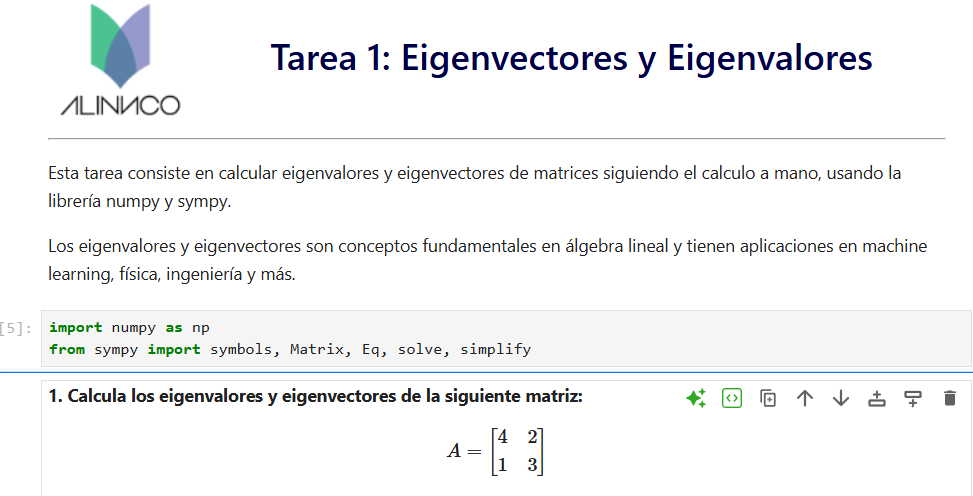
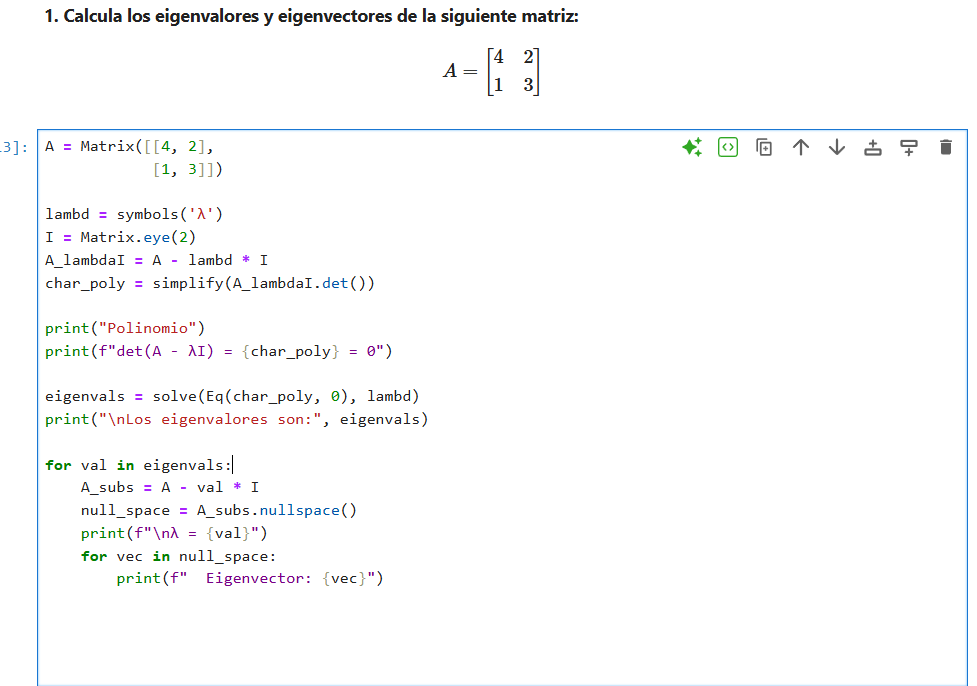
Tarea 1 eigenvalores y engevectores





1.-

A = Matrix([[4, 2],

[1, 3]])

λ = symbols('λ')

I = Matrix.eye(2)

A\_lambdaI = A - lambd \* I

char\_poly = simplify(A\_lambdaI.det())

print("Polinomio:")

print(f"det(A - λI) = {char\_poly} = 0")

eigenvals = solve(Eq(char\_poly, 0), lambd)

print("\nlos autovalores son:", eigenvals)

for val in eigenvals:

A\_subs = A - val \* I

null\_space = A\_subs.nullspace()

print(f"λ = {val}")

for vec in null\_space:

print(f" Eigenvector: {vec}")

