# Bình thường xài font: **Times New roman**

Code xài font: **Fira code**

**Câu hỏi:** mỗi người có block riêng

**Phần trả lời (bao gồm luôn người hỏi):** ngay dưới câu hỏi phải thực hiện bằng cú pháp

**<Tên>**

**Câu trả lời**

|  |
| --- |
| **Thân** |

1. Neural Networks có phải là mạng ANN không?

ANN là 1 dạng bài toán hồi quy (input là số, output là 1 giá trị thực), khác với CNN (input là ảnh)? Hay ANN là 1 thuật ngữ chung để nói đến NN, CNN, RNN,...

Em nghĩ ANN là 1 bài toán ML đơn giản (mạng không sâu như DL). Với input, output đều là giá trị thực. VD: Bài dự đoán giá nhà là ANN

**<Tiềm>** Theo em Neural Network là ANN(Artificial Neural Network), ANN không chỉ là 1 dạng bài hồi quy, nó có thể xử lí các bài toán phi tuyến, một số thuật toán trong ANN như Perceptron, Hebb, LMS, Lan truyền ngược, Softmax Regression…..

**<Annie>**

ANN (Artificial Neural Network), hay còn được gọi tắt là Neutral Network (NN), là 1 computing networks gồm có 3 hoặc nhiều hơn 3 layers (input, hidden (black box) và output)

CNN (Convolutional Neutral Network) thuộc ANN, được dùng trong phân tích hình ảnh.

