+B #matrix summation
```

```
      [,1] [,2]
[1,]     1     1
[2,]     8     6
[3,]    -2     4
```

```
A-B #matrix subtraction
```

```
      [,1] [,2]
[1,]     3    -1
[2,]    -6     4
[3,]     8     0
```

```
2*A #scalar-matrix product
```

```
      [,1] [,2]
[1,]     4     0
[2,]     2    10
[3,]     6     4
```

```
A/2 #division of entries by 2
```

```
      [,1] [,2]
[1,]   1.0   0.0
[2,]   0.5   2.5
[3,]   1.5   1.0
```

```
A*B #element-wise multiplication
```

```
      [,1] [,2]  
[1,]   -2   0  
[2,]    7   5  
[3,]  -15   4
```

2.4 Vector-vector multiplication

2.4.1 Inner product

The **inner product** (also known as dot product) of two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^k$ is

$$\mathbf{a}'\mathbf{b} = a_1b_1 + a_2b_2 + \dots + a_kb_k = \sum_{i=1}^k a_ib_i \in \mathbb{R}.$$

Example:

$$\mathbf{a} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} -2 \\ 0 \\ 2 \end{pmatrix}, \quad \mathbf{a}'\mathbf{b} = 1 \cdot (-2) + 2 \cdot 0 + 3 \cdot 2 = 4.$$

The inner product is commutative:

$$\mathbf{a}'\mathbf{b} = \mathbf{b}'\mathbf{a}.$$

Two vectors \mathbf{a} and \mathbf{b} are called **orthogonal** if $\mathbf{a}'\mathbf{b} = 0$. The vectors \mathbf{a} and \mathbf{b} are called **orthonormal** if, in addition to $\mathbf{a}'\mathbf{b}$, we have $\mathbf{a}'\mathbf{a} = 1$ and $\mathbf{b}'\mathbf{b} = 1$.

2.4.2 Outer product

The outer product (also known as dyadic product) of two vectors $\mathbf{x} \in \mathbb{R}^k$ and $\mathbf{y} \in \mathbb{R}^m$ is

$$\mathbf{xy}' = \begin{pmatrix} x_1y_1 & x_1y_2 & \dots & x_1y_m \\ x_2y_1 & x_2y_2 & \dots & x_2y_m \\ \vdots & \vdots & & \vdots \\ x_ky_1 & x_ky_2 & \dots & x_ky_m \end{pmatrix} \in \mathbb{R}^{k \times m}.$$

Example:

$$\mathbf{x} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} -2 \\ 0 \\ 2 \end{pmatrix}, \quad \mathbf{xy}' = \begin{pmatrix} -2 & 0 & 2 \\ -4 & 0 & 4 \end{pmatrix}.$$

2.4.3 Vector multiplication in R

For vector multiplication in R, we use the operator `%*%` (recall that `*` is already reserved for element-wise multiplication). Let's implement some multiplications.

```
y = c(2,7,4,1) #y is treated as a column vector
t(y) %*% y #the inner product of y with itself
```

```
      [,1]
[1,]    70
```

```
y %*% t(y) #the outer product of y with itself
```

```
      [,1] [,2] [,3] [,4]
[1,]     4    14     8     2
[2,]    14    49    28     7
[3,]     8    28    16     4
[4,]     2     7     4     1
```

```
c(1,2) %*% t(c(-2,0,2)) #the example from above
```

```
      [,1] [,2] [,3]
[1,]    -2     0     2
[2,]    -4     0     4
```

2.5 Matrix-matrix multiplication

The **matrix product** of a $k \times m$ matrix \mathbf{A} and a $m \times n$ matrix \mathbf{B} is the $k \times n$ matrix $\mathbf{C} = \mathbf{AB}$ with the components

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{im}b_{mj} = \sum_{l=1}^m a_{il}b_{lj} = \mathbf{a}'_i \mathbf{b}_j,$$

where $\mathbf{a}_i = (a_{i1}, \dots, a_{im})'$ is the i -th row of \mathbf{A} written as a column vector, and $\mathbf{b}_j = (b_{1j}, \dots, b_{mj})'$ is the j -th column of \mathbf{B} . The full matrix product can be written as

$$\mathbf{AB} = \begin{pmatrix} \mathbf{a}'_1 \\ \vdots \\ \mathbf{a}'_k \end{pmatrix} (\mathbf{b}_1 \quad \dots \quad \mathbf{b}_n) = \begin{pmatrix} \mathbf{a}'_1 \mathbf{b}_1 & \dots & \mathbf{a}'_1 \mathbf{b}_n \\ \vdots & & \vdots \\ \mathbf{a}'_k \mathbf{b}_1 & \dots & \mathbf{a}'_k \mathbf{b}_n \end{pmatrix}.$$

The matrix product is only defined if the number of columns of the first matrix equals the number of rows of the second matrix. Therefore, we say that the $k \times m$ matrix \mathbf{A} and the $m \times n$ matrix \mathbf{B} are **conformable for matrix multiplication**.

Example: Let

$$\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 2 & 1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -1 & 2 \\ -3 & 0 \end{pmatrix}.$$

Their matrix product is

$$\begin{aligned} \mathbf{AB} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} -1 & 2 \\ -3 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 \cdot (-1) + 0 \cdot (-3) & 1 \cdot 2 + 0 \cdot 0 \\ 0 \cdot (-1) + 1 \cdot (-3) & 0 \cdot 2 + 1 \cdot 0 \\ 2 \cdot (-1) + 1 \cdot (-3) & 2 \cdot 2 + 1 \cdot 0 \end{pmatrix} = \begin{pmatrix} -1 & 2 \\ -3 & 0 \\ -5 & 4 \end{pmatrix}. \end{aligned}$$

The `%*%` operator is used in R for matrix-matrix multiplications:

