

List slice //列表切片

list[: not include this number: plus]

```
[3] c=np.array([[[1, 2, 3],
                [4, 5, 6]],
               [[1, 2, 3],
                [4, 5, 6]],
               [[1, 2, 3],
                [4, 5, 6]]])
c.shape
(3, 2, 3)

[4] c[:, :, 2]
array([[3, 6],
       [3, 6],
       [3, 6]])

[5] c[:, :, :2]
array([[[1, 2],
       [4, 5]],
       [[1, 2],
       [4, 5]],
       [[1, 2],
       [4, 5]]])

[9] c[:, :, ::2]
array([[[1, 3],
       [4, 6]],
       [[1, 3],
       [4, 6]],
       [[1, 3],
       [4, 6]]])
```

Number

import numpy as np //匯入

np.array([,])

np.arange(, not include this number, plus)

len(np.arange(,)) //計算這多少數字

np.exp(x) //e^x

np.exp2(x) // 2^x

np.power(3, x) // 3^x

np.log(x) //log 自然對數

np.log2(x) //以 2 為基底的對數

np.log10(x) //以 10 為基底的對數

Number coordinate

imprort matplotlib.pyplot as plt //匯入

plt.scatter(, , color=' ') //座標點

plt.plot(,) //畫線

```
LinearRegression //線性回歸 (解決輸入值少，發生過擬合的情形)
from sklearn.linear_model import LinearRegression //匯入
model=LinearRegression() //啟動模組
x_modify=x.reshape(,) //線性預測處理，資料必須為二維陣列
[[x1],[x2],[x3].....]
x.reshape(-1) //懶得計算，反正請你幫我變成一維陣列
model.fit(x.reshape(,), y) //模型訓練參數設定及訓練
plt.plot(x,model.predict(x.reshape(,))) //畫出 x 值及用 x 線性回歸預測的 y
值
```

```
train_test_split //隨機劃分訓練集和測試集
from sklearn.model_selection import train_test_split //匯入
x_train, x_test, y_train, y_test=train_test_split(x, y, test_size=0.9, random
state=) //分割 x, y 訓練組 1-test_size 比例組，x, y 測試組 test_size 比例組，
設置隨機種子會固定隨機方式
//拿 x_train, y_train 帶入線性回歸
```

```
from sklearn.svm import SVC //支援向量機(將資料放入三度空間分類)
model=SVC() //啟動模組
list(zip(x,y)) //將資料鏈起來
x, y=meshgrid(np.arange(, ,), np.arange(, ,)) //生成網格點座標矩陣
from sklearn.datasets import load_iris //匯入鳶尾花資料
iris_data=load_iris //啟動模組
x=iris_data.data //x=鳶尾花訓練資料
label=iris_data.target //label=鳶尾花訓練目標
np.nonzero(label!=model.predict(x.reshape(-1))) //找出訓練函式與目標的不
同
model.predict(x.reshape(-1))[n] //找出訓練函式 n 這個值多少
label[n] //找出目標函式 n 這個值多少
```