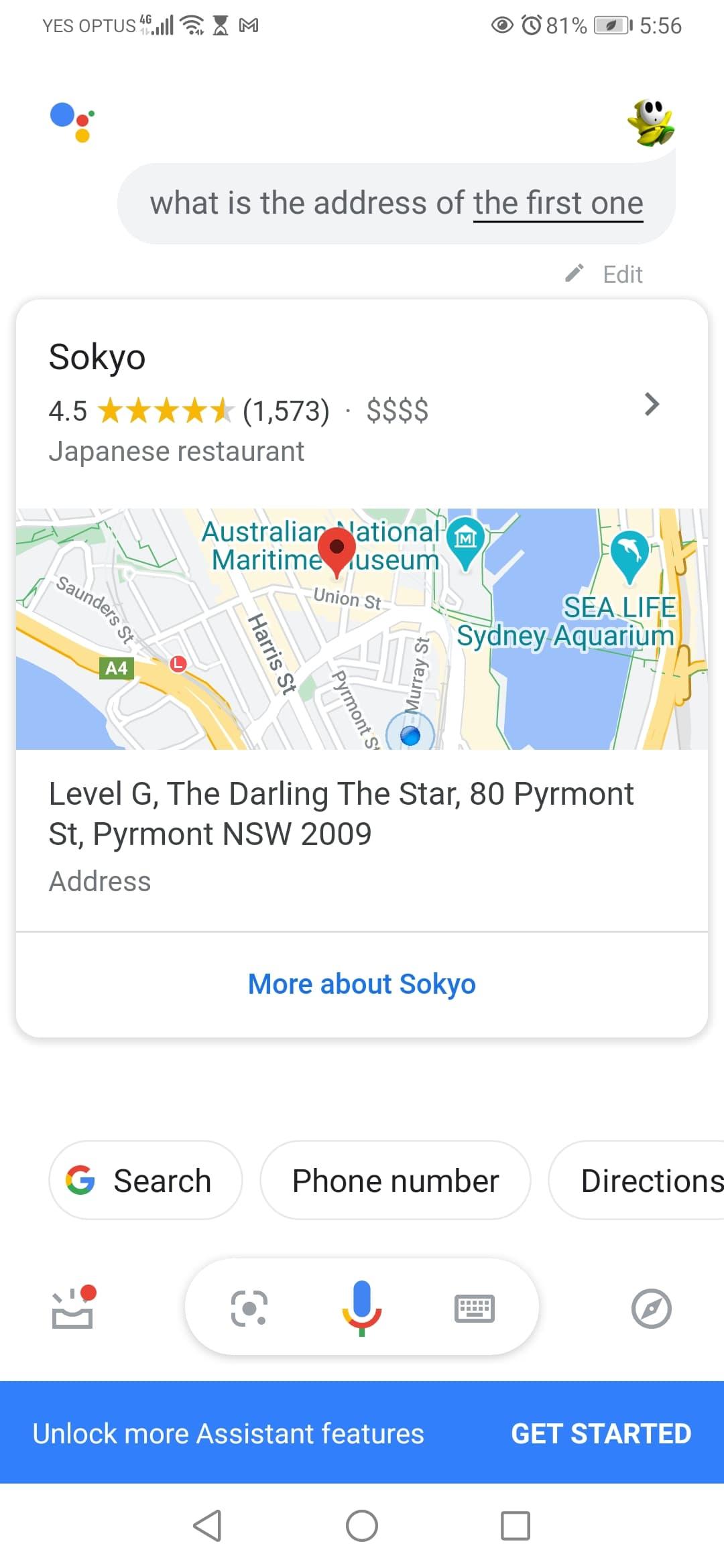
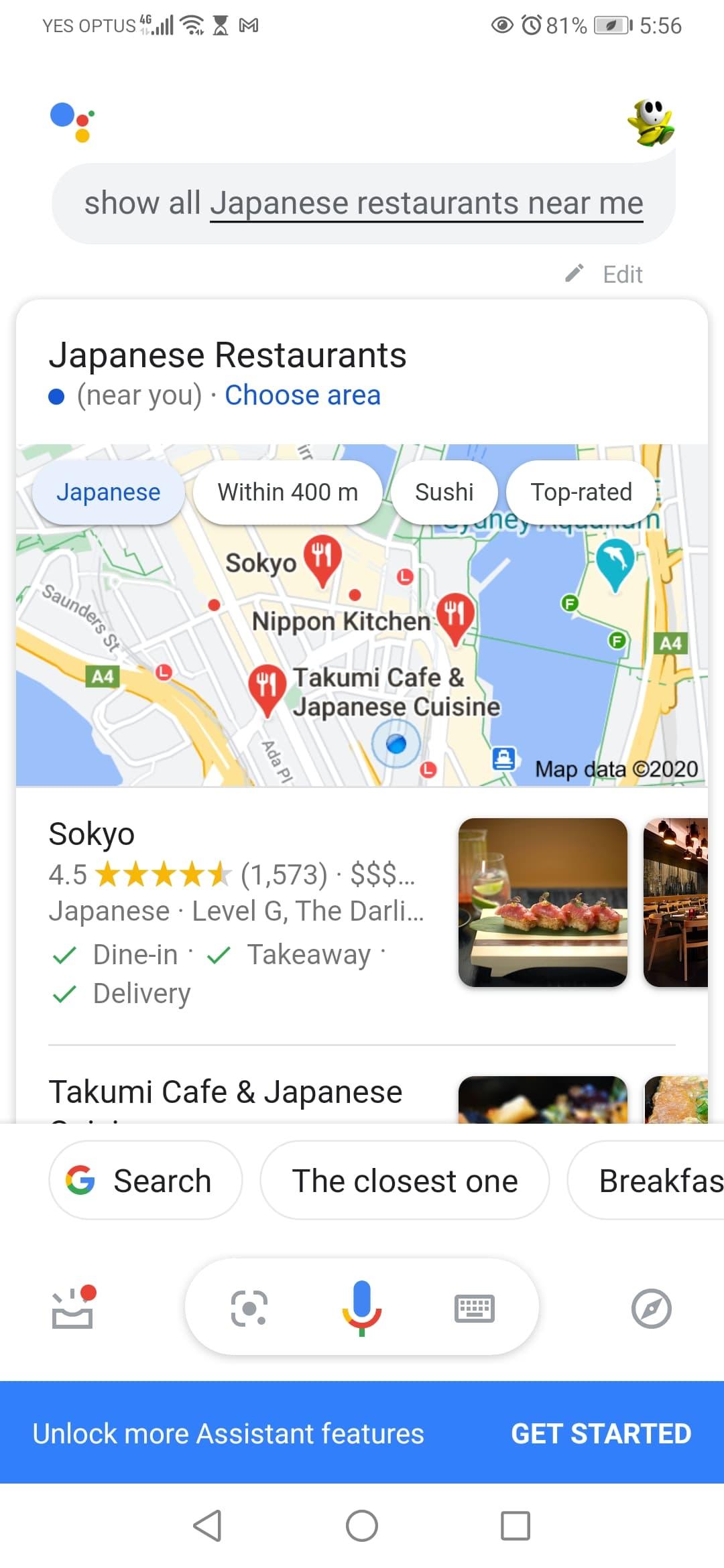
*"They are also known as shift invariant of space invariant artificial neutral networks, based on their shared-weights architexture and translation invariance characteristics"*

