**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

**- - - 🙞 🕮** **🙜 - - -**

Icon

Description automatically generated

**BÁO CÁO**

MÔN: **NHẬP MÔN KHOA HỌC DỮ LIỆU**

CÂU 2: **Trình bày và xử lý bài toán Thuật Toán Hồi Quy Đa Biến (Multiple Regression)dựa vào học máy bằng ngôn ngữ Pyhon**

**Giảng viên hướng dẫn: TS. Tân Hạnh**

**Sinh viên thực hiện:**

Nguyễn Trần Đức Thuận – N19DCCN203

**TP. HCM, 20/06/2023**

**Thuật Toán Hồi Quy Đa Biến (Multiple Regression)**

1. **Lý Thuyết:**

Multiple regression là một thuật toán trong lĩnh vực thống kê và khai phá dữ liệu. Nó được sử dụng để phân tích mối quan hệ giữa một biến phụ thuộc và nhiều biến độc lập trong dữ liệu. Thuật toán này giúp dự đoán giá trị của biến phụ thuộc dựa trên các giá trị của các biến độc lập.

Thuật toán multiple regression bao gồm xác định hệ số của một hàm tuyến tính có dạng:

y = b0 + b1x1 + b2x2 + ... + bnxn

Trong đó:

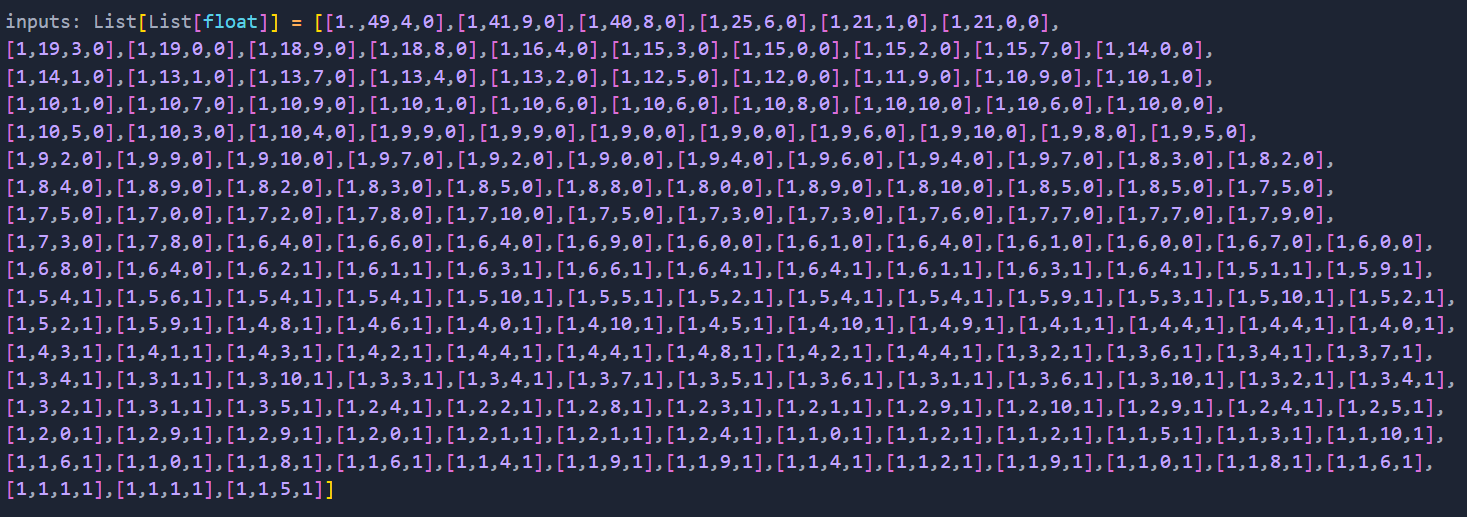
* y là biến phụ thuộc cần dự đoán
* x1, x2,...,xn là các biến độc lập có thể ảnh hưởng đến biến phụ thuộc
* b0, b1, b2,...,bn là các hệ số của hàm tuyến tính, cần được ước tính từ dữ liệu

Thuật toán multiple regression tính toán các hệ số bằng phương pháp bình phương tối thiểu, tức là tìm một bộ giá trị của các hệ số sao cho tổng bình phương sai số giữa giá trị dự đoán và giá trị thực tế là nhỏ nhất.