```
A = matrix(c(1,0,2,0,1,1), ncol=2)
B = matrix(c(-1,-3,2,0), ncol=2)
A %*% B
```

```
      [,1] [,2]
[1,]   -1    2
[2,]   -3    0
[3,]   -5    4
```

Matrix multiplication is **not commutative**. In general, we have $\mathbf{AB} \neq \mathbf{BA}$. *Example:*

$$\begin{aligned} \mathbf{AB} &= \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 3 & 5 \\ 7 & 11 \end{pmatrix}, \\ \mathbf{BA} &= \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} = \begin{pmatrix} 4 & 6 \\ 7 & 10 \end{pmatrix}. \end{aligned}$$

Even if neither of the two matrices contains zeros, the matrix product can give the zero matrix:

$$\mathbf{AB} = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} 2 & -4 \\ -1 & 2 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = \mathbf{0}.$$

The following rules of calculation apply (provided the matrices are conformable):

- (i) $\mathbf{A}(\mathbf{BC}) = (\mathbf{AB})\mathbf{C}$ (associativity)
- (ii) $\mathbf{A}(\mathbf{B} + \mathbf{D}) = \mathbf{AB} + \mathbf{AD}$ (distributivity)
- (iii) $(\mathbf{B} + \mathbf{D})\mathbf{C} = \mathbf{BC} + \mathbf{DC}$ (distributivity)
- (iv) $\mathbf{A}(\lambda\mathbf{B}) = \lambda(\mathbf{AB})$ (scalar commutativity)
- (v) $\mathbf{AI}_n = \mathbf{A}$, (identity element)
- (vi) $\mathbf{I}_m\mathbf{A} = \mathbf{A}$ (identity element)
- (vii) $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$ (product transposition)
- (viii) $(\mathbf{ABC})' = \mathbf{C}'\mathbf{B}'\mathbf{A}'$ (product transposition)

3 Rank and inverse

3.1 Linear combination

Let $\mathbf{x}_1, \dots, \mathbf{x}_n$ be vectors of the same order, and let $\lambda_1, \dots, \lambda_n$ be scalars. The vector

$$\lambda_1 \mathbf{x}_1 + \lambda_2 \mathbf{x}_2 + \dots + \lambda_n \mathbf{x}_n$$

is called **linear combination** of $\mathbf{x}_1, \dots, \mathbf{x}_n$. A linear combination can also be written as a matrix-vector product. Let $\mathbf{X} = (\mathbf{x}_1 \ \dots \ \mathbf{x}_n)$ be the matrix with columns $\mathbf{x}_1, \dots, \mathbf{x}_n$, and let $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_n)'$. Then,

$$\lambda_1 \mathbf{x}_1 + \lambda_2 \mathbf{x}_2 + \dots + \lambda_n \mathbf{x}_n = \mathbf{X}\boldsymbol{\lambda}.$$

The vectors $\mathbf{x}_1, \dots, \mathbf{x}_n$ are called **linearly dependent** if at least one can be written as a linear combination of the others. That is, there exists a nonzero vector $\boldsymbol{\lambda}$ with

$$\mathbf{X}\boldsymbol{\lambda} = \lambda_1 \mathbf{x}_1 + \dots + \lambda_n \mathbf{x}_n = \mathbf{0}.$$

The vectors $\mathbf{x}_1, \dots, \mathbf{x}_n$ are called **linearly independent** if

$$\mathbf{X}\boldsymbol{\lambda} = \lambda_1 \mathbf{x}_1 + \dots + \lambda_n \mathbf{x}_n \neq \mathbf{0}$$

for all nonzero vectors $\boldsymbol{\lambda}$.

To check whether the vectors are linearly independent, we can solve the system of equations $\mathbf{X}\boldsymbol{\lambda} = \mathbf{0}$ by Gaussian elimination. If $\boldsymbol{\lambda} = \mathbf{0}$ is the only solution, then the columns of \mathbf{X} are linearly independent. If there is a solution $\boldsymbol{\lambda}$ with $\boldsymbol{\lambda} \neq \mathbf{0}$, then the columns of \mathbf{X} are linearly dependent.

3.2 Column rank

The **rank** of a $k \times m$ matrix $\mathbf{A} = (\mathbf{a}_1 \ \dots \ \mathbf{a}_m)$, written as $\text{rank}(\mathbf{A})$, is the number of linearly independent columns \mathbf{a}_i . We say that \mathbf{A} has **full column rank** if $\text{rank}(\mathbf{X}) = m$.

The identity matrix \mathbf{I}_k has full column rank (i.e., $\text{rank}(\mathbf{I}_n) = k$). As another example, consider

$$\mathbf{X} = \begin{pmatrix} 2 & 1 & 4 \\ 0 & 1 & 2 \end{pmatrix},$$

which has linearly dependent columns since the third column is a linear combination of the first two columns:

$$\begin{pmatrix} 4 \\ 2 \end{pmatrix} = 1 \begin{pmatrix} 2 \\ 0 \end{pmatrix} + 2 \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

The first two columns are linearly independent since $\lambda_1 = 0$ and $\lambda_2 = 0$ are the only solutions to the equation

$$\lambda_1 \begin{pmatrix} 2 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Therefore, we have $\text{rank}(\mathbf{X}) = 2$, i.e., \mathbf{X} does not have a full column rank.

Some useful properties are

- i) $\text{rank}(\mathbf{A}) \leq \min(k, m)$
- ii) $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}')$
- iii) $\text{rank}(\mathbf{AB}) = \min(\text{rank}(\mathbf{A}), \text{rank}(\mathbf{B}))$
- iv) $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}'\mathbf{A}) = \text{rank}(\mathbf{AA}')$.

We can use the `qr()` function to extract the rank in R. Let's compute the rank of the matrices

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 9 & 1 \\ 0 & 11 & 5 \end{pmatrix},$$

$\mathbf{B} = \mathbf{I}_3$, and \mathbf{X} from the example above:

