# **Trường đại học Thủy Lợi**

# 

# **BÁO CÁO BÀI TẬP LỚN MÔN HỌC MÁY**

# **ĐỀ TÀI: Sử dụng hồi quy tuyến tính để dự đoán giá phòng trọ tại Hà Nội theo diện tích**

Nhóm thực hiện: Nhóm 20

Thành viên nhóm: Nguyễn Mai Trực, Nguyễn Đức Quân

Giảng viên hướng dẫn: Nguyễn Hữu Quỳnh, Cù Việt Dũng

*Hà Nội, Ngày 03 tháng 5 năm 2020*

1. **Tổng quan**
   1. Giới thiệu về học máy

Machine learning là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kỹ thuật cho phép các hệ thống “học” tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Ví dụ như các máy có thể “học” cách phân loại [thư điện tử](https://vi.wikipedia.org/wiki/Th%C6%B0_%C4%91i%E1%BB%87n_t%E1%BB%AD) xem có phải [thư rác (spam)](https://vi.wikipedia.org/wiki/Th%C6%B0_r%C3%A1c) hay không và tự động xếp thư vào thư mục tương ứng. Học máy rất gần với [suy diễn thống kê](https://vi.wikipedia.org/wiki/Suy_di%E1%BB%85n_th%E1%BB%91ng_k%C3%AA) (statistical inference) tuy có khác nhau về thuật ngữ. cách phân loại [thư điện tử](https://vi.wikipedia.org/wiki/Th%C6%B0_%C4%91i%E1%BB%87n_t%E1%BB%AD) xem có phải [thư rác (spam)](https://vi.wikipedia.org/wiki/Th%C6%B0_r%C3%A1c) hay không và tự động xếp thư vào thư mục tương ứng. Học máy rất gần với [suy diễn thống kê](https://vi.wikipedia.org/wiki/Suy_di%E1%BB%85n_th%E1%BB%91ng_k%C3%AA) (statistical inference) tuy có khác nhau về thuật ngữ.

Học máy có liên quan lớn đến [thống kê](https://vi.wikipedia.org/wiki/Khoa_h%E1%BB%8Dc_Th%E1%BB%91ng_k%C3%AA" \o "Khoa học Thống kê), vì cả hai lĩnh vực đều nghiên cứu việc phân tích dữ liệu, nhưng khác với thống kê, học máy tập trung vào sự phức tạp của các giải thuật trong việc thực thi tính toán. Nhiều bài toán suy luận được xếp vào loại bài toán [NP-khó](https://vi.wikipedia.org/wiki/NP-kh%C3%B3), vì thế một phần của học máy là nghiên cứu sự phát triển các giải thuật suy luận xấp xỉ mà có thể xử lý được.

Dưới góc nhìn của [trí tuệ nhân tạo](https://vi.wikipedia.org/wiki/Tr%C3%AD_tu%E1%BB%87_nh%C3%A2n_t%E1%BA%A1o" \o "Trí tuệ nhân tạo), động lực chính học máy bởi là nhu cầu thu nhận tri thức (knowledge acquisition). Thật vậy, trong nhiều trường hợp ta cần kiến thức chuyên gia là khan hiếm (không đủ chuyên gia ngồi phân loại lừa đảo thẻ tín dụng của tất cả giao dịch hàng ngày) hoặc chậm vì một số nhiệm vụ cần đưa ra quyết định nhanh chóng dựa trên xử lý dữ liệu khổng lồ (trong mua bán chứng khoán phải quyết định trong vài khoảng khắc của giây chẳng hạn) và thiếu ổn định thì buộc phải cần đến máy tính. Ngoài ra, đại đa số dữ liệu sinh ra ngày nay chỉ phù hợp cho máy đọc (computer readable) tiềm tàng ngưồn kiến thức quan trọng. Máy học nghiên cứu cách thức để mô hình hóa bài toán cho phép máy tính tự động hiểu, xử lý và học từ dữ liệu để thực thi nhiệm vụ được giao cũng như cách đánh giá giúp tăng tính hiệu quả.

Những năm gần đây, AI nổi lên như một bằng chứng của cuộc cách mạng công nghiệp lần thứ tư. Trí tuệ nhân tạo có thể được định nghĩa như một nghành của khoa học máy tính liên quan đến việc tự động hóa các hành vi thông minh.

