Trường Đại học Khoa học Tự nhiên
Đại học Quốc gia TP.HCM

ĐỒ ÁN MÔN HỌC - ĐỀ 3

Môn học: Thực hành Nhập môn Công nghệ thông tin
Giảng viên: Phạm Minh Hoàng

Thông tin nhóm 9:

1. Trịnh Chấn Duy - 23120419
2. Đỗ Duy lợi - 23120293
3. Nguyễn Gia Thịnh - 23120167
4. Phạm Bảo Thắng - 23120416

Mục lục

1 Chức năng

Xây dựng một mô hình dự đoán giá thuê nhà đơn giản.

2 Nội dung thực hiện

2.1 Cài đặt môi trường lập trình, thư viện và chuẩn bị dữ liệu

- Cài đặt thành công môi trường lập trình: ngôn ngữ Python với công cụ lập trình Miniconda.
- Cài đặt thành công thư viện Pandas, thư viện Numpy, thư viện máy học và dữ liệu scikit-learn.
- Xử lý dữ liệu và chia làm 2 tập (90% dữ liệu là tập huấn luyện train & 10% dữ liệu là tập kiểm thử test).
- Xử lý mỗi tập trong 2 tập trên thành các mảng numpy: Mảng 2 chiều data và mảng 1 chiều label.
- Tập kiểm thử test được lựa chọn ngẫu nhiên.

2.2 Xử lý dữ liệu chuẩn bị cho hồi quy tuyến tính

- Thêm vào module **preprocessing** trong thư viện scikit-learn để chuẩn hóa dữ liệu.
- Tiền xử lý dữ liệu: Sử dụng MinMaxScaler để chuẩn hóa dữ liệu về khoảng [0, 1] theo công thức:

$$x^{(i)} = \frac{x^{(i)} - x^{\max}}{x^{\max} - x^{\min}} \tag{1}$$

  Trong đó: $x^{(i)}$ là giá trị của mẫu dữ liệu thứ i, $x^{\min}$ là giá trị nhỏ nhất trong dữ liệu và $x^{\max}$ là giá trị lớn nhất trong dữ liệu.

2.3 Xây dựng hàm dự đoán bằng hồi quy tuyến tính

Xây dựng mô hình hồi quy tuyến tính thông qua thuật toán **Gradient Descent** để tìm các trọng số w và hệ số tự do b cho hàm dự đoán:

2.3.1 Khởi tạo các tham số của mô hình

- Khởi tạo hệ số tỉ lệ học α (learning rate) = 0.0007.
- Khởi tạo ϵ (ngưỡng dừng) = 10^{-6}.
- Khởi tạo biến `max_loops` (số lần lặp tối đa) = 10000.
- Khởi tạo ngẫu nhiên $w^{(0)}$ là mảng 2 chiều có kích thước 1:6 - các trọng số của hàm dự đoán.
- Khởi tạo ngẫu nhiên $b^{(0)}$ là giá trị thực - hệ số tự do của hàm dự đoán.

2.3.2 Thuật toán Gradient Descent

- Khởi tạo biến lặp $k = 0$.
- Lặp cho đến khi $|L^{(k)} - L^{(k-1)}| < \epsilon$ hoặc $k =$ `max_loops`, tiến hành cập nhập giá trị sau mỗi vòng lặp theo các công thức:

$$w^{(k)} = w^{(k-1)} - \alpha \frac{dL}{dw^{(k-1)}}$$

$$b^{(k)} = b^{(k-1)} - \alpha \frac{dL}{db^{(k-1)}}$$

$$L^{(k)} = \frac{1}{2} \sum_{i=1}^{N} (w^{(k)}.x^{(i)} + b^{(k)} - y^{(i)})^2$$

$$k = k + 1$$

Trong đó: N là số mẫu trong tập train_data.

$$\frac{dL}{dw} = \sum_{i=1}^{N} (w \cdot x^{(i)} + b - y^{(i)}) \cdot x^{(i)}$$

$$\frac{dL}{db} = \sum_{i=1}^{N} (w \cdot x^{(i)} + b - y^{(i)})$$

- Ý nghĩa: Thuật toán Gradient Descent là một phương pháp tối ưu hóa được sử dụng để điều chỉnh các tham số của một mô hình máy học. Mục tiêu của thuật toán là tìm ra giá trị tối thiểu của hàm dự đoán giúp cho mô hình gần với thực tế nhất.

2.4 Tính MSE để đánh giá hiệu suất của mô hình

2.4.1 MSE là gì ?

MSE viết tắt của Mean Squared Error (Sai số toàn phương trung bình) là một phương pháp để đo lường sự chênh lệch giữa giá trị dự đoán và giá trị thực tế trong mô hình hồi quy tuyến tính.

2.4.2 Đánh giá hiệu suất mô hình

- Áp dụng mô hình để dự đoán trên tập kiểm thử bằng cách xây dựng mảng predict $\hat{y}$:

  $$\hat{y}^{(i)} = w \cdot x^{(i)} + b$$

  với $x^{(i)}$ là mẫu dữ liệu thứ i trong tập kiểm thử `test_data`.

- Tính MSE:

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

  Trong đó: M là số lượng dữ liệu của tập kiểm thử `test_data`.

2.5 Chạy thử nghiệm trên tập kiểm thử

- Thay đổi hệ số tỉ lệ học α (learning rate) để MSE càng nhỏ, giúp mô hình trở nên hiệu quả hơn.

- Lập bảng giá trị của các hệ số tỉ lệ học α gồm số lần lặp loops trong thuật toán Gradient Descent và giá trị MSE tương ứng.

3 Kết quả thực nghiệm

α	0.0001	0.0007	0.001	0.0015	0.002	0.0025	0.002302
Loops	10000	3431	2468	1835	1368	Error	1202
MSE	0.004937	0.004790	0.004787	0.004783	0.004781	Error	0.004780

Bảng 1: Khảo sát hệ số tỉ lệ học

Chọn $\alpha = 0.002302$ là giá trị tối ưu nhất tìm được.

Tài liệu

Thuật toán Gradient Descent `https://leon.bottou.org/publications/pdf/tricks-2012.pdf`