```
A = matrix(c(1,3,0,2,9,11,3,1,5), nrow=3)
qr(A)$rank
```

```
[1] 3
```

```
B = matrix(c(1,1,1,1,1,1,1,1,1), nrow=3)
qr(B)$rank
```

```
[1] 1
```

```
X = matrix(c(2,0,1,1,4,2), ncol=3)
qr(X)$rank
```

```
[1] 2
```


3.3 Nonsingular matrix

A square $k \times k$ matrix \mathbf{A} is called **nonsingular** if it has full rank, i.e., $\text{rank}(\mathbf{A}) = k$. Conversely, \mathbf{A} is called **singular** if it does not have full rank, i.e., $\text{rank}(\mathbf{A}) < k$.

3.4 Determinant

Consider a square $k \times k$ matrix \mathbf{A} . The determinant $\det(\mathbf{A})$ is a measure of the volume of the geometric object formed by the columns of \mathbf{A} (a parallelogram for $k = 2$, a parallelepiped for $k = 3$, a hyper-parallelepiped for $k > 3$). For 2×2 matrices, the determinant is easy to calculate:

$$\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \quad \det(\mathbf{A}) = ad - bc.$$

If \mathbf{A} is triangular (upper or lower), the determinant is the product of the diagonal entries, i.e., $\det(\mathbf{A}) = \prod_{i=1}^k a_{ii}$. Hence, Gaussian elimination can be used to compute the determinant by transforming the matrix to a triangular one. The exact definition of the determinant is technical and of little importance to us. A useful relation is the following:

$$\begin{aligned} \det(\mathbf{A}) = 0 & \Leftrightarrow \mathbf{A} \text{ is singular} \\ \det(\mathbf{A}) \neq 0 & \Leftrightarrow \mathbf{A} \text{ is nonsingular.} \end{aligned}$$

In R, we have the `det()` function to compute the determinant:

```
det(A)
```

```
[1] 103
```

```
det(B)
```

```
[1] 0
```

Since $\det(\mathbf{A}) \neq 0$ and $\det(\mathbf{B}) = 0$, we conclude that \mathbf{A} is nonsingular and \mathbf{B} is singular.

3.5 Inverse matrix

The **inverse** \mathbf{A}^{-1} of a square $k \times k$ matrix \mathbf{A} is defined by the property

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}_k.$$

When multiplied from the left or the right, the inverse matrix produces the identity matrix. The inverse exists if and only if \mathbf{A} is nonsingular, i.e., $\det(\mathbf{A}) \neq 0$. Therefore, a nonsingular matrix is also called **invertible matrix**. Note that only square matrices can be inverted.

For 2×2 matrices, there exists a simple formula:

$$\mathbf{A}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix},$$

where $\det(\mathbf{A}) = ad - bc$. We swap the main diagonal elements, reverse the sign of the off-diagonal elements, and divide all entries by the determinant. *Example:*

$$\mathbf{A} = \begin{pmatrix} 5 & 6 \\ 1 & 2 \end{pmatrix}$$

We have $\det(\mathbf{A}) = ad - bc = 5 \cdot 2 - 6 \cdot 1 = 4$, and

$$\mathbf{A}^{-1} = \frac{1}{4} \cdot \begin{pmatrix} 2 & -6 \\ -1 & 5 \end{pmatrix}.$$

Indeed, \mathbf{A}^{-1} is the inverse of \mathbf{A} since

$$\mathbf{A}\mathbf{A}^{-1} = \begin{pmatrix} 5 & 6 \\ 1 & 2 \end{pmatrix} \cdot \frac{1}{4} \cdot \begin{pmatrix} 2 & -6 \\ -1 & 5 \end{pmatrix} = \frac{1}{4} \cdot \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{I}_2.$$

One way to calculate the inverse of higher order square matrices is to solve equation $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$ with Gaussian elimination. R can compute the inverse matrix quickly using the function `solve()`:

```
solve(A) #inverse of A
```

```
      [,1]      [,2]      [,3]  
[1,] 0.3300971 0.22330097 -0.24271845  
[2,] -0.1456311 0.04854369 0.07766990  
[3,] 0.3203883 -0.10679612 0.02912621
```

We have the following relationship between invertibility, rank, and determinant of a square matrix \mathbf{A} :

- \mathbf{A} is nonsingular
- \Leftrightarrow all columns of \mathbf{A} are linearly independent
- \Leftrightarrow \mathbf{A} has full column rank
- \Leftrightarrow the determinant is nonzero ($\det(\mathbf{A}) \neq 0$).

Similarly,

- \mathbf{A} is singular
- \Leftrightarrow \mathbf{A} has linearly dependent columns
- \Leftrightarrow \mathbf{A} does not have full rank
- \Leftrightarrow the determinant is zero ($\det(\mathbf{A}) = 0$).

Below you will find some important properties for nonsingular matrices:

- i) $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$
- ii) $(\mathbf{A}')^{-1} = (\mathbf{A}^{-1})'$
- iii) $(\lambda\mathbf{A})^{-1} = \frac{1}{\lambda}\mathbf{A}^{-1}$ for any $\lambda \neq 0$
- iv) $\det(\mathbf{A}^{-1}) = \frac{1}{\det(\mathbf{A})}$
- v) $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$
- vi) $(\mathbf{ABC})^{-1} = \mathbf{C}^{-1}\mathbf{B}^{-1}\mathbf{A}^{-1}$
- vii) If \mathbf{A} is symmetric, then \mathbf{A}^{-1} is symmetric.

4 Advanced concepts

4.1 Trace

The **trace** of a $k \times k$ square matrix \mathbf{A} is the sum of the diagonal entries:

$$\text{tr}(\mathbf{A}) = \sum_{i=1}^n a_{ii}$$

Example:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 9 & 1 \\ 0 & 11 & 5 \end{pmatrix} \Rightarrow \text{tr}(\mathbf{A}) = 1 + 9 + 5 = 15$$

In R we have

```
A = matrix(c(1,3,0,2,9,11,3,1,5), nrow=3)
sum(diag(A)) #trace = sum of diagonal entries
```

