**BỘ GIÁO DỤC VÀ ĐÀO TẠO BỘ NÔNG NGHIỆP VÀ PTNT**

**TRƯỜNG ĐẠI HỌC THỦY LỢI**

****

**BÀI TẬP LỚN**

**MÔN HỌC MÁY**

**ĐỀ TÀI:**

**ỨNG DỤNG PHƯƠNG PHÁP HỒI QUY TUYẾN TÍNH DỰ ĐOÁN GIÁ XE OTO Ở LONDON**

**Nhóm sinh viên thực hiện:**

1.Lâm Văn Thái - 60TH1

2.Phạm Thị Phương Nga - 60TH1

3.Đặng Thị Thúy - 60TH1

**Giảng viên hướng dẫn:**

TS. Nguyễn Thị Kim Ngân

HÀ NỘI, NĂM 2020

**MỤC LỤC**

**PHẦN I: TỔNG QUAN**

[**1. Giới thiệu về học máy**](#_ta1z0t10347l) **4**

[**2. Giới thiệu về thuật toán Linear regression**](#_u63vezbn7xgo) **5**

[2.1 Khái niệm](#_lhw3a0q13049) 5

[2.2 Thuật toán](#_yzbdqp9ay5bt) 5

**PHẦN II: THỰC NGHIỆM**

[**2.1. Mô tả đề tài:**](#_gv2njlms089) **9**

[2.1.1 Tên đề tài](#_jxfl7icccgtz) 9

[2.1.2 Thuật toán đề tài](#_jv1x9vde7z32) 10

[**2.2.Mô tả tập dữ liệu của bài toán**](#_mkj68w1rngl) **11**

[**2.3. Chia tập dữ liệu**](#_14mv9g84293d) **11**

[**2.4. Mô tả ma trận dữ liệu huấn luyện (xtrain, ytrain) và ma trận dữ liệu kiểm tra (xtest, ytest)**](#_xj22ofcvilmv) **12**

[2.4.1 Tập dữ liệu huấn luyện(Training)](#_iixmb2ujj3ou) 12

[2.4.2 Tập dữ liệu kiểm tra(Testing)](#_wmtbyfb5yc9w) 12

**2.5. Dùng phương pháp Hồi quy tuyến tính để xây dựng mô hình cho bài toán với tập dữ liệu huấn luyện.** [**13**](#_o1uv6ybrcm1x)

**PHẦN III: ĐÁNH GIÁ**

[**3.1 Mô tả kết quả dự đoán**](#_injr9gxl3vw2) **13**

[**3.2 Mô tả biểu đồ đánh giá mô hình**](#_bo6ketla99ou) **16**

[**3.3 Đánh giá chất lượng mô hình**](#_3uavoi76hess) **16**

[**KẾT LUẬN**](#_je3hcsd7l80n) **17**

[**TÀI LIỆU THAM KHẢO**](#_lnxbz9) **17**

**LỜI NÓI ĐẦU**

Những năm gần đây, trí tuệ nhân tạo (artificial intelligence–AI) nổi lên như một bằng chứng của cuộc cách mạng công nghiệp lần thứ tư (1–động cơ hơi nước, 2–năng lượng điện, 3–công nghệ thông tin). Trí tuệ nhân tạo đã và đang trở thành thành phần cốt lõi trong các hệ thống công nghệ cao. Nó đã len lỏi vào hầu hết các lĩnh vực trong đời sống mà có thể chúng ta không nhận ra. Xe tự hành của Google và Tesla, hệ thống tự tag khuôn mặt trong ảnh của Facebook; trợ lý ảo Siri của Apple, hệ thống gợi ý sản phẩm của Amazon, hệ thống gợi ý phim của Netflix, hệ thống dịch đa ngôn ngữ Google Translate, máy chơi cờ vây AlphaGo và gần đây là AlphaGo Zero của Google DeepMind, v.v., chỉ là một vài ứng dụng nổi bật trong vô vàn những ứng dụng của trí tuệ nhân tạo.

