

MẠNG NƠ-RON NHÂN TẠO

THỰC HÀNH PHƯƠNG PHÁP SỐ CHO KHOA HỌC DỮ LIỆU

Ngày 1 tháng 6 năm 2023

Liên hệ

GOOGLE CLASSROOM: **mrhvcvm**

TRỢ GIẢNG:

- ▶ Nguyễn Thị Kiều Trang: ntktrang@hcmus.edu.vn
- ▶ Lý Như Bình: lnbinh@hcmus.edu.vn

LƯU Ý:

- ▶ Email đăng nhập google classroom thể hiện đầy đủ họ và tên, tránh sử dụng email có biệt danh.
- ▶ Tiêu đề mail (bắt buộc):
[2023-HK2-THPPSKHDL] [Tiêu đề thư]
VD: [2023-HK2-THPPSKHDL] HỎI BÀI
Vui lòng giới thiệu họ tên, MSSV và tên ca học khi gửi email.

Một vài điều về lớp

Điểm thực hành: Chiếm 40% tổng điểm:

- ▶ Điểm danh: 1 điểm (Mỗi buổi)
- ▶ Bài tập: 3 điểm (Nộp bài tập thực hành mỗi tuần)

Cách thức nộp bài:

- ▶ Nộp trên google classroom
- ▶ Nộp file .txt
- ▶ Tên file: Y_MSSV_Hoten_baix.txt,
 - ▶ $Y = LB$ nếu bạn học phòng C204.
 - ▶ $Y = LT$ nếu bạn học phòng C203.
 - ▶ $x \in \{1, 2, 3, 4, \dots\}$

1. Phép toán XOR (hay cổng logic XOR)

- ▶ Đầu vào: 0 và 1.
- ▶ Đầu ra: phép toán XOR tuân theo quy luật sau:
 - ▶ Nếu 2 giá trị đầu vào giống nhau thì giá trị đầu ra là 0.
 - ▶ Nếu 2 giá trị đầu vào khác nhau thì giá trị đầu ra là 1. *Bảng chân lí:*

Đầu vào 1	Đầu vào 2	Đầu ra
0	0	0
1	1	0
1	0	1
0	1	1

Tạo một mạng nơ-ron mô phỏng phép toán XOR và dùng Python để mô phỏng lại quá trình huấn luyện mạng nơ-ron đó. Lưu ý: dùng thuật toán Gradient Descent để hiệu chỉnh trọng số liên kết (w) và giá trị độ lệch (b).

2. Hãy dùng Python để mô phỏng lại quá trình huấn luyện mạng nơ-ron của phép toán XOR như trên, nhưng dùng thuật toán Stochastic Gradient Descent để hiệu chỉnh trọng số liên kết (w) và giá trị độ lệch (b).

3. Phép toán OR (hay cổng logic OR)

- ▶ Đầu vào: 0 và 1.
- ▶ Đầu ra: phép toán OR tuân theo quy luật sau:

Đầu vào 1	Đầu vào 2	Đầu ra
0	0	0
0	1	0
1	0	1
1	1	1

Tạo một mạng nơ-ron mô phỏng phép toán OR và dùng Python để mô phỏng lại quá trình huấn luyện mạng nơ-ron đó. Lưu ý: dùng thuật toán Gradient Descent để hiệu chỉnh trọng số liên kết (w) và giá trị độ lệch (b).