```
[1] 15
```

The following properties hold for square matrices \mathbf{A} and \mathbf{B} and scalars λ :

- i) $\text{tr}(\lambda \mathbf{A}) = \lambda \text{tr}(\mathbf{A})$
- ii) $\text{tr}(\mathbf{A} + \mathbf{B}) = \text{tr}(\mathbf{A}) + \text{tr}(\mathbf{B})$
- iii) $\text{tr}(\mathbf{A}') = \text{tr}(\mathbf{A})$
- iv) $\text{tr}(\mathbf{I}_k) = k$

For $\mathbf{A} \in \mathbb{R}^{k \times m}$ and $\mathbf{B} \in \mathbb{R}^{m \times k}$ we have

$$\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA}).$$

4.2 Idempotent matrix

The square matrix \mathbf{A} is called **idempotent** if $\mathbf{A}\mathbf{A} = \mathbf{A}$. The identity matrix is idempotent: $\mathbf{I}_n\mathbf{I}_n = \mathbf{I}_n$. Another example is the matrix

$$\mathbf{A} = \begin{pmatrix} 4 & -1 \\ 12 & -3 \end{pmatrix}.$$

We have

$$\begin{aligned} \mathbf{A}\mathbf{A} &= \begin{pmatrix} 4 & -1 \\ 12 & -3 \end{pmatrix} \begin{pmatrix} 4 & -1 \\ 12 & -3 \end{pmatrix} \\ &= \begin{pmatrix} 16 - 12 & -4 + 3 \\ 48 - 36 & -12 + 9 \end{pmatrix} \\ &= \begin{pmatrix} 4 & -1 \\ 12 & -3 \end{pmatrix} = \mathbf{A}. \end{aligned}$$

4.3 Eigendecomposition

4.3.1 Eigenvalues

An **eigenvalue** λ of a $k \times k$ square matrix is a solution to the equation

$$\det(\lambda\mathbf{I}_k - \mathbf{A}) = 0.$$

The function $f(\lambda) = \det(\lambda\mathbf{I}_k - \mathbf{A})$ has exactly k roots so that $\det(\lambda\mathbf{I}_k - \mathbf{A}) = 0$ has exactly k solutions. The solutions $\lambda_1, \dots, \lambda_k$ are the k eigenvalues of \mathbf{A} .

Most applications of eigenvalues in econometrics concern symmetric matrices. In this case, all eigenvalues are real-valued. In the case of non-symmetric matrices, some eigenvalues may be complex-valued.

Useful properties of the eigenvalues of a symmetric $k \times k$ matrix are:

- i) $\det(\mathbf{A}) = \lambda_1 \cdot \dots \cdot \lambda_k$
- ii) $\text{tr}(\mathbf{A}) = \lambda_1 + \dots + \lambda_k$
- iii) \mathbf{A} is nonsingular if and only if all eigenvalues are nonzero
- iv) \mathbf{AB} and \mathbf{BA} have the same eigenvalues.

4.3.2 Eigenvectors

If λ_i is an eigenvalue of \mathbf{A} , then $\lambda_i \mathbf{I}_k - \mathbf{A}$ is singular, which implies that there exists a linear combination vector \mathbf{v}_i with $(\lambda_i \mathbf{I}_k - \mathbf{A})\mathbf{v}_i = \mathbf{0}$. Equivalently,

$$\mathbf{A}\mathbf{v}_i = \lambda_i \mathbf{v}_i,$$

which can be solved by Gaussian elimination. It is convenient to normalize any solution such that $\mathbf{v}_i' \mathbf{v}_i = 1$. The solutions $\mathbf{v}_1, \dots, \mathbf{v}_k$ are called eigenvectors of \mathbf{A} to corresponding eigenvalues $\lambda_1, \dots, \lambda_k$.

4.3.3 Spectral decomposition

If \mathbf{A} is symmetric, then $\mathbf{v}_1, \dots, \mathbf{v}_k$ are pairwise orthogonal (i.e., $\mathbf{v}_i' \mathbf{v}_j = 0$ for $i \neq j$). Let $\mathbf{V} = (\mathbf{v}_1 \dots \mathbf{v}_k)$ be the $k \times k$ matrix of eigenvectors and let $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_k)$ be the $k \times k$ diagonal matrix with the eigenvalues on the main diagonal. Then, we can write

$$\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}',$$

which is called the **spectral decomposition** of \mathbf{A} . The matrix of eigenvalues can be written as $\mathbf{\Lambda} = \mathbf{V}'\mathbf{A}\mathbf{V}$.

4.3.4 Eigendecomposition in R

The function `eigen()` computes the eigenvalues and corresponding eigenvectors.

```
B=t(A)%*%A
B #A'A is symmetric
```

```
      [,1] [,2] [,3]
[1,]   10   29   6
[2,]   29  206  70
[3,]    6   70  35
```

```
eigen(B) #eigenvalues and eigenvector matrix
```

```
eigen() decomposition
$values
[1] 234.827160 12.582227 3.590613

$vectors
      [,1]      [,2]      [,3]
[1,] -0.1293953 -0.5312592 0.8372697
[2,] -0.9346164 -0.2167553 -0.2819739
[3,] -0.3312839 0.8190121 0.4684764
```

4.4 Definite matrix

The $k \times k$ square matrix \mathbf{A} is called **positive definite** if

$$\mathbf{c}'\mathbf{A}\mathbf{c} > 0$$

holds for all nonzero vectors $\mathbf{c} \in \mathbb{R}^k$. If

$$\mathbf{c}'\mathbf{A}\mathbf{c} \geq 0$$

for all vectors $\mathbf{c} \in \mathbb{R}^k$, the matrix is called **positive semi-definite**. Analogously, \mathbf{A} is called **negative definite** if $\mathbf{c}'\mathbf{A}\mathbf{c} < 0$ and **negative semi-definite** if $\mathbf{c}'\mathbf{A}\mathbf{c} \leq 0$ for all nonzero vectors $\mathbf{c} \in \mathbb{R}^k$. A matrix that is neither positive semi-definite nor negative semi-definite is called **indefinite**.

The definiteness property of a symmetric matrix \mathbf{A} can be determined using its eigenvalues:

- i) \mathbf{A} is positive definite \Leftrightarrow all eigenvalues of \mathbf{A} are strictly positive
- ii) \mathbf{A} is negative definite \Leftrightarrow all eigenvalues of \mathbf{A} are strictly negative
- iii) \mathbf{A} is positive semi-definite \Leftrightarrow all eigenvalues of \mathbf{A} are non-negative
- iv) \mathbf{A} is negative semi-definite \Leftrightarrow all eigenvalues of \mathbf{A} are non-positive

