

Actividades Tema 1

1. Implementa en Python las funciones siguientes:

- a) Dados dos vectores $u, v \in \mathbb{R}^n$, la proyección u sobre v y la ortogonal de u sobre v . [1 punto]
 - Genera algunos pares de vectores aleatorios $u, v \in \mathbb{R}^n$, con valores de n a tu elección y con la función `numpy.random.rand`, y calcula la correspondiente proyección de u sobre v y ortogonal de u sobre v . ¿Qué puede decirse del producto escalar entre $\text{Proj}_v(u)$ y $\text{Ort}_v(u)$? ¿Qué puede decirse, por tanto, de la posición relativa entre sendos vectores? [0.75 puntos]
 - Demuestra que $\text{Proj}_v(u) \cdot \text{Ort}_v(u)$ tiene el valor que has conjeturado en el apartado anterior $\forall u, v \in \mathbb{R}^n$. [0.75 puntos]
- b) Dadas dos matrices de entrada A y B , comprobación de que sus dimensiones son compatibles para realizar el producto y, en tal caso, devolución del producto $C = AB$. La implementación del producto de matrices debe realizarse sin utilizar la función `numpy.multiply` ni ninguna otra función preprogramada que calcule directamente el producto entre dos matrices. [1.5 puntos]
- c) Dada una matriz A , comprobación de que sea una matriz cuadrada y, en tal caso, cálculo de su determinante. La implementación del determinante de una matriz debe realizarse sin utilizar la función `numpy.linalg.det` ni ninguna otra función que calcule directamente el determinante de una matriz. [1.5 puntos]

2. Realiza diferentes programas en Python que, dada una matriz $A \in \mathbb{R}^{n \times n}$ y un exponente $k \in \mathbb{N}$, calculen A^k :

- a) Multiplicando secuencialmente la matriz A , mediante la función que has implementado en el apartado b) del ejercicio anterior. [1.5 puntos]
- b) Aprovechando la forma diagonalizada de A (puedes utilizar para ello la función preprogramada de `Scipy numpy.linalg.eig`). [1.5 puntos]
- c) Compara sendas funciones generando una matriz aleatoria $A \in \mathbb{R}^{50 \times 50}$ simétrica (utilizando el script adjunto `pseudorandom_matrix.py`) y analizando el tiempo que se tarda en calcular A^{100} en cada uno de los métodos. [1.5 puntos]