

Vector manipulation in Python

In this lab, you will have the opportunity to practice once again with the NumPy library. This time, we will explore some advanced operations with arrays and matrices.

At the end of the previous module, we used PCA to transform a set of many variables into a set of only two uncorrelated variables. This process was made through a transformation of the data called rotation.

In this week's assignment, you will need to find a transformation matrix from English to French vector space embeddings. Such a transformation matrix is nothing else but a matrix that rotates and scales vector spaces.

In this notebook, we will explain in detail the rotation transformation.

Transforming vectors

There are three main vector transformations:

- Scaling
- Translation
- Rotation

In previous notebooks, we have applied the first two kinds of transformations. Now, let us learn how to use a fundamental transformation on vectors called *rotation*.

The rotation operation changes the direction of a vector, letting unaffected its dimensionality and its norm. Let us explain with some examples.

In the following cells, we will define a NumPy matrix and a NumPy array. Soon we will explain how this is related to matrix rotation.

```
In [1]: import numpy as np                # Import numpy for array manipulation
import matplotlib.pyplot as plt          # Import matplotlib for charts
from utils_nb import plot_vectors        # Function to plot vectors (arrows)
```

Example 1

```
In [2]: # Create a 2 x 2 matrix
R = np.array([[2, 0],
               [0, -2]])
```