Linear systems of equations
A brief encounter with linear algebra

MaLe
Norman Juchler
Life Sciences and Facility Management
Institute of Computational Life Sciences

In simple terms, a matrix is a rectangular array of numbers, arranged in rows and columns. Each number in a matrix is called an element. For example, the matrix

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

has two rows and three columns. The element in the first row, second column is 2 (denoted as a_{01}), and the element in the second row, third column is 6 (denoted as a_{12}):

$$A = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{pmatrix}$$

We can use matrices to express linear systems of equations in condensed form. Consider the following example:

$$\begin{vmatrix} y_0 = a_{00}x_0 + a_{01}x_1 + a_{02}x_2 \\ y_1 = a_{21}x_0 + a_{22}x_1 + a_{23}x_2 \end{vmatrix} \tag{1}$$

This exemplary system can be represented in matrix form as $y = A \cdot x$:

$$\begin{pmatrix} y_0 \\ y_1 \end{pmatrix} = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix} \tag{2}$$

In this notation, it holds that

- $A \in \mathbb{R}^{3\times 3}$ is the coefficient matrix, containing the coefficients a_{ij} of the variables
- $x \in \mathbb{R}^3$ is the column vector of variables $(x_0, x_1, x_2)^T$
- $y \in \mathbb{R}^2$ is the column vector of constants $(y_0, y_1)^T$

Note: To refer to row i of A, we write $a_{i,:}$ or $A(i,:)$ – and $a_{:,j}$ or $A(:,j)$ for column j. The use of ":" reminds us of the slice operator that we would use in Python for arrays.

Besides collecting the numbers in matrices and vectors, the notation also implies a calculation scheme (dt.: *Rechenschema*): The multiplication of a matrix A with a column vector x is defined as:

- (Step 1): Row-wise multiplication of the elements of A with the elements of x
- (Step 2): Subsequent summation of the resulting products

We can express this also very compactly using the sum notation:

$$y_i = A(i,:) \cdot x = \sum_{j=0}^{N} a_{ij} \cdot x_j \ , \quad i \in \{0, 1\}$$

If we apply this scheme, we indeed recover the original system of equations (1):

$$
\begin{aligned}
y_0 &= A_{0,:} \cdot x = a_{00}x_0 + a_{01}x_1 + a_{02}x_2 \\
y_1 &= A_{1,:} \cdot x = a_{10}x_0 + a_{11}x_1 + a_{12}x_2
\end{aligned}
$$

Notes:

- In the above formulas, we employ zero-based indices, meaning that we start counting the indices at zero: $x = (x_0, x_1, x_3)^T$. This choice corresponds to the convention used in Python and other programming languages, where vectors and matrices also use zero-based indices. Conversely, in mathematics and certain other coding languages such as R, it is common to use indices that start with the value 1: $x = (x_1, x_2, x_3)^T$. This is always a source of confusion! Here we use the computer scientist notation.
- With the operator ^{T}, we can *transpose* (flip) of a vector or matrix. For example:

$$
\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^T = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}
$$

$$
\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{pmatrix}^T = \begin{pmatrix} a_{00} & a_{10} \\ a_{01} & a_{11} \\ a_{02} & a_{12} \end{pmatrix}
$$

- We used matrix notation in equation (2) to multiply a matrix with a column vector. We can generalize this scheme to *multiply two matrices* A and B.

$$
A \cdot B = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{pmatrix} \cdot \begin{pmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \\ b_{20} & b_{21} \end{pmatrix} = \begin{pmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{pmatrix} = C
$$

$$
C(i, j) = A(i, :) \cdot B(:, j)
$$

  As an example (see colored elements), by using the above scheme of multiplying and summing row $A(0, :)$ with column $B(:, 1)$, we compute matrix element $C(0, 1) = c_{01}$. Note that the matrices must have matching sizes for this to work!

There is a lot more to say about matrices. The mathematical discipline that studies the properties of matrices and related topics is called *linear algebra*.

Further reading

- A crash course on linear algebra for machine learners and Pythonists. Link
- Essential concepts of linear algebra, by Rob Taylor. Part 1, Part 2
- Videos on linear algebra by 3blue1brown, wonderfully produced. Link