

NAME:- KETAN DILIP ATTARDE

REG NO:- 24-27-06

MTECH DATA SCIENCE

✓ Finding Dominant Eigenvalue and Eigenvector using Power Method

```
import numpy as np

def power_method(A, num_iterations=1000, tolerance=1e-6):
    n = A.shape[0]

    # Initialize a random vector as an approximation to the dominant eigenvector
    x = np.random.rand(n)
    x /= np.linalg.norm(x)

    for i in range(num_iterations):
        # Perform the power iteration
        x_old = x
        x = np.dot(A, x)

        # Normalize the eigenvector
        x /= np.linalg.norm(x)

        # Check for convergence
        if np.linalg.norm(x - x_old) < tolerance:
            break

    # Calculate the dominant eigenvalue
    eigenvalue = np.dot(x, np.dot(A, x))

    return eigenvalue.round(3), x.round(3)
```

```
# Example usage:  
A = np.array([[1., 2., 0.], [2., 1., 0.], [0., 0., -1.]])  
dominant_eigenvalue, dominant_eigenvector = power_method(A)
```

```
print("Original matrix A:\n", A)  
print("\nDominant eigenvalue:", dominant_eigenvalue)  
print("Dominant eigenvector:", dominant_eigenvector)
```

⇒ Original matrix A:

```
[[ 1.  2.  0.]  
 [ 2.  1.  0.]  
 [ 0.  0. -1.]]
```

Dominant eigenvalue: 3.0

Dominant eigenvector: [0.707 0.707 -0.]