Trí tuệ nhân tạo là một bộ phận của khoa học máy tính và do đó nó phải được đặt trên những nguyên lý lý thuyết vững chắc, có khả năng ứng dụng được của lĩnh vực này. Ở thời điểm hiện tại, thuật ngữ này thường dùng để nói đến các máy tính có mục đích không nhất định và ngành khoa học nghiên cứu về các lý thuyết và các ứng dụng của trí tuệ nhân tạo. Theo đà phát triển của công nghệ, ứng dụng trí tuệ nhân tạo luôn là xu hướng công nghệ tương lai mà các hãng công nghệ trên toàn thế giới đua nhau sáng tạo, nó là nền tảng cốt lõi của cuốc cách mạng công nghệ 4.0. ML (Machine Learning) là một lĩnh vực của trí tuệ nhân tạo, được sinh ra từ khả năng nhận diện mẫu và từ lý thuyết các máy tính có thể học mà không cần phải lập trình để xử lý các nhiệm vụ cụ thể nào đó. Hầu hết mọi nghành công nghiệp đang làm việc với hàm lượng lớn dữ liệu đều nhận ra tầm quan trọng của công nghệ ML. Những cái nhìn sáng suốt từ nguồn dữ liệu này – chủ yếu dạng thời gian thực – sẽ giúp các tổ chức vận hành hiệu quả hơn hoặc tạo lợi thế cạnh tranh so với các đối thủ. Các ứng dụng của ML đã quá quen thuộc với con người: xe tự hành của Google và Tesla, hệ thống tự tag khuôn mặt trên Facebook, hệ thống gợi ý sản phẩm của Amazon, hệ thống gợi ý phim của Netflix…, chỉ là một vài trong vô vàn những ứng dụng của trí tuệ nhân tạo và cụ thể là ML.

* 1. Trình bày bài toán

Supervised learning

* + - * + Kích thước: 150 mẫu, 2 thuộc tính (attributes)
        + Đầu vào (input): numeric
        + Đầu ra (output): Hàm hồi quy tuyến tính (Linear regression) y = *f*(x)
        + Dạng của hàm y = *f*(x): x =[x1] là một vector chứa thông tin input, y là một số vô hướng (scalar) biểu diễn output.
        + Bộ dữ liệu dataset: Được lấy từ các trang mạng cho thuê phòng trọ
  1. Hồi quy tuyến tính

Phân tích hồi quy tuyến tính là một phương pháp phân tích quan hệ giữa biến phụ thuộc Y với một hay nhiều biến độc lập X. Mô hình hóa sử dụng hàm tuyến tính. Các tham số của mô hình được ước lượng từ dữ liệu. Hồi quy tuyến tính được sử dụng rộng rãi trong thực tế do tính chất đơn giản hóa của hồi quy.

3.1. Dạng của Linear Regression

3.2. Sai số dự đoán

3.3. Hàm mất mát

1. **Thực nghiệm**
   1. Giới thiệu về bộ dữ liệu được sử dụng trong bài toán

Bộ dữ liệu “dataBTL” được thu thập và hoàn chỉnh từ các trang mạng cho thuê nhà trọ, rao bán nhà đất như “ <https://batdongsan.com.vn/cho-thue-nha-tro-phong-tro-ha-noi> ” trong vòng 4,5 tháng trở lại tại Hà Nội. Số lượng dữ liệu là 150 mẫu với 2 thuộc tính là giá phòng trọ và diện tích phòng trọ. Số lượng các quan sát cho các lớp là không đồng đều. Chi tiết về các thuộc tính theo thứ tự từ trái qua phải như sau:

Price: Giá phòng trọ (đơn vị: triệu việt nam đồng)

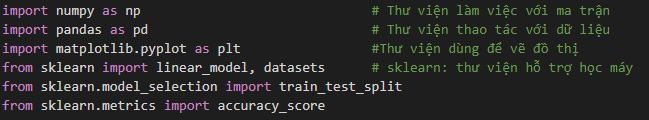
Acreage: Diện tích phòng trọ (Đơn vị: m2 )

* 1. Mô tả tập ví dụ huấn luyện và tập test

Ta tách 150 mẫu dữ liệu trong “dataBTL” thành 2 phần, training set(tập ví dụ huấn luyện) và test set(tập test). Thuật toán sẽ dựa vào thông tin ở training set để training ra hàm dự đoán. Sau khi đã có hàm dự đoán, dữ liệu từ training set sẽ được thay vào hàm và so sánh với tập test set cũng được thay vào hàm dự đoán.

* 1. Mô tả chương trình demo

Bước 1: Khai báo các thư viện cần thiết



Bước 2: Load dữ liệu từ dataset



Dữ liệu được load từ file “dataBTL” được đặt cùng đường dẫn với file code

Bước 3: Tạo ma trận one



Tạo ra một ma trận có tên là “**one**” có số hàng bằng số điểm dữ liệu, số cột bằng 1, giá trị các phần tử bằng 1. Sau đó thêm ma trận vừa tạo vào dữ liệu, đặt tên cột là “**B**”, value là giá trị của ma trận “**one**” vừa tạo.

Bước 4: Gán giá trị cho data\_X và data\_Y



khai báo và gán “**data\_X**” bằng tất cả dữ liệu của cột “**B**” và cột “**Acreage**”. “**data\_Y**” bằng tất cả dữ liệu của cột “**Price**”.