(Đoạn này search trên mạng mà không hiểu lắm, cũng không biết tiếng việt nói thế nào nên chắc nhờ anh Vinh giải thích thêm :D). CNN được dùng nhiều nhất trong phân tích hình ảnh và video.

RNN (Recurrent Neutral Network) thuộc ANN, được dùng cho sequential data (dữ liệu dãy?). Vì có khả năng nhớ 1 chuỗi các input (được lưu trữ trong bộ nhớ trong), RNN được dùng để giải quyết các bài toán về Time Series data, Question-Answering (Text data, Audio data). RNN còn được dùng trong các ứng dụng Siri của Apple và tìm kiếm bằng giọng nói của Google.

*(Reference:* [*https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/*](https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/)*)*

Vd: RNN ứng dụng trong Question-Answering



Ở đây Google nhớ câu trả lời của câu hỏi đầu tiên, và biết được “the first one” trong câu hỏi thứ hai chỉ đến *Sokyo*

1. Bài toán dự đoán giá nhà, bài này em hiểu là dùng ANN, vậy em dùng KNN được không?

Em nghĩ dùng cả ANN và KNN đều được: ANN thì model cho hàm ypre nên dự đoán chính xác hơn. KNN thì dự đoán dựa trên K phần tử kế bên.

**<Tiềm>**

Theo em thì bài dự đoán giá nhà mà Ad đang dạy là sử dụng thuật toán hồi quy tuyến tính trong machine learning, nếu sử dụng ANN thì mô hình mạng nó sẽ gồm: Input layer, hiden layer, output layer mà mô hình ad đang sử dụng là hồi quy tuyến tính.

**<Annie>**

Em giống ý Tiềm. Theo em thì ANN là unsupervised learning, trong khi đó KNN là supervised learning. Bài toán dự đoán giá nhà là bài toán có label nên nó là supervised learning.

Tuy nhiên bài toán này là thuộc dạng regression nên sẽ không dùng KNN (KNN được dùng cho bài toán classification, dùng label của những phần tử gần nhất để dự đoán label của phần tử cần tìm). Những bài toán regression có thể giải bằng Ensemble Regressors hoặc là Ridge Regession.

Anh có thể tham khảo ở đây để biết nên cho model nào để dùng cho các dạng bài toán supervised learning khác nhau: <https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html>

1. Bài toán dự đoán giá nhà, em không train model, mà dùng đa thức nội suy để xấp xỉ 1 hàm đa thức được không?

Dùng đa thức nội suy Lagrange sẽ dẫn đến overfitting. Chúng ta cần 1 model tổng quát, đa thức nội suy Lagrange chỉ fix trên tập train.

<Vinh> Theo kinh nghiệm anh từng đọc qua. thì bên linear algebra họ có cách giải close-form luôn! Nghĩa là có công thức để tìm giá trị cuối cùng luôn.  
  
https://stats.stackexchange.com/questions/336860/derivation-of-the-closed-form-solution-to-minimizing-the-least-squares-cost-func

Nhưng ở đây, mục đích mình học linear regression là để minh họa cách train và quá trình train một model. Nó là bước đệm cho cách phần sau, nên trong bài không đề cập đến cách giải close-form ở trên.

1. Bài toán dự đoán giá nhà:

Model: y = w1.x1 + b

Nếu em xấp xỉ 1 model có dạng y = w1.x1 + w2.(x1)^2 + b, như vậy model có còn là dạng linear không?