```
eigen(B)$values #B is positive definite (all eigenvalues positive)
```

```
[1] 234.827160 12.582227 3.590613
```

The matrix analog of a positive or negative number (scalar) is a positive definite or negative definite matrix. Therefore, we use the notation

- i) $\mathbf{A} > 0$ if \mathbf{A} is positive definite
- ii) $\mathbf{A} < 0$ if \mathbf{A} is negative definite
- iii) $\mathbf{A} \geq 0$ if \mathbf{A} is positive semi-definite
- iv) $\mathbf{A} \leq 0$ if \mathbf{A} is negative semi-definite

The notation $\mathbf{A} > \mathbf{B}$ means that the matrix $\mathbf{A} - \mathbf{B}$ is positive definite.

4.5 Cholesky decomposition

Any positive definite and symmetric matrix \mathbf{B} can be written as

$$\mathbf{B} = \mathbf{P}\mathbf{P}',$$

where \mathbf{P} is a lower triangular matrix with strictly positive diagonal entries $p_{jj} > 0$. This representation is called **Cholesky decomposition**. The matrix \mathbf{P} is unique. For a 2×2 matrix \mathbf{B} we have

$$\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} p_{11} & 0 \\ p_{21} & p_{22} \end{pmatrix} \begin{pmatrix} p_{11} & p_{21} \\ 0 & p_{22} \end{pmatrix} \\ = \begin{pmatrix} p_{11}^2 & p_{11}p_{21} \\ p_{11}p_{21} & p_{21}^2 + p_{22}^2 \end{pmatrix},$$

which implies $p_{11} = \sqrt{b_{11}}$, $p_{21} = b_{21}/p_{11}$, and $p_{22} = \sqrt{b_{22} - p_{21}^2}$. For a 3×3 matrix we obtain

$$\begin{pmatrix} b_{11} & b_{12} & b_{31} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} p_{11} & 0 & 0 \\ p_{21} & p_{22} & 0 \\ p_{31} & p_{32} & p_{33} \end{pmatrix} \begin{pmatrix} p_{11} & p_{21} & p_{31} \\ 0 & p_{22} & p_{32} \\ 0 & 0 & p_{33} \end{pmatrix} \\ = \begin{pmatrix} p_{11}^2 & p_{11}p_{21} & p_{11}p_{31} \\ p_{11}p_{21} & p_{21}^2 + p_{22}^2 & p_{21}p_{31} + p_{22}p_{32} \\ p_{11}p_{31} & p_{21}p_{31} + p_{22}p_{32} & p_{31}^2 + p_{32}^2 + p_{33}^2 \end{pmatrix},$$

which implies

$$p_{11} = \sqrt{b_{11}}, \quad p_{21} = \frac{b_{21}}{p_{11}}, \quad p_{31} = \frac{b_{31}}{p_{11}}, \quad p_{22} = \sqrt{b_{22} - p_{21}^2}, \\ p_{32} = \frac{b_{32} - p_{21}p_{31}}{p_{22}}, \quad p_{33} = \sqrt{b_{33} - p_{31}^2 - p_{32}^2}.$$

Let's compute the Cholesky decomposition of

$$\mathbf{B} = \begin{pmatrix} 1 & -0.5 & 0.6 \\ -0.5 & 1 & 0.25 \\ 0.6 & 0.25 & 1 \end{pmatrix}$$

using the R function `chol()`:


```
B = matrix(c(1, -0.5, 0.6, -0.5, 1, 0.25, 0.6, 0.25, 1), ncol=3)
chol(B)
```

```
      [,1]      [,2]      [,3]
[1,]      1 -0.5000000 0.6000000
[2,]      0  0.8660254 0.6350853
[3,]      0  0.0000000 0.4864840
```

4.6 Vectorization

The **vectorization operator** `vec()` stacks the matrix entries column-wise into a large vector. The vectorized $k \times m$ matrix \mathbf{A} is the $km \times 1$ vector

$$\text{vec}(\mathbf{A}) = (a_{11}, \dots, a_{k1}, a_{12}, \dots, a_{k2}, \dots, a_{1m}, \dots, a_{km})'.$$

```
c(A) #vectorize the matrix A
```

```
[1]  1  3  0  2  9 11  3  1  5
```

4.7 Kronecker product

The **Kronecker product** \otimes multiplies each element of the left-hand side matrix with the entire matrix on the right-hand side. For a $k \times m$ matrix \mathbf{A} and a $r \times s$ matrix \mathbf{B} , we get the $kr \times ms$ matrix

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & \dots & a_{1m}\mathbf{B} \\ \vdots & & \vdots \\ a_{k1}\mathbf{B} & \dots & a_{km}\mathbf{B} \end{pmatrix},$$

where each entry $a_{ij}\mathbf{B}$ is a $r \times s$ matrix.