Bước 5: Tách training set và test set



Hàm train\_test\_split() sẽ tách tập dự liệu mẫu thành 2 phần train set và test set. Số điểm dữ liệu trong test set sẽ được gán bằng test\_size, trong bài toán này, dữ liệu sẽ được tách với test set = 50 và train set = 100. Các giá trị lấy vào test set sẽ được lấy ngẫu nhiên trong tập ví dụ huấn luyện.

Bước 6: Khai báo biến “regr”



Giá trị “fit\_intercept = False” nghĩa là cho phép hàm không tính toán các điểm cắt trục y(Acreage)

Bước 7: Tìm hàm dự đoán cho bài toán



Từ training set, ta sẽ train cho ra hàm dự đoán. Ta đưa giá trị X\_train, y\_train đã được tách từ bước 5 vào hàm fit.

Bước 8: Kiểm thử



Sau khi hàm fit đã train ra hàm dự đoán, ta sẽ dùng test set để kiểm thử, hàm predict là hàm dự đoán với tham số đầu vào là “**X\_test**”. Giá trị trả về sẽ được gán cho Y\_pred

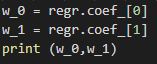
Bước 9: Vẽ đồ thị hồi quy tuyến tính



Biểu diễn các điểm dữ liệu trong dataframe dựa vào cột Acreage và Price, các giá trị sẽ được thay vào hàm dự đoán sau đó cho ra tọa độ các điểm dữ liệu và được biểu diễn bằng các chấm đỏ (ro) trên biểu đồ, nếu cho tham số “ro-” các điểm dữ liệu sẽ được nối với nhau.



Biểu diễn một đường thẳng được xác định bởi các điểm dữ liệu của tập test, tọa độ các điểm này được xác định với tung độ bằng giá trị của “**Y\_pred**”, hoành độ bằng giá trị của “**X\_test.Acreage**”



In giá trị của w\_0 và w\_1 cần tìm ra màn hình



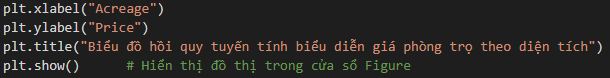
Xác định giới hạn cho đồ thị, 2 mẫu có giá trị từ 10 đến 60 ứng với trục hoành (Acreage), bởi vì giá trong tập ví dụ huấn luyện chỉ dao động từ giá trị từ 10 m2 đến 20 m2.



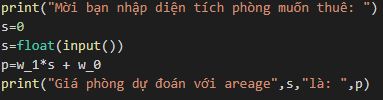
Giá trị y0 được tính bằng công thức như trên để ứng với hàm dự đoán đã được training



Biểu diễn một đường thẳng được xác định với x0, y0 đã được định nghĩa ở trên và giá trị color được gán bằng yellow.



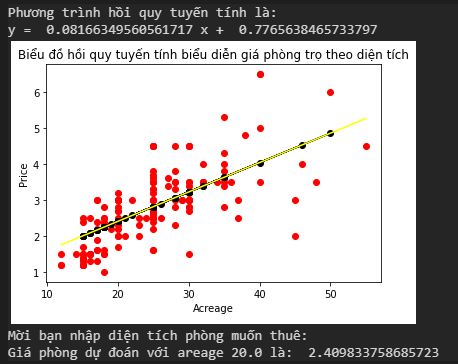
Thêm các giá trị cho biểu đồ với **plt.xlabel(“Acreage”)** là tên của trục hoành, **plt.ylabel(“Price”)** là tên của trục tung, **plt.title(“”)** là tên title của biểu đồ**. Plt.show()** là hàm hiển thị đồ thị được vẽ trong cửa số figure.



Cho phép bạn nhập một giá trị diện tích muốn thuê từ bàn phím có kiểu float, kết quả hiện thị sẽ là giá tiền dự đoán (đơn vị triệu Việt Nam đồng) được hiển thị ra màn hình.

* 1. Mô tả kết quả dự đoán

Sau khi hàm fit() training ra được hàm dự đoán, ta sẽ có dữ liệu hệ số **w1** và **w0** cần tìm. Ta sẽ show kết quả tìm được ra màn hình figure.



Giá trị của **w1** và **w0** mỗi lần chạy sẽ có kết quả khác nhau, do dữ liệu được đưa vào tập test và tập train là khác nhau.



Để kiểm tra sự chính xác của hàm hồi quy tuyến tính đã train được, ta sẽ dùng tập test để so sánh với tập train. Ta có thể sử dụng 1 trong 2 cách trên để so sánh. Cách thứ nhất ta sẽ vẽ một đường màu đen để so sánh xem đường này có nằm gần sát hoặc trùng với đường vàng là đường được vẽ bằng tập train hay không. Cách thứ hai tương tự cách 1 nhưng ta sẽ vẽ các điểm dữ liệu của tập test lên modal để xem các điểm đen có nằm trên đường vàng hay không. Nếu nằm trên có nghĩa là đã có sự chính xác khá cao đối với hàm hồi quy tuyến tính đã xây dựng được.