Học máy (machine learning–ML) là một tập con của trí tuệ nhân tạo. Nó là một lĩnh vực nhỏ trong khoa học máy tính, có khả năng tự học hỏi dựa trên dữ liệu được đưa vào mà không cần phải được lập trình cụ thể (Machine Learning is the subfiled of computer science, that “gives computers the ability to learn without being explicitly programmed” –Wikipedia).

Những năm gần đây, sự phát triển của các hệ thống tính toán cùng với lượng dữ liệu khổng lồ được thu thập bởi các hãng công nghệ lớn đã giúp machine learning tiến thêm một bước dài. Một lĩnh vực mới được ra đời được gọi là học sâu (deep learning–DL). Deep learning đã giúp máy tính thực thi những việc tưởng chừng như không thể vào mười năm trước: phân loại cả ngàn vật thể khác nhau trong các bức ảnh, tự tạo chú thích cho ảnh, bắt chước giọng nói và chữ viết của con người, giao tiếp với con người, chuyển đổi ngôn ngữ, hay thậm chí cả sáng tác văn thơ hay âm nhạc.

**PHẦN 1:TỔNG QUAN**

# **1. Giới thiệu về học máy**

* Học máy là một lĩnh vực trong khoa học máy tính, là một trường con của trí tuệ nhân tạo (AI). Mục đích của việc học máy được sử dụng để phân tích và trích xuất kiến thức từ một tập dữ liệu lớn.
* Học máy có liên quan chặt chẽ với thống kê tính toán, có mối quan hệ chặt chẽ với tối ưu hóa toán học và khám phá các thuật toán có thể đưa ra dự đoán cao cấp về dữ liệu. Thuật toán học máy cho phép máy tính đào tạo trên đầu vào dữ liệu và sử dụng phân tích thống kê để xuất các giá trị nằm trong phạm vi cụ thể. Do vậy, học máy tạo điều kiện cho các máy tính trong việc xây dựng các mô hình từ dữ liệu mẫu để tự động hóa các quá trình ra quyết định dựa trên dữ liệu đầu vào.
* Các hệ thống học máy được phân thành bốn loại tùy thuộc vào bản chất của phương thức học tập bao gồm:
* Học có giám sát.
* Học không giám sát.
* Học bán giám sát.
* Học tăng cường.
* Hai trong số các phương pháp học máy được áp dụng rộng rãi nhất là học có giám sát và học không giám sát. Học có giám sát đào tạo các thuật toán dựa trên dữ liệu đầu vào và đầu ra mẫu được gắn nhãn bởi con người, còn đối với học không giám dữ liệu không có nhãn mà nó phải tự tìm cấu trúc dựa trên các dữ liệu đầu vào mà nó nhận được.

# **2. Giới thiệu về thuật toán Linear regression**

## 2.1 Khái niệm

* Linear regression là một thuật toán supervised, ở đó quan hệ giữa đầu vào và đầu ra được mô tả bởi một hàm tuyến tính. Thuật toán này còn được gọi là linear fitting hoặc linear least square.

## 2.2 Thuật toán

* Hàm dự đoán: *y* = *f*(x).
* Vector đặc trưng x = [x1, x2, x3] T là một vector cột chứa thông tin đầu vào
* Vector y là giá trị thực của đầu ra (ground truth), trong khi yˆ là giá trị đầu ra dự đoán (predicted output) của mô hình linear regression.
* Ta có thể mô hình quan hệ giữa đầu ra và đầu vào bằng một hàm tuyến tính đơn giản:

*f*(x) =w\_0 + w\_1 \* x0 (1)

+ w\_0, w\_1 là các hằng số.

