13_Numpy_Algebra

November 24, 2014

Figure 1: BY-SA

Authors : *Sonia Estradé*
José M. Gómez
Ricardo Graciani
Manuel López
Xavier Luri
Josep Sabater

1 Àlgebra lineal

Numpy implementa les operacions d'algebra lineal, és a dir, les operacions amb vectors i matrius. Les eines d'àlgebra lineal es troben en el mòdul numpy.linalg:

```
from numpy.linalg import *
```

1.1 Eines d'àlgebra lineal

1.1.1 Trasposició

Es pot fer amb el mètode `transpose()` dels ndarrays:

```
In [1]: import numpy as np

        # Generem un array 2D
        A = np.linspace(1.,9.,9).reshape(3,3)
        print("Array A")
        print(A)

        # El transposem
        print("Array A transposat")
        print(A.transpose())
```

```
Array A
[[ 1.  2.  3.]
 [ 4.  5.  6.]
 [ 7.  8.  9.]]
```

```
Array A transposat
[[ 1.  4.  7.]
 [ 2.  5.  8.]
 [ 3.  6.  9.]]
```

1.1.2 Determinants i matriu inversa

Els determinants es calculen amb la funció `linalg.det()` i la matriu inversa amb la funció `linalg.inv()`

```
In [2]: import numpy as np

        # Generem un array 2D
        A = np.array( [[1,1,2], [2,1,1], [1,1,1]] )
        print("Array A")
        print(A)

        # Calculem el determinant
        print("Determinant de A: ", np.linalg.det(A))

        # L'invertim
        print("Inversa de A")
        print(np.linalg.inv(A))
```

```
Array A
[[1 1 2]
 [2 1 1]
 [1 1 1]]
Determinant de A:  1.0
Inversa de A
[[ 0.  1. -1.]
 [-1. -1.  3.]
 [ 1. -0. -1.]]
```

1.1.3 Rang d'una matriu

El rang d'una matriu es pot calcular usant la funció `numpy.linalg.matrix_rank(A)`

```
In [3]: import numpy as np

        # Definim la matriu en un array de numpy
        A = np.array( [ [1,2,3,1], [2,1,1,-2], [5,7,10,1], [-1,4,7,7] ])

        print("Matriu: ")
        print(A)
        print("Rang:")
        print(np.linalg.matrix_rank(A))
```

```
Matriu:
[[ 1  2  3  1]
 [ 2  1  1 -2]
 [ 5  7 10  1]
 [-1  4  7  7]]
Rang:
2
```

1.1.4 Traça d'una matriu

La traça d'una matriu es calcula amb la funció `trace()`:

```
In [4]: import numpy as np

        # Generem un array 2D
        A = np.array( [[1,1,2], [2,1,1], [1,1,1]] )
        print("Array A")
        print(A)

        # Calculem la traça
        print("Traça de A: ", np.trace(A))
```

```
Array A
[[1 1 2]
 [2 1 1]
 [1 1 1]]
Traça de A:  3
```

1.1.5 Producte de matrius

El producte de matrius s'implementa amb la funció `dot()`:

```
In [5]: import numpy as np

        # Generem un array 2D
        A = np.array( [[1,1,2], [2,1,1], [1,1,1]] )
        print("Array A")
        print(A)

        # El multipliquem per ell mateix
        print("A*A")
        print(np.dot(A,A))

        # Multipliquem la matriu per un vector
        v = np.array( [3,1,3] )
        print("A*v")
        print(np.dot(A,v))
```

```
Array A
[[1 1 2]
 [2 1 1]
 [1 1 1]]
A*A
[[5 4 5]
 [5 4 6]
 [4 3 4]]
A*v
[10 10  7]
```

1.1.6 Solució de sistemes d'equacions

Un sistema d'equacions

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}$$

es pot representar en forma matricial

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$A\vec{x} = \vec{b}$$

Donada la matriu A del sistema i el vector de termes independents $\vec{b}$ es pot obtenir la solució $\vec{x}$ amb la funció `solve()`:

```
In [6]: import numpy as np

        # Matriu del sistema
        A = np.array( [[1,1,2], [2,1,1], [1,1,1]] )
        print("Matriu del sistema A")
        print(A)

        # Vector de termes independents
        b = np.array( [1,1,1] )
        print("Vector de termes independents")
        print(b)

        # Calculem la solució
        x = np.linalg.solve(A,b)
        print("Solució")
        print(x)
```

```
Matriu del sistema A
[[1 1 2]
 [2 1 1]
 [1 1 1]]
Vector de termes independents
[1 1 1]
Solució
[ 0.  1. -0.]
```

1.1.7 Valors propis i vectors propis

Es poden determinar els valors propis i vectors propis d'una matriu mitjançant la funció `linalg.eig()`. Retorna la llista de valors propis amb la seva multiplicitat i els vectors propis corresponents. Si interessen només els valors propis es pot usar `linalg.eigvals()`

```
In [7]: import numpy as np

        # Generem un array 2D
        A = np.array( [[1,1,0], [2,1,1], [1,1,1]] )
        print("Array A")
        print(A)

        # Calculem els valors propis
        print("Valors propis de A: ", np.linalg.eig(A)[0])
        print("Vectors propis de A: ")
        print(np.linalg.eig(A)[1])

        # Usem eigvals
        print("Valors propis de A (2): ", np.linalg.eigvals(A))
```

```
Array A
[[1 1 0]
 [2 1 1]
 [1 1 1]]
Valors propis de A:  [ 2.87938524  0.65270364 -0.53208889]
Vectors propis de A:
[[ 0.38126346  0.46360414  0.53698978]
 [ 0.71654092 -0.16100803 -0.82271608]
 [ 0.58412951 -0.87129078  0.18649459]]
Valors propis de A (2):  [ 2.87938524  0.65270364 -0.53208889]
```

1.2 Exemples d'exercicis d'àlgebra amb numpy (llistat de problemes de determinants i sistemes d'equacions lineals)

1.2.1 Exercici 7

Donada la matriu de $M_3(\mathbb{R})$

$$\begin{pmatrix} -3 & 2 & -1 \\ 2 & 0 & 1 \\ -1 & 2 & 1 \end{pmatrix}$$

comproveu que és invertible, $det(M) \neq 0$, i calculeu-ne la inversa.

1.2.2 Exercici 10

Trobeu el rang de la matriu

$$\begin{pmatrix} 1 & 2 & 3 & 1 \\ 2 & 1 & 1 & -2 \\ 5 & 7 & 10 & 1 \\ -1 & 4 & 7 & 7 \end{pmatrix}$$

Indicació: useu `numpy.linalg.matrix_rank(A)`

1.2.3 Exercici 13

Donat el sistema

$$\begin{array}{rcrcrcl} x & - & y & + & z & = & 4 \\ 2x & + & y & - & 3z & = & 1 \\ 7x & - & y & - & 3z & = & 14 \end{array}$$

Determineu els rangs de la matriu i la matriu ampliada (numpy) i discutiu les solucions del sistema.