Julia 程式語言學習馬拉松

Day 13

Cupoy 陪跑專家：James Huang

內建線性代數
(LinearAlgebra)
模組簡介

重要知識點

- Julia 內建提供了功能強大的線性代數模組，對於矩陣操作也提供相關的函式以進行線性代數運算，在今天的內容中將簡介內建線性代數模組的功能，包括：
  - 矩陣操作
  - 特殊矩陣
  - 矩陣分解 (Matrix Factorization)
- 矩陣本身在 Julia 即為二維陣列 (Array)。在前面的課程中我們看到 Julia 提供非常強大的陣列操作及運算，建議可以配合陣列的觀念及內容，更可以快速熟悉以下介紹常見的線性代數運算及矩陣操作。
- 另外，在範例中使用 LaTeX 撰寫矩陣範例，有興趣的話可以在 Jupyter Notebook 中看到語法。

矩陣 (Matrix) 操作 – 跡 (Trace)

CUPOY

- 矩陣的跡是指在方陣 (square matrix) 中從左上方至右下方的對角線）上各個元素的總和。

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$tr(A) = a_{11} + a_{22} + a_{33}$$

- Julia 跡的運算可以透過 tr() 函式求得。
- 若矩陣非方陣，tr() 函式則會產生錯誤。

矩陣 (Matrix) 操作 – 行列式 (Determinant)

CUPOY

- 矩陣行列式將一個方陣映射到一個純量的函式。

$$
|A| = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}
$$
$$
= a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{23}a_{32}a_{11} - a_{33}a_{12}a_{21}
$$

- 行列式的運算可以透過 det() 函式求得。
- 若矩陣非方陣，det() 函式則會產生錯誤。

矩陣 (Matrix) 操作 – 反矩陣 (Inverse)

- 若反矩陣存在的話，矩陣 AB = BA = 單位矩陣 I，則稱 B 矩陣是 A 矩陣的反矩陣。

以三階矩陣為例：

$$A = \left[\begin{array}{ccc|ccc} a_{11} & a_{12} & a_{13} & 1 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 1 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & 1 \end{array}\right]$$

透過高斯約當法 (Gaussian Elimination)，可以得到右邊的矩陣即為 A 的反矩陣

$$A^{-1} = \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & b_{11} & b_{12} & b_{13} \\ 0 & 1 & 0 & b_{21} & b_{22} & b_{23} \\ 0 & 0 & 1 & b_{31} & b_{32} & b_{33} \end{array}\right]$$

- 反矩陣可以透過 inv() 函式求得。
- 若矩陣非方陣，inv() 函式則會產生錯誤。

矩陣 (Matrix) 操作 – 轉置 (Transpose)

- 把矩陣 A 的直行寫為橫列，把橫列寫為的直行，即為轉置的矩陣 A^T。

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$A^T = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{bmatrix}$$

- 轉置矩陣可透過下列 2 種方式：
    - transpose() 函式，或是
    - 矩陣加單引號，例如 A’

矩陣 (Matrix) 操作 – 特徵向量與特徵值

CUPOY

- Wikipedia 對於特徵向量與特徵值的定義：
  - “對於一個給定的方陣 A，它的特徵向量 (eigenvector) v 經過這個線性變換之後，得到的新向量仍然與原來的 v 保持在同一條直線上，但其長度或方向也許會改變。即 $Av = \lambda v$。
  - λ 為純量，即特徵向量的長度在該線性變換下縮放的比例，稱為其特徵值 (eigenvalue)。”
- 每個方陣必有其特徵值及特徵向量，但特徵值未必是實數，特徵向量為非零向量。

矩陣 (Matrix) 操作 – 特徵向量與特徵值

- Julia 提供函式以求得特徵向量及特徵值：
  - 特徵向量：eigenvecs()
  - 特徵值：eigenvals()
  - eigen() 函式可以求得特徵向量與特徵值。

In [19]:
```
eigen(A)
```

Out[19]:
```
Eigen{Float64,Float64,Array{Float64,2},Array{Float64,1}}
eigenvalues:
3-element Array{Float64,1}:
 -5.000000000000005
  3.0
  6.0
eigenvectors:
3×3 Array{Float64,2}:
  0.816497   0.534522  0.0584206
  0.408248  -0.801784  0.350524
 -0.408248  -0.267261  0.93473
```

矩陣 (Matrix) 特殊矩陣

CUPOY

特殊矩陣名稱	Julia 函式名稱
對稱矩陣	Symmetric
對角矩陣、雙對角矩陣、三對角矩陣	Diagonal, Bidiagonal, Tridiagonal
對稱三對角矩陣	SymTridiagonal
上三角矩陣	UpperTriangular
下三角矩陣	LowerTriangular
均勻縮放矩陣	UniformScaling
單位下三角矩陣	UnitLowerTriangular
Hermitian 矩陣	Hermitian

矩陣 (Matrix) 分解 (Factorization)

CUPOY

- 矩陣分解是將矩陣拆解為數個矩陣的乘積。
- 常見的矩陣分解及其對應的 Julia 函式

名稱	Julia 函式
Cholesky 分解	cholesky()
QR 分解	qr()
LU 分解	lu()
SVD 分解	svd()

矩陣 (Matrix) 分解 (Factorization)