Model có dạng y = w1.x1 + w2.(x1)^2, model vẫn có dạng linear: Ở đây linear theo w (chứ không phải linear theo x), chúng ta cần tối ưu w với b. Với x1 là dữ liệu số thì x^2 vẫn là 1 hằng số

**<Nhi>**

Cái này em cũng hiểu nó theo kiểu này. Vì thực chất bản chất của model này vẫn là linear. Để dễ thấy hơn thì ta đặt x1=a, x1^2=b, lúc đó y trở thành y = a.w1+b.w2+c. Tương tự với model có dạng phưc tạp hơn xíu như y=w1.x1+w2.x1.x2,...thì bản chất nó vẫn là linear.

<Vinh> Hàm linear hay non-linear dựa vào mũ của x, nên có (x1)^2 thì không còn là hàm linear nữa rồi. Hàm linear hay non-linear là do mình thiết kế ban đầu và ngụ ý data sẽ tuân theo nó. Còn việc đi tìm các giá trị tham số quá trình tối ưu hóa, không liên quan gì đến kiểu model.

https://www.varsitytutors.com/hotmath/hotmath\_help/topics/quadratic-regression#:~:text=A%20quadratic%20regression%20is%20the,using%20the%20least%20squares%20method.

1. Hàm loss trong bài dự đoán giá nhà, mình dùng hàm loss L(w, b) = (ypre - ytrue)^4 được không?

L(w, b) = (ypre - ytrue)^n, với n chẵn là Ok hết. Vì nó đảm bảo, loss không âm và giảm. Nhưng n càng cao thì trong tính toán đạo hàm phức tạp hơn, trường hợp nếu sai số giữa ypre - ytrue lớn thì hàm loss cho kết quả cao.

**<Tiềm>**

Theo em người ta hay dùng các công thức đo khoảng cách như eclipse, mahattan, cosin…

<Vinh> Mình có thể dùng bất kì distance nào để làm hàm loss cũng được. Mình có thể dùng bất kì loại norm nào cũng được. Bậc 2 mình gọi là Euclidean distance.  
<https://www.maths.dundee.ac.uk/nasc/na-reports/NA221.pdf>

(optional) Nhi chuyên toán thử đọc hiểu bài trên. Sau đó, tìm hiểu (survey) các loại toán mà có thể làm distance được, rồi đánh giá performance của chúng cho bài toán linear regression và các model phía sau. Đây là một case study anh nghĩ cần nhiều công sức để đọc và chạy thực nghiệm. Nhi và vài bạn trong nhóm có thể làm chung.

1. Cơ sở để chọn trọng số Learning rate, nên dùng khoảng bao nhiêu là hợp lý?

Cái này ko có cơ sở, chúng ta phải thử. Kinh nghiệm thì khoảng 10^-4. Em nghe nói có thể dùng Gridsearch hoặc bayesian optimization để tìm hyperparameters của model (nhưng e ko biết cái này, Gridsearch kiểu như early stopping)

**<Annie>**

Giá trị learning rate được khuyên nên thử đầu tiên là 0.1 và 0.01. Và đối input đã được tiêu chuẩn hóa (standardized inputs) hoặc inputs được chuyển về nằm trong khoảng (0,1), giá trị learning rate tiêu chuẩn là nhỏ hơn 1 và lớn hơn 10^-6. Trong bài viết này cũng có nhắc đến dùng Diagnostic plot và Grid search để xem learning rate được chọn có tốt chưa (nhưng cụ thể làm như thế nào thì em cũng không biết)

<https://machinelearningmastery.com/learning-rate-for-deep-learning-neural-networks/>