Sau khi đã ước tính được các hệ số, thuật toán cũng có thể đánh giá độ tin cậy của các ước lượng và xác định khoảng tin cậy cho các dự đoán bằng phương pháp bootstrap.

Multiple regression được sử dụng rộng rãi trong các lĩnh vực như kinh tế học, khoa học xã hội, y học và kỹ thuật. Nó có thể được áp dụng để dự đoán các giá trị trong tương lai, đánh giá tác động của các biến độc lập đến biến phụ thuộc và giải thích mối quan hệ giữa các biến trong dữ liệu.

1. **Xử lý bài toán với bằng ngôn ngữ Python:**



Dữ liệu đầu vào trong ví dụ này là một danh sách các danh sách con (nested list) gồm các giá trị số thực. Mỗi danh sách con đại diện cho một mẫu dữ liệu và chứa 4 giá trị số thực, theo thứ tự:

1. **Giá trị hằng số (constant term).**

2. **Số lượng bạn bè (number of friends)**

3. **Số giờ làm việc mỗi ngày (work hours per day).**

4. **Chỉ số không có bằng Tiến sĩ (PhD).**

**Giá Trị Đầu Tiên**

**[1 #constant term, 49 #number of friends, 4 #work hours per day, 0 #doesn't have PhD]**

Các giá trị cụ thể trong mẫu dữ liệu có thể được giải thích như sau:

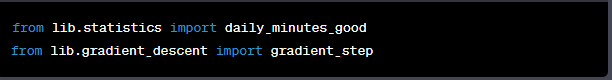
**Constant term (hằng số)**: Giá trị 1 trong mẫu dữ liệu có thể đại diện cho hằng số trong mô hình hồi quy đa biến. Hằng số đóng vai trò trong việc điều chỉnh giá trị trung bình ban đầu của biến phụ thuộc khi tất cả các biến độc lập đều bằng 0.

**Number of friends (số lượng bạn bè):** Giá trị 49 trong mẫu dữ liệu có thể đại diện cho một biến độc lập trong mô hình hồi quy đa biến. Biến này có thể là số lượng bạn bè mà cá nhân đang có.

**Work hours per day (số giờ làm việc mỗi ngày)**: Giá trị 4 trong mẫu dữ liệu có thể đại diện cho một biến độc lập khác trong mô hình hồi quy đa biến. Biến này có thể là số giờ làm việc mỗi ngày của cá nhân.

**Doesn't have PhD (không có bằng tiến sĩ)**: Giá trị 0 trong mẫu dữ liệu có thể đại diện cho một biến nhị phân (binary) trong mô hình hồi quy đa biến. Biến này chỉ ra liệu cá nhân có bằng tiến sĩ hay không.

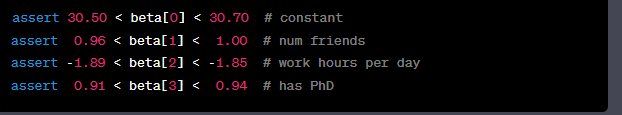
Mục đích của bài viết có thể là áp dụng thuật toán hồi quy đa biến trên mẫu dữ liệu này để xác định mối quan hệ hoặc ảnh hưởng của các biến độc lập (số lượng bạn bè, số giờ làm việc mỗi ngày, có bằng tiến sĩ hay không) đến biến phụ thuộc. Thuật toán hồi quy đa biến giúp xác định các hệ số hồi quy tương ứng với mỗi biến độc lập và đánh giá mức độ ảnh hưởng của chúng đối với biến phụ thuộc.