CUPOY

- Julia 還提供了 factorize() 函式，能夠根據矩陣的類型，自動選置有效的分解方法。

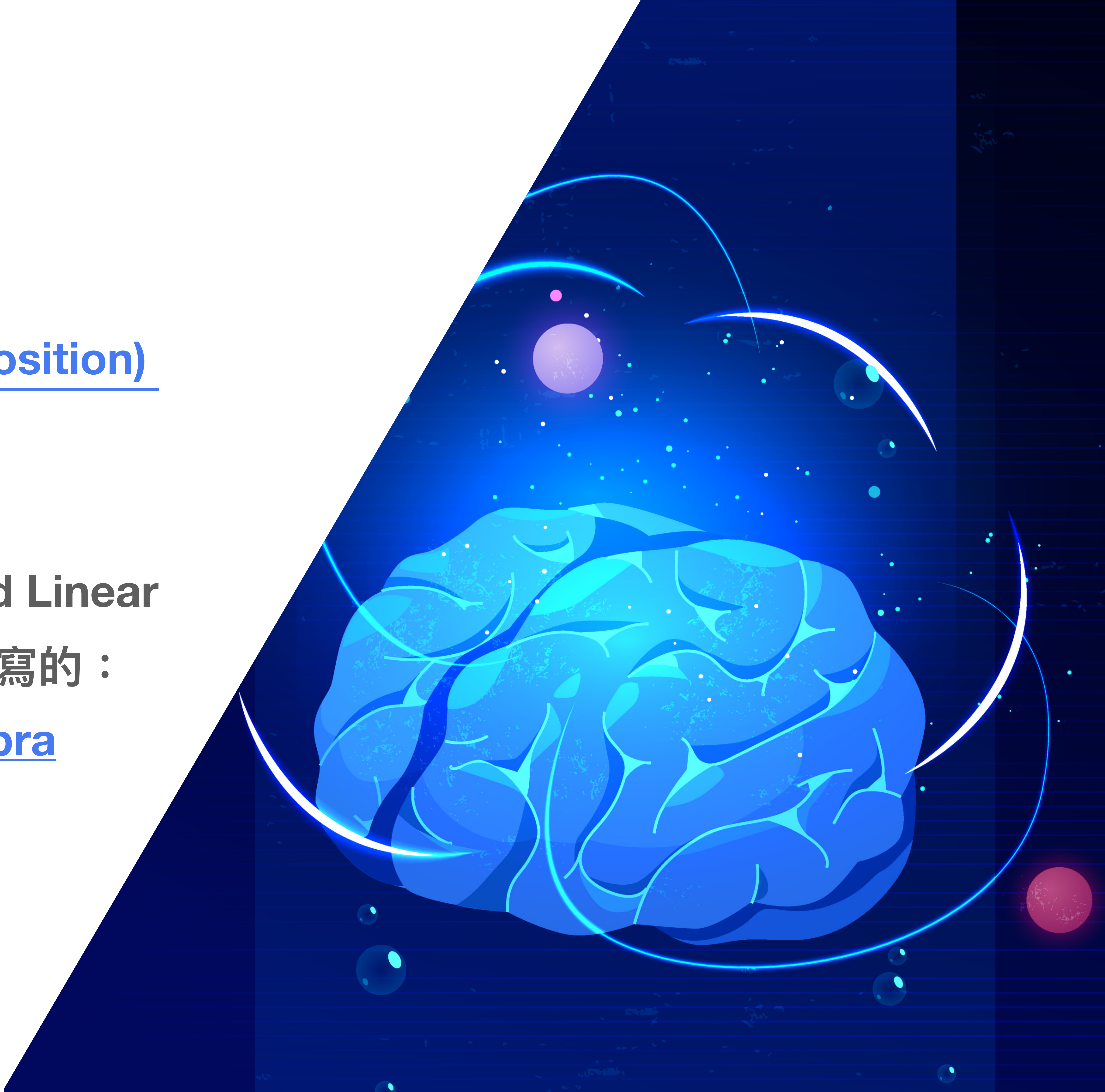
矩陣屬性	分解方法類型	矩陣屬性	分解方法類型
正定矩陣 (Positive-definite)	Cholesky	Bidiagonal	Bidiagonal
稠密對稱矩陣或Hermitian矩陣	Bunch-Kaufman	Tridiagonal	LU
稀疏對稱矩陣或Hermitian矩陣	LDLt	Symmetric real tridiagonal	LDLt
Triangular	Triangular	一般方陣	LU
Diagonal	Diagonal	一般非方陣	QR

知識點回顧

- 今天介紹 Julia Linear Algebra 線性代數函式庫，需要相關運算的話，可以使用內建的函式庫。
- 在範例程式中示範矩陣的建立、操作、eigenvalues / eigenvectors、特殊矩陣、以及矩陣分解。
- 在最後也列出內建的 factorize() 函式如何自動判斷適用的矩陣分解。

推薦閱讀

- Wikipedia 矩陣分解 (Matrix decomposition)
- 線代啟示錄-特殊矩陣
- 線代啟示錄-矩陣分解
- 史丹佛大學的 Introduction to Applied Linear Algebra 教科書的範例，是用 Julia 撰寫的：Introduction to Applied Linear Algebra



解題時間

請跳出 PDF 至官網 Sample Code
& 作業開始解題