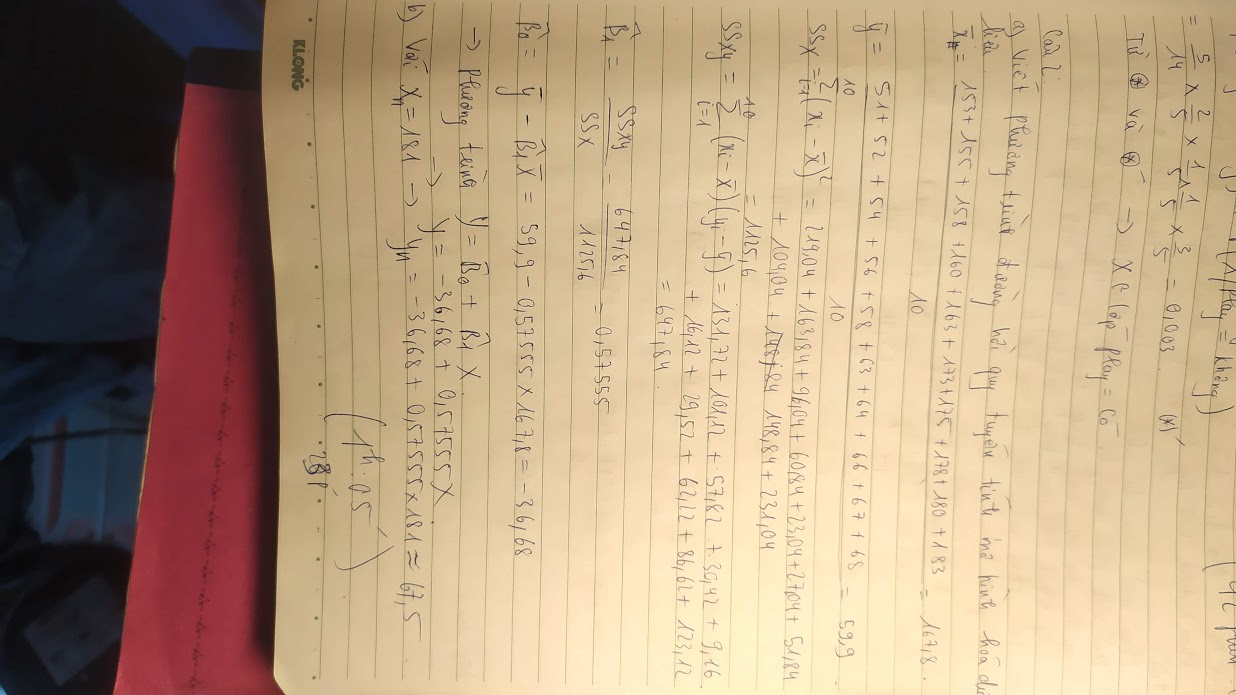
1. Bài toán dự đoán giá nhà, giá trị khởi tạo w, b có ảnh hưởng đến tính hội tụ của model không? Mình chọn ngẫu nhiên hay có cơ sở?

Cái này chắc là chọn ngẫu nhiên rồi, nếu đối với bài toán đơn giản có thể dự đoán được gần đúng thì càng tốt, khi đó model tối ưu nhanh hơn

**<Tiềm>**

Theo em bài này trọng số và b ban đầu mình tìm nó từ tập dữ liệu

Em có 1 bài sử dụng hồi quy tuyến tính sau



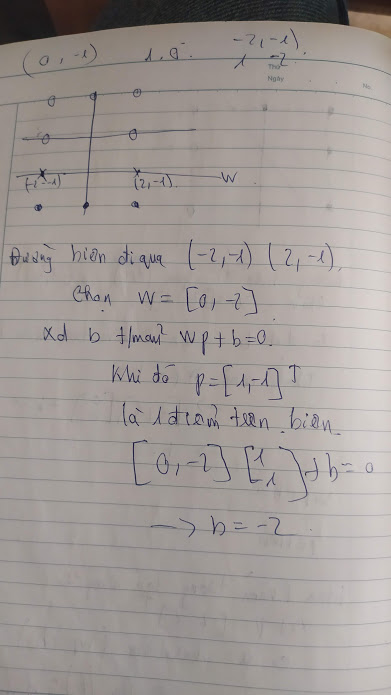
**<Nhi>**

Tiềm ơi, ảnh mờ quá không thấy gì nơi em ơi!>

**<Tiềm>** Chị mở app Tài liệu xem rõ hơn đấy chị>

Bài này bước đầu em tìm w và b hoàn toàn từ dữ liệu ban đầu

Trong mạng Nơ ron nhân tạo, người ta thường chọn w là 1 vecto vuông góc với đường biên giả sử em cần phân lớp bài toán như sau