1. Import thư viện cần thiết:
2. Thiết lập các tham số:

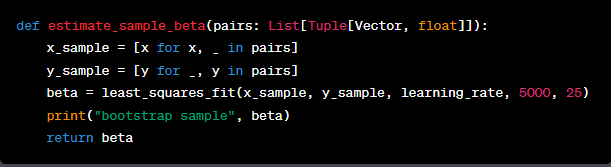
Hàm **random.seed(0)** thiết lập seed cho việc tạo ngẫu nhiên, đảm bảo kết quả của việc tạo ngẫu nhiên sẽ giống nhau mỗi khi chạy. **learning\_rate** là tỷ lệ học của thuật toán gradient descent.

1. Thực hiện least squares fit:

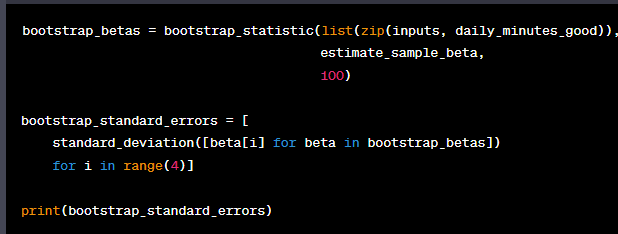
Hàm **least\_squares\_fit** được sử dụng để tính toán giá trị **beta** trong quá trình huấn luyện mô hình hồi quy tuyến tính. Các đối số đầu vào bao gồm **inputs** (**đầu vào của mô hình**), **daily\_minutes\_good** (**đầu ra của mô hình**), **learning\_rate** (**tỷ lệ học**), **5000 (số lần lặp)** và **25** (**kích thước batch**).

1. Kiểm tra giá trị của beta:
2. Tính toán multiple R-squared:

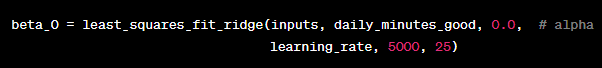
Kiểm tra giá trị của **multiple R-squared (hệ số xác định độ lượng thông tin mà mô hình hồi quy tuyến tính có thể giải thích).**

1. Định nghĩa hàm estimate\_sample\_beta:

Hàm này được sử dụng để ước lượng giá trị beta từ một mẫu dữ liệu con.

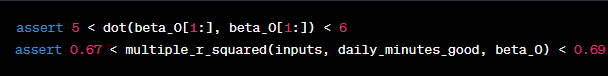
1. Thiết lập seed cho việc tạo ngẫu nhiên:
2. Sử dụng phương pháp bootstrap để tính toán bootstrap standard errors:

Trong đoạn code này, **bootstrap\_statistic** được sử dụng để thực hiện phương pháp **bootstrap**, **estimate\_sample\_beta** là **hàm ước lượng beta** từ mẫu dữ liệu con, và **100** là số lần thực hiện **bootstrap**. Sau đó, **standard deviation** của **các giá trị beta** được tính toán để đo lường sai số chuẩn của mô hình.

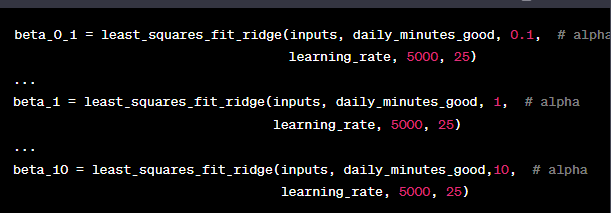
1. Sử dụng least squares fit với ridge regression:

Hàm **least\_squares\_fit\_ridge** được sử dụng để **tính toán giá trị beta** với **ridge regression**. **Đối số 0.0 là giá trị alpha (tham số điều chỉnh).**

