









LAB 5: HUẤN LUYỆN MẠNG ANN

(SV thực hiện tại lớp)

Bài tập 1: KHỞI TẠO MẠNG HUẤN LUYỆN

1.1. Sử dụng ngôn ngữ lập trình Python, đưa dữ liệu đầu vào và khởi tạo mạng ngẫu nhiên của 1000 lần bước lặp

color								
length	3	2	4	3	3.5	2	5.5	1
width	1.5	1	1.5	1	.5	.5	1	1

Trong đó:

- Dữ liệu đầu vào là Chiều dài, chiều rộng và màu hoa

```
import matplotlib.pyplot as plt

data = [[3, 1.5, 1],
        [2, 1, 0],
        [4, 1.5, 1],
        [3, 1, 0],
        [3.5, .5, 1],
        [2, .5, 0],
        [5.5, 1, 1],
        [1, 1, 0]]
```

- In ra màn hình dữ liệu ngẫu nhiên theo hình sau trong 1000 bước lặp

```
[3.5, 0.5, 1]
[4, 1.5, 1]
[3, 1.5, 1]
[3, 1.5, 1]
[5.5, 1, 1]
[3, 1, 0]
[3, 1, 0]
[5.5, 1, 1]
[4, 1.5, 1]
[2, 1, 0]
[4, 1.5, 1]
[2, 1, 0]
.....
```

1.2. Thực hiện in ra màn hình 1000 dữ liệu theo dữ liệu bài 1.1 thỏa mãn công thức:

$$y(\text{dự đoán}) = f((a1 * w1 + a2 + w2)) + b)$$

Trong đó:

- **w1,w2,bias** dùng hàm random: randn()
- **a1, a2:** Chiều dài và chiều rộng của các cánh hoa theo dữ liệu đầu vào.
- **f((a1 * w1 + a2 + w2)) + b):** Trả về hàm **Sigmoid(x)** là kết quả dự đoán của máy.

Kết quả hiện ra màn hình:

```
.....
[3, 1.5, 1]
Kết quả dự đoán lần thứ 993 là 0.04473530023740187
[4, 1.5, 1]
Kết quả dự đoán lần thứ 994 là 0.01449501530296377
[4, 1.5, 1]
Kết quả dự đoán lần thứ 995 là 0.01449501530296377
[2, 0.5, 0]
Kết quả dự đoán lần thứ 996 là 0.09535056364056646
[3, 1, 0]
Kết quả dự đoán lần thứ 997 là 0.03788173483747191
[3, 1.5, 1]
Kết quả dự đoán lần thứ 998 là 0.04473530023740187
[3.5, 0.5, 1]
Kết quả dự đoán lần thứ 999 là 0.018214176608231235
```

Bài tập 2: SAI SỐ TOÀN PHƯƠNG TRUNG BÌNH (MEAN SQUARED ERROR)

2.1. Dựa vào công thức toàn phương trung bình (MSE) hãy dùng Python để tính toán công thức với dữ liệu đầu vào trong bảng: (Không dùng thư Viện)

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}$$

Mean Square Error

Act(kết quả)	1.1	2	1.7
Pre(Dự đoán)	1	1.7	1.5

```
act = np.array([1.1,2,1.7])  
pred = np.array([1,1.7,1.5])
```

2.2. Với dữ liệu đầu vào trên, hãy áp dụng bằng cách thực hiện tự động import thư viện: *sklearn.metrics*