```
A %x% B #Kronecker product in R
```

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
[1,]  1.0 -0.50 0.60  2.0 -1.00  1.20  3.0 -1.50 1.80
[2,] -0.5  1.00 0.25 -1.0  2.00  0.50 -1.5  3.00 0.75
[3,]  0.6  0.25 1.00  1.2  0.50  2.00  1.8  0.75 3.00
[4,]  3.0 -1.50 1.80  9.0 -4.50  5.40  1.0 -0.50 0.60
[5,] -1.5  3.00 0.75 -4.5  9.00  2.25 -0.5  1.00 0.25
[6,]  1.8  0.75 3.00  5.4  2.25  9.00  0.6  0.25 1.00
```

[7,]	0.0	0.00	0.00	11.0	-5.50	6.60	5.0	-2.50	3.00
[8,]	0.0	0.00	0.00	-5.5	11.00	2.75	-2.5	5.00	1.25
[9,]	0.0	0.00	0.00	6.6	2.75	11.00	3.0	1.25	5.00

4.8 Vector and matrix norm

A norm $\|\cdot\|$ of a vector or a matrix is a measure of distance from the origin. The most commonly used norms are the Euclidean vector norm

$$\|\mathbf{a}\| = \sqrt{\mathbf{a}'\mathbf{a}} = \sqrt{\sum_{i=1}^k a_i^2}$$

for $\mathbf{a} \in \mathbb{R}^k$, and the Frobenius matrix norm

$$\|\mathbf{A}\| = \sqrt{\sum_{i=1}^k \sum_{j=1}^m a_{ij}^2}$$

for $\mathbf{A} \in \mathbb{R}^{k \times m}$.

A norm satisfies the following properties:

- i) $\|\lambda\mathbf{A}\| = |\lambda|\|\mathbf{A}\|$ for any scalar λ (absolute homogeneity)
- ii) $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$ (triangle inequality)
- iii) $\|\mathbf{A}\| = 0$ implies $\mathbf{A} = \mathbf{0}$ (definiteness)

5 Matrix calculus

Let $f(\beta_1, \dots, \beta_k) = f(\boldsymbol{\beta})$ be a twice-differential real-valued function that depends on some vector $\boldsymbol{\beta} = (\beta_1, \dots, \beta_k)'$. Examples that frequently appear in econometrics are functions of the inner product form $f(\boldsymbol{\beta}) = \mathbf{a}'\boldsymbol{\beta}$, where $\mathbf{a} \in \mathbb{R}^k$, and functions of the sandwich form $f(\boldsymbol{\beta}) = \boldsymbol{\beta}'\mathbf{A}\boldsymbol{\beta}$, where $\mathbf{A} \in \mathbb{R}^{k \times k}$.

5.1 Gradient

The **first derivatives vector** or **gradient** is

$$\frac{\partial f(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \begin{pmatrix} \frac{\partial f(\boldsymbol{\beta})}{\partial \beta_1} \\ \vdots \\ \frac{\partial f(\boldsymbol{\beta})}{\partial \beta_k} \end{pmatrix}$$

If the gradient is evaluated at some particular value $\boldsymbol{\beta} = \mathbf{b}$, we write

$$\frac{\partial f}{\partial \boldsymbol{\beta}}(\mathbf{b})$$

Useful properties for inner product and sandwich forms are

$$\begin{aligned} (i) \quad & \frac{\partial(\mathbf{a}'\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \mathbf{a} \\ (ii) \quad & \frac{\partial(\boldsymbol{\beta}'\mathbf{A}\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = (\mathbf{A} + \mathbf{A}')\boldsymbol{\beta}. \end{aligned}$$

5.2 Hessian

The **second derivatives matrix** or **Hessian** is the $k \times k$ matrix

$$\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'} = \begin{pmatrix} \frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_1 \partial \beta_1} & \cdots & \frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_k \partial \beta_1} \\ \vdots & & \vdots \\ \frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_1 \partial \beta_k} & \cdots & \frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_k \partial \beta_k} \end{pmatrix}.$$

If the Hessian is evaluated at some particular value $\beta = \mathbf{b}$, we write

$$\frac{\partial^2 f}{\partial \beta \partial \beta'}(\mathbf{b})$$

The Hessian is symmetric. Each column of the Hessian is the derivative of the components of the gradient for the corresponding variable in β' :

$$\begin{aligned} \frac{\partial^2 f(\beta)}{\partial \beta \partial \beta'} &= \frac{\partial(\partial f(\beta)/\partial \beta)}{\partial \beta'} \\ &= \begin{bmatrix} \frac{\partial(\partial f(\beta)/\partial \beta)}{\partial \beta_1} & \frac{\partial(\partial f(\beta)/\partial \beta)}{\partial \beta_2} & \cdots & \frac{\partial(\partial f(\beta)/\partial \beta)}{\partial \beta_n} \end{bmatrix} \end{aligned}$$