1 con numpy

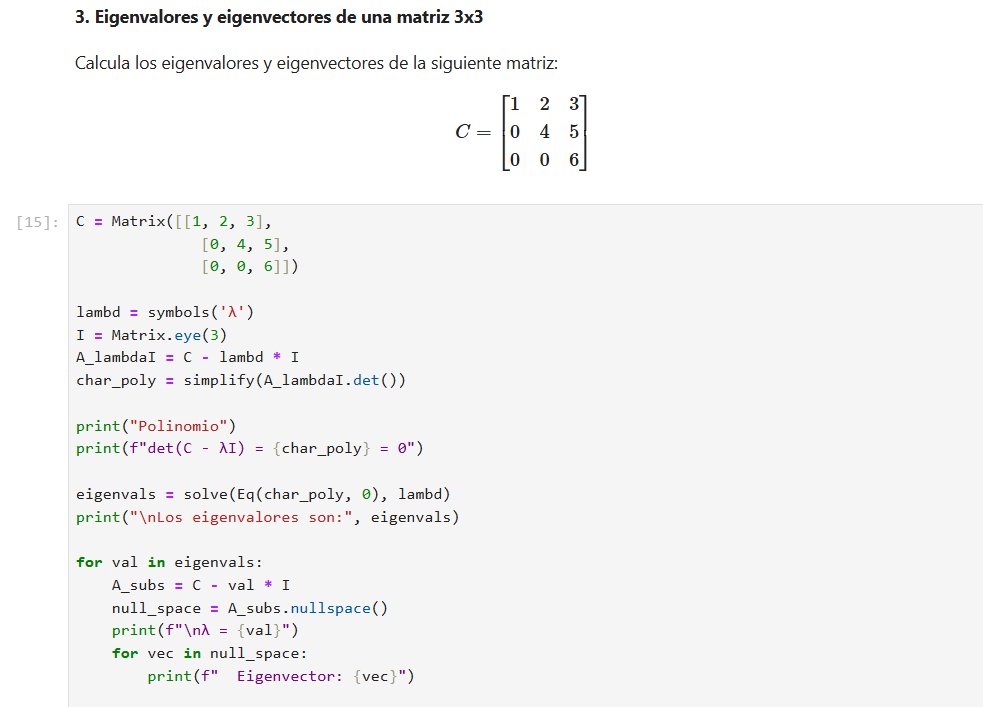
A = np.array([[4, 2],

[1, 3]])

eigenvalues, eigenvectors = np.linalg.eig(A)

print("Eigenvalores:", eigenvalues)

print("Eigenvectores:\n", eigenvectors)



C = Matrix([[1, 2, 3],

[0, 4, 5],

[0, 0, 6]])

lambd = symbols('λ')

I = Matrix.eye(3)

A\_lambdaI = C - lambd \* I

char\_poly = simplify(A\_lambdaI.det())

print("Polinomio")

print(f"det(C - λI) = {char\_poly} = 0")

eigenvals = solve(Eq(char\_poly, 0), lambd)

print("\nLos eigenvalores son:", eigenvals)

for val in eigenvals:

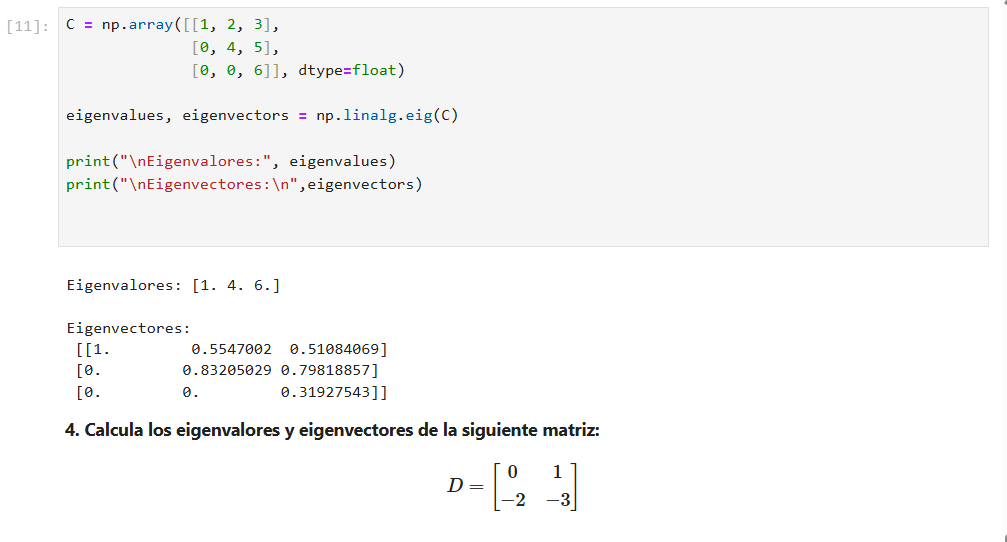
A\_subs = C - val \* I

null\_space = A\_subs.nullspace()

print(f"\nλ = {val}")

for vec in null\_space:

print(f" Eigenvector: {vec}")



C = np.array([[1, 2, 3],

[0, 4, 5],

[0, 0, 6]], dtype=float)

eigenvalues, eigenvectors = np.linalg.eig(C)

print("\nEigenvalores:", eigenvalues)

print("\nEigenvectores:\n",eigenvectors)

4.-

E = np.random.randint(0, 10, size=(4,4))

print("Matriz E:")

print(E)

eigenvalues, eigenvectors = np.linalg.eig(E)

print("Eigenvalores:", eigenvalues)

print("Eigenvectores:\n", eigenvectors)