+ w\_0 còn được gọi là bias b

* Mối quan hệ giữa y = f(x) là mối quan hệ tuyến tính (linear), bài toán chúng ta đang làm thuộc loại regression. Bài toán đi tìm các hệ số tối ưu {w\_0, w\_1} chính vì vậy được gọi là bài toán Linear Regression.
* Linear hay tuyến tính hiểu một cách đơn giản là thẳng, phẳng. Trong không gian hai chiều, một hàm số được gọi là tuyến tính nếu đồ thị của nó có dạng một đường thẳng. Trong không gian ba chiều, một hàm số được gọi là tuyến tính nếu đồ thị của nó có dạng một mặt phẳng. Trong không gian nhiều hơn 3 chiều, khái niệm mặt phẳng không còn phù hợp nữa, thay vào đó, một khái niệm khác ra đời được gọi là siêu mặt phẳng (hyperplane). Các hàm số tuyến tính là các hàm đơn giản nhất, vì chúng thuận tiện trong việc hình dung và tính toán. Chúng ta sẽ được thấy trong các bài viết sau, tuyến tính rất quan trọng và hữu ích trong các bài toán Machine Learning.

- Dạng của Linear Regression:

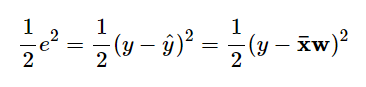
* Vector đặc trưng x=[x1; x2; x3; … ;xn] => x T = x1, x2, x3, … ,xn
* Vector trọng số w = [w1; w2; w3; … ;wn]
* Hàm số f(x) mô tả mối quan hệ giữa các đại lượng đầu vào:

*y ≈ yˆ = f(x) = w1x1 + w2x2 + w3x3 = x T. w*

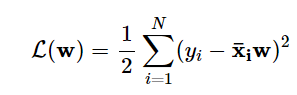
* Sai số dự đoán:

y ≈ y^ = x T.w

* Chúng ta muốn sự sai khác e giữa giá trị thực y và giá trị dự đoán y^ là nhỏ nhất:



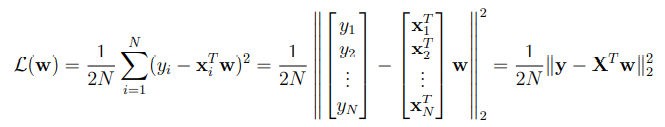
* Hệ số 1/2 là để thuận tiện cho việc tính toán (khi tính đạo hàm thì số 1/2 sẽ bị triệt tiêu) Sự sai khác giữa y và *e* 2 không đồng nghĩa với e nhỏ nhất. Vì e = y − y^ có thể là một số âm, khi e = −∞ là rất nhỏ nhưng sự sai lệch là rất lớn. Do đó chúng ta cần *e* 2
* Hàm mất mát:
* Chúng ta muốn tống sai số là nhỏ nhất, tương đương với việc tìm w để hàm số sau đạt giá trị nhỏ nhất:

**

* Hàm số *L*(w) chính là **hàm mất mát** của linear regression. Chúng ta luôn mong muốn rằng sự mất mát là nhỏ nhất, điều này có thể đạt được bằng cách tối thiểu hàm mất mát theo w:



* Trước khi đi xây dựng nghiệm cho bài toán tối ưu hàm mất mát, ta thấy rằng hàm số này có thể được viết gọn lại dưới dạng ma trận, vector, và norm như dưới đây:



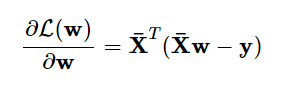
* với y = [y1; y2; . . . ; yN ] T , X = [x1, x2, . . . , xN ]. Như vậy *L*(w) ta được một dạng đơn giản của hàm mất mát

* Nghiệm cho bài toán Linear Regression:
* Cách phổ biến nhất để tìm nghiệm của bài toán tối ưu là giải phương trình đạo hàm (gradient) bằng 0. Nhưng chỉ trường hợp:

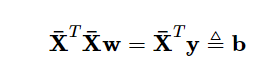
o Tính đạo hàm.

o Giải phương trình bằng 0.

* Với mô hình tuyến tính, hai việc này là khả thi.
* Đạo hàm theo w của hàm mất mát là:



Phương trình đạo hàm bằng 0 tương đương với:

(\*)

Đặt:





* ·Nếu ma trận A khả nghịch thì phương trình (\*) có nghiệm duy nhất là :



* Nếu A không khả nghịch (có định thức bằng 0) thì phương trình (\*) vô nghiệm hoặc vô số nghiệm, khi đó ta dùng giả nghịch đảo để giải bài toán.
* Với khái niệm giả nghịch đảo, điểm tối ưu của bài toán Linear Regression có dạng: 𝑊=𝐴−1𝑏

**PHẦN 2: THỰC NGHIỆM**

# **2.1. Mô tả đề tài:**

## 2.1.1 Tên đề tài

## Ứng dụng phương pháp hồi quy tuyến tính dự đoán giá xe Oto ở London

+Input: gồm

* SỐ CHỖ
* CHIỀU DÀI(inch)
* CHIỀU RỘNG (inch)
* CHIỀU CAO (inch)
* SỐ XILANH (int)
* KÍCH THƯỚC ĐỘNG CƠ (cc)
* TỈ LỆ NÉN (vc)
* MÃ LỰC (hp)

+Output: dự đoán GIÁ (dola) xe của mỗi Oto.

## 2.1.2 Thuật toán đề tài

* Nghiên cứu về phương pháp học máy có giám sát và cụ thể là thuật toán hồi quy tuyến tính với đề tài “Ứng dụng phương pháp hồi quy tuyến tính dự đoán giá xe Oto ở LonDon”.
* Hàm số cơ bản nhất mô tả mối quan hệ giữa giá nhà và 3 đại lượng đầu vào là:

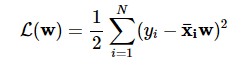
y ≈ f(x) = y’

f(x) = w1x1 + w2x2 + w3x3 + w0

* Với y là giá trị thực, y’ là giá trị dự đoán, w1,2,3 là hằng số,x1,2,3 là các giá trị của một điểm dữ liệu đầu vào, w0 là bias (kiểu hệ số tự do).
* Bài toán chúng ta trở thành đi tìm các hệ số tối ưu { w1, w2 ,w3 ,w0 }
* Với e là sự sai khác giữa giá trị thực y và giá trị dự đoán y’(sai số dự đoán), để thuận tiện cho việc tính toán, ta xét :

c

* Vớif và e là vector (hàng) dữ liệu đầu vào mở rộng. Số 1 ở đầu được thêm vào để phép tính đơn giản hơn và thuận tiện cho việc tính toán.
* Điều chúng ta muốn, tổng sai số là nhỏ nhất, tương đương với việc tìm w để hàm số sau đạt giá trị nhỏ nhất:

(Hàm mất mát)

* Sau các biến đổi đại số, ta tìm được w thỏa mãn yêu cầu là:

g

* Từ đó có thể xây dựng code dự đoán được giá bán xe Oto dựa trên các đặc điểm đã cho.

# **2.2.Mô tả tập dữ liệu của bài toán**

* Dữ liệu bài toán gồm:
* Ma trận x là vector đầu vào tập hợp các điểm dữ liệu, mối điểm dữ liệu mô tả giá của xe Oto(đơn vị:USD) là ma trận y (giá trị thực của đầu ra) gồm có 8 thông tin dữ liệu:
* SỐ CHỖ
* CHIỀU DÀI(inch)
* CHIỀU RỘNG (inch)
* CHIỀU CAO (inch)
* SỐ XILANH (int)
* KÍCH THƯỚC ĐỘNG CƠ (cc)
* TỈ LỆ NÉN (vc)
* MÃ LỰC (hp)
* Bài toán có 200 điểm dữ liệu.
* Dữ liệu bài toán

Dataset: <https://www.kaggle.com> .

# **2.3. Chia tập dữ liệu**

* Tập dữ liệu có 200 điểm dữ liệu chia: 140 điểm dữ liệu dùng để huấn luyện mô hình, 60 điểm dữ liệu dùng để kiểm tra sự phù hợp của mô hình.
* Dữ liệu tập huấn luyện (Training)
* Ma trận xtrain là vector đầu vào tập hợp các điểm dữ liệu, mối điểm dữ liệu mô tả giá của xe Oto là ma trận ytrain (giá trị thực của đầu ra) gồm có 8 thông tin dữ liệu: SỐ CHỖ, CHIỀU DÀI, CHIỀU RỘNG, CHIỀU CAO, SỐ XILANH, KÍCH THƯỚC ĐỘNG CƠ, TỈ LỆ NÉN, MÃ LỰC.
* Tập dữ liệu huấn luyện gồm 140 điểm dữ liệu.
* Dữ liệu tập kiểm tra (Testing)
* Ma trận xtest là vector đầu vào tập hợp các điểm dữ liệu, mối điểm dữ liệu mô tả giá của xe Oto là ma trận ytest (giá trị thực của đầu ra) gồm có 8 thông tin dữ liệu: SỐ CHỖ, CHIỀU DÀI, CHIỀU RỘNG, CHIỀU CAO, SỐ XILANH, KÍCH THƯỚC ĐỘNG CƠ, TỈ LỆ NÉN, MÃ LỰC.
* Tập dữ liệu huấn luyện gồm 60 điểm dữ liệu.

# **2.4. Mô tả ma trận dữ liệu huấn luyện (xtrain, ytrain) và ma trận dữ liệu kiểm tra (xtest, ytest)**

## 2.4.1 Tập dữ liệu huấn luyện(Training)

* Link: [data\_train\_oto](https://docs.google.com/spreadsheets/d/1gNEPsL5WdLOsnyPpoy_S7ZpZ2fZ69gz-Nazl1plmIbs/edit#gid=0)
* Demo 10 điểm dữ liệu trong 140 điểm dữ liệu huấn luyện

|  |  |
| --- | --- |
| Tập giá trị xtrain | Tập giá trị ytrain |
|  |  |

## 2.4.2 Tập dữ liệu kiểm tra(Testing)

* Link: [data\_test\_oto](https://docs.google.com/spreadsheets/d/1aPlLr142kz78oMuxX10GjhIRFwvxQUXcqGLS7md6DCw/edit#gid=0)
* Demo 10 điểm dữ liệu trong 60 điểm dữ liệu kiểm tra

|  |  |
| --- | --- |
| Tập giá trị xtest | Tập giá trị ytest |
|  |  |

# **2.5. Dùng phương pháp Hồi quy tuyến tính để xây dựng mô hình cho bài toán với tập dữ liệu huấn luyện.**

Demo chương trình python

**PHẦN 3: ĐÁNH GIÁ**

# **3.1 Mô tả kết quả dự đoán**

# Demo 20 kết quả dự đoán

Với xtest là [ 2. 176.8 66.6 50.5 4. 132. 8.7 90. ] thì kết quả dự đoán là 13434.52 kết quả thực tế là 13499.0

Sai lệch : 64.48

Với xtest là [ 2. 186.6 66.5 56.1 4. 121. 9.31 110. ] thì kết quả dự đoán là 15216.76 kết quả thực tế là 15850.0

Sai lệch : 633.24

Với xtest là [ 4. 186.6 66.5 56.1 4. 121. 9.3 110. ] thì kết quả dự đoán là 14363.78 kết quả thực tế là 14170.0

Sai lệch : 193.78

Với xtest là [ 2. 186.6 66.5 56.1 4. 121. 9.3 110. ] thì kết quả dự đoán là 15215.03 kết quả thực tế là 15040.0

Sai lệch : 175.03

Với xtest là [ 4. 186.6 66.5 56.1 4. 121. 9.3 110. ] thì kết quả dự đoán là 14363.78 kết quả thực tế là 15510.0

Sai lệch : 1146.22

Với xtest là [ 2. 186.6 66.5 56.1 4. 121. 9. 160. ] thì kết quả dự đoán là 18646.09 kết quả thực tế là 18150.0

Sai lệch : 496.09

Với xtest là [ 4. 186.6 66.5 56.1 4. 121. 9. 160. ] thì kết quả dự đoán là 17794.84 kết quả thực tế là 18620.0

Sai lệch : 825.16

Với xtest là [ 2. 156.9 63.4 53.7 4. 97. 9. 69. ] thì kết quả dự đoán là 6981.82 kết quả thực tế là 6518.0

Sai lệch : 463.82

Với xtest là [ 2. 157.9 63.6 53.7 4. 108. 8.7 73. ] thì kết quả dự đoán là 8414.45 kết quả thực tế là 8053.0

Sai lệch : 361.45

Với xtest là [ 2. 157.3 63.8 55.7 4. 108. 8.7 73. ] thì kết quả dự đoán là 8847.7 kết quả thực tế là 8603.0

Sai lệch : 244.7

Với xtest là [ 4. 172. 65.4 52.5 4. 108. 9.5 82. ] thì kết quả dự đoán là 9457.88 kết quả thực tế là 8926.0

Sai lệch : 531.88

Với xtest là [ 4. 172. 65.4 52.5 4. 108. 9.5 82. ] thì kết quả dự đoán là 9457.88 kết quả thực tế là 9175.0

Sai lệch : 282.88

Với xtest là [ 4. 172. 65.4 52.5 4. 108. 9. 94. ] thì kết quả dự đoán là 10207.12 kết quả thực tế là 9960.0

Sai lệch : 247.12

Với xtest là [ 4. 172. 65.4 54.3 4. 108. 9. 82. ] thì kết quả dự đoán là 9701.47 kết quả thực tế là 9233.0

Sai lệch : 468.47

Với xtest là [ 4. 172. 65.4 54.3 4. 108. 7.7 111. ] thì kết quả dự đoán là 11496.25 kết quả thực tế là 11259.0

Sai lệch : 237.25

Với xtest là [ 4. 173.5 65.4 53. 4. 108. 9. 82. ] thì kết quả dự đoán là 9520.84 kết quả thực tế là 8963.0

Sai lệch : 557.84

Với xtest là [ 4. 173.5 65.4 53. 4. 108. 9. 94. ] thì kết quả dự đoán là 10356.78 kết quả thực tế là 10198.0

Sai lệch : 158.78

Với xtest là [ 4. 173.6 65.4 54.9 4. 108. 9. 82. ] thì kết quả dự đoán là 9873.34 kết quả thực tế là 9513.0

Sai lệch : 360.34

Với xtest là [ 4. 173.6 65.4 54.9 4. 108. 7.7 111. ] thì kết quả dự đoán là 11668.11 kết quả thực tế là 11694.0

Sai lệch : 25.89

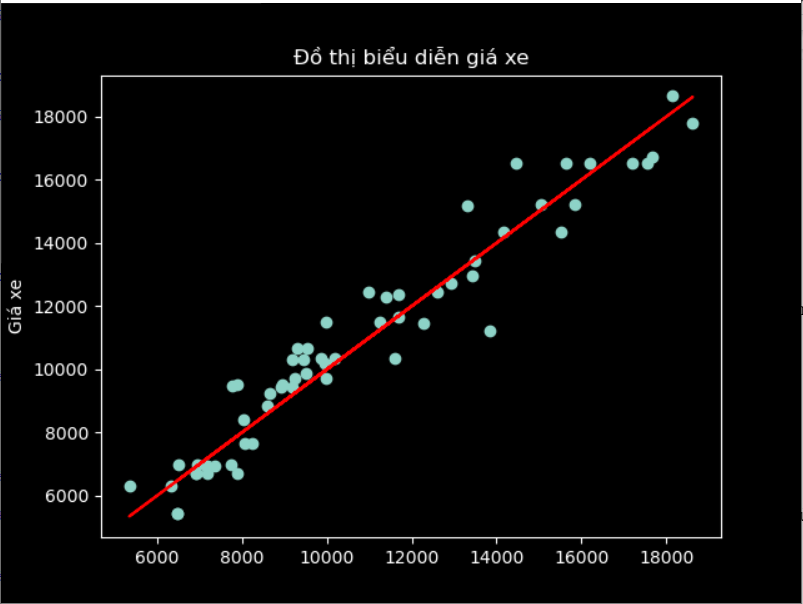
Với xtest là [ 2. 158.7 63.6 54.5 4. 92. 9. 62. ] thì kết quả dự đoán là 6309.92 kết quả thực tế là 5348.0

Sai lệch : 961.92

# 

# 

# **3.2 Mô tả biểu đồ đánh giá mô hình**



Chú thích: 

# **3.3 Đánh giá chất lượng mô hình**

* Từ mô tả kết quả từ tập dữ liệu kiểm tra ta thấy kết quả dự đoán của mô hình chênh lệch sai số không nhiều so với kết quả thực tế,
* Từ đồ thị trên ta thấy các giá trị dự đoán tập trung gần với giá trị thực tế có sự phân bổ đều.

=> Chất lượng mô hình được xây dựng từ tập dữ liệu huấn luyện(training) là phù hợp và hoạt động tốt.

**KẾT LUẬN**

Mục tiêu của hồi quy là xác định các giá trị của các tham số để tối thiểu hóa tổng các giá trị còn lại bình phương (dự đoán chênh lệch between và quan sát) cho tập hợp các quan sát. Do hồi quy tuyến tính bị hạn chế trong việc kết hợp các hàm tuyến tính (đường thẳng / mặt phẳng) với dữ liệu, nên nó không phù hợp với dữ liệu trong thế giới thực vì các kỹ thuật tổng quát hơn như mạng thần kinh có thể mô hình hóa các hàm phi tuyến tính. Nhưng hồi quy tuyến tính có một số lợi thế thú vị:

* Hồi quy tuyến tính là phương pháp được sử dụng rộng rãi nhất, và nó được hiểu rõ.
* Đào tạo một mô hình hồi quy tuyến tính thường nhanh hơn nhiều so với các phương pháp như mạng lưới thần kinh.
* Các mô hình hồi quy tuyến tính rất đơn giản và yêu cầu bộ nhớ tối thiểu để thực hiện, vì vậy chúng hoạt động tốt trên các bộ điều khiển nhúng có không gian bộ nhớ hạn chế.
* Bằng cách kiểm tra độ lớn và dấu hiệu của các hệ số hồi quy, bạn có thể suy ra cách biến dự báo ảnh hưởng đến kết quả mục tiêu.
* Đây là một trong những thuật toán đơn giản nhất và có sẵn trong một số gói, ngay cả Microsoft Excel.

Thu hoạch của nhóm: Trau dồi thêm nhiều kiến thức về môn học “Học máy”. Qua bài tập lớn, các thành viên đã phần nào hiểu được những khái niệm và những công việc cơ bản nhất để xây dựng một mô hình học máy dựa trên các thuật toán và áp dụng được những lý thuyết đã học vào trong đề tài thực tế.

Và chúng em xin được cảm ơn cô Nguyễn Thị Kim Ngân đã tận tình giảng dạy những kiến thức về machine learning. Trong một khoảng thời gian nghiên cứu và thực hiện đề tài, nhóm đã hoàn thành được đề tài: *Ứng dụng phương pháp hồi quy tuyến tính dự đoán giá xe Oto ở London.* Do kiến thức chúng em còn chưa thật vững chắc nên vẫn còn những khuyết điểm trong đề tài này. Mong cô thông cảm và góp ý cho nhóm để đề tài của chúng em được hoàn thiện hơn và rút ra kinh nghiệm để thức hiện những đề tài sau này.

**TÀI LIỆU THAM KHẢO**

* Tham khảo slide “Học máy” của cô TS.Nguyễn Thị Kim Ngân
* Tài liệu file: <https://github.com/tiepvupsu/ebookMLCB> của thầy Vũ Hữu Tiệp
* Dataset:[https://www.kaggle.com/](https://www.kaggle.com/goyalshalini93/car-price-prediction-linear-regression-rfe/data?fbclid=IwAR0x2sXrpPkU5PBKFhvSrWrJL344T-Kwyx5Qtunm_z4TOkW3xHsPJ7Slg3s)