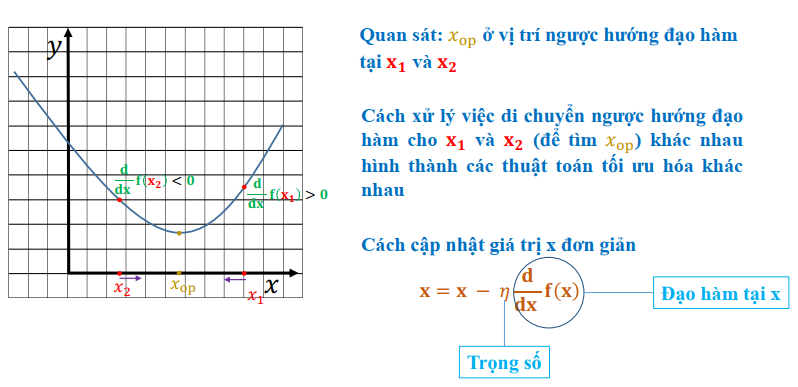
Một số bài toán phức tạp hơn em nghĩ người ta sẽ chọn theo kinh nghiệm hoặc có một số phương pháp tìm siêu tham số mô hình bằng thuật toán di truyền….

<Vinh> Chắc chắn giá trị khởi tạo sẽ ảnh hưởng đến quá trình training. Ở bài này do số biến ít nên chúng ta không cảm nhận rõ ràng. Sau này khi học tới CNN với trả triệu biến thì việc tạo giá trị ban đầu không tốt có thể làm cho model không train được luôn.

1. Tại sao trong công thức cập nhập lại w và b thì **giá trị đạo hàm tại điểm x** được dùng để cập nhập lại w và b chứ không phải giá trị khác?

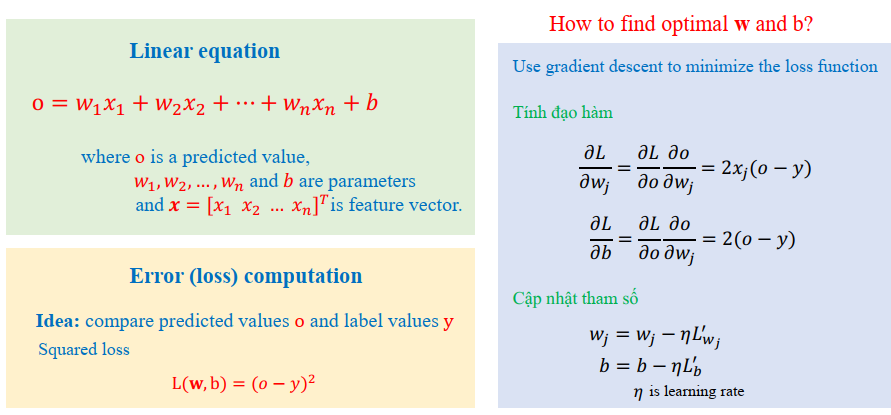
Em nghĩ có sự nhầm ở chỗ này, hok biết mình giải thích đúng hok:

a/ Đây là mô tả thuật toán GD, lấy hàm số y = f(x) (ở đây x là tham số cần tối ưu) để minh họa:



b/ Áp dụng thuật toán GD cho bài dự đoán giá nhà, lúc này là hàm loss J(w,b), chúng ta đạo hàm riêng theo từng biến wi và b để tối ưu hàm loss, như hình

dưới:



**<Vinh>** Anh đã update lại trong slide rồi. Lát anh gửi slide bản final nhé.

1. Em đọc thấy mấy bài phân loại dùng cả hàm loss và accuracy để đánh giá model. Bài dự đoán giá nhà em không dùng loss để đánh giá model, mà dùng accuracy có được không?

Bài này như hôm qua anh có giải thích: Với bài toán regression: loss giảm thì accuracy tăng và ngược lại. Còn bài toán phân loại thì chưa chắc nên dùng cả 2 (và có thể dùng nhiều thứ khác nữa, vd như F-score chẳng hạn). Nhưng em ko biết cơ sở toán học.

<Vinh> Giá trị accuracy (%) được tạo ra để cho chúng ta (humans) biết được model nào tốt hơn, chứ accuracy ko dùng làm loss được.   
<https://kharshit.github.io/blog/2018/12/07/loss-vs-accuracy>  
https://datascience.stackexchange.com/