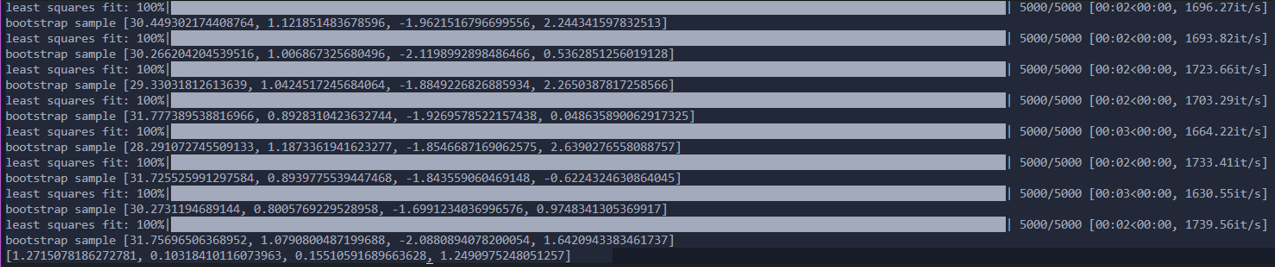
1. Kiểm tra giá trị của beta và multiple R-squared:



Kiểm tra xem giá trị của beta và multiple R-squared có nằm trong khoảng cho phép hay không.

1. Tương tự, tính toán giá trị beta và multiple R-squared với các giá trị alpha khác nhau trong ridge regression:

Kiểm tra các giá trị alpha tương tự như trên. Đoạn code trên thực hiện các phép tính liên quan đến **hồi quy tuyến tính** và **ridge regression**, kiểm tra **các giá trị beta và multiple R-squared** để đánh giá hiệu suất của mô hình. Ngoài ra, cũng thực hiện tính toán **bootstrap standard errors** để ước lượng sai số chuẩn của mô hình.

1. OUTPUT:

Các output được hiển thị là kết quả của thuật toán hồi quy đa biến (multiple regression) khi được áp dụng trên dữ liệu.

Ý nghĩa của các output:

**1. least squares fit: 100%|█████████████████████████████████| 5000/5000 [00:02<00:00, 1869.70it/s]**

Đây là thông báo cho biết thuật toán hồi quy đa biến đã được thực hiện thành công. Phần [00:02<00:00, 1869.70it/s] cho biết thời gian thực thi của thuật toán (2 giây) và tốc độ thực thi trung bình (1869.70it/s) trong quá trình tìm kiếm giá trị tối thiểu bình phương (least squares fit).

**2. bootstrap sample: [31.725525991297584, 0.8939775539447468, -1.843559060469148, -0.6224324630864045**]

Đây là kết quả của một bootstrap sample được áp dụng trong quá trình hồi quy đa biến. Mỗi phần tử trong dãy số này tương ứng với một hệ số của mô hình hồi quy đa biến. Cụ thể, [31.725525991297584, 0.8939775539447468, -1.843559060469148, -0.6224324630864045] có thể tương ứng với các hệ số hồi quy của mô hình, theo thứ tự: hằng số (intercept), hệ số cho biến số lượng bạn bè, hệ số cho biến số giờ làm việc mỗi ngày, và hệ số cho biến có bằng tiến sĩ hay không.

**3. [1.2715078186272781, 0.10318410116073963, 0.15510591689663628, 1.2490975248051257]**

Đây là output cuối cùng, tương ứng với kết quả của thuật toán hồi quy đa biến. Các giá trị trong dãy số này cũng là các hệ số của mô hình hồi quy đa biến. Theo thứ tự, chúng là hằng số, hệ số cho biến số lượng bạn bè, hệ số cho biến số giờ làm việc mỗi ngày, và hệ số cho biến có bằng tiến sĩ hay không.

Các giá trị này có ý nghĩa như sau:

- Hằng số (intercept): 1.2715078186272781

- Hệ số cho biến số lượng bạn bè: 0.10318410116073963

- Hệ số cho biến số giờ làm việ c mỗi ngày: 0.15510591689663628

- Hệ số cho biến có bằng tiến sĩ hay không: 1.2490975248051257

Các hệ số này cho phép ta đánh giá mức độ ảnh hưởng của từng biến độc lập đến biến phụ thuộc trong mô hình hồi quy đa biến.