The Hessian of a sandwich form function is

$$\frac{\partial^2(\beta' \mathbf{A} \beta)}{\partial \beta \partial \beta'} = \mathbf{A} + \mathbf{A}'.$$

5.3 Optimization

Recall the *first-order* (necessary) and *second-order* (sufficient) conditions for optimum (maximum or minimum) in the univariate case:

- **First-order condition:** the first derivative evaluated at the optimum is zero.
- **Second-order condition:** the second derivative at the optimum is negative for a maximum and positive for a minimum.

Similarly, we formulate first and second-order conditions for a function $f(\beta)$. The **first-order condition** for an optimum (maximum or minimum) at \mathbf{b} is

$$\frac{\partial f}{\partial \beta}(\mathbf{b}) = \mathbf{0}.$$

The **second-order condition** is

$$\begin{aligned} \frac{\partial^2 f}{\partial \beta \partial \beta'}(\mathbf{b}) &> 0 \quad \text{for a minimum at } \mathbf{b}, \\ \frac{\partial^2 f}{\partial \beta \partial \beta'}(\mathbf{b}) &< 0 \quad \text{for a maximum at } \mathbf{b}. \end{aligned}$$

Recall that, in the context of matrices, the notation “ > 0 ” means positive definite, and “ < 0 ” means negative definite.

6 Problems

Problem 1

Consider the matrix

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}.$$

- a) Determine \mathbf{A}' . Is \mathbf{A} symmetric?
- b) Is \mathbf{A} idempotent?
- c) Compute the determinant and the rank. Is \mathbf{A} nonsingular?
- d) Compute the inverse.
- e) Compute the trace.

Problem 2

- a) Let $\mathbf{AB} = \mathbf{C}$, where

$$\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 2 & 2 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix}.$$

Compute \mathbf{B} .

- b) $\boldsymbol{\delta}$ and $\boldsymbol{\gamma}$ are $c \times 1$ vectors, \mathbf{X} is a $d \times c$ matrix, and \mathbf{Y} is a $c \times d$ matrix. Determine the orders of \mathbf{XY} , \mathbf{YX} , $\boldsymbol{\gamma}'\boldsymbol{\gamma}$, $\boldsymbol{\gamma}\boldsymbol{\gamma}'$, and $\boldsymbol{\delta}'\mathbf{YX}\boldsymbol{\gamma}$. Under which conditions do the expressions \mathbf{Y}^{-1} and $\boldsymbol{\delta}'\mathbf{YX} + \boldsymbol{\gamma}'\boldsymbol{\gamma}$ exist?
- c) Compute $\text{tr}(\lambda \mathbf{R}'\mathbf{R})$ for $\lambda \in \mathbb{R}$ and

$$\mathbf{R} = \begin{pmatrix} \frac{1}{4} & \frac{\sqrt{3}}{4} \\ \frac{\sqrt{3}}{4} & \frac{3}{4} \end{pmatrix}.$$

Problem 3

Let \mathbf{A} be nonsingular. Simplify the expression

$$\left(\frac{1}{\sqrt{2}} \mathbf{A}^{-1} \left(\frac{1}{\sqrt{2}} \mathbf{A}'' + \frac{\sqrt{2}}{2} \mathbf{A} \right) \right).$$

Problem 4

Consider the $n \times k$ matrix \mathbf{X} with $\text{rank}(\mathbf{X}) = k$. Moreover, let $\mathbf{P} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$, and let $\mathbf{M} = \mathbf{I}_n - \mathbf{P}$

- a) Determine the order of the following matrices: \mathbf{I}_n , $\mathbf{X}'\mathbf{X}$, \mathbf{P} , \mathbf{M}
- b) Which matrices from a) are symmetric?
- c) Which matrices from a) are idempotent?
- d) Compute the trace of \mathbf{I}_n and \mathbf{P} .

Problem 5

Let \mathbf{X} be a $n \times k$ matrix. Show that $\mathbf{X}'\mathbf{X}$ is positive semi-definite. Under which condition is $\mathbf{X}'\mathbf{X}$ positive definite?

Problem 6

Let $\mathbf{y} \in \mathbb{R}^n$, \mathbf{X} be a $n \times k$ matrix, and $\boldsymbol{\beta} \in \mathbb{R}^k$. Compute the derivatives

$$\frac{\partial f(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}, \quad \frac{\partial^2 f(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'},$$

for the function $f(\boldsymbol{\beta}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$.

6.1 Solutions

Solutions to the problems are available [here](#) (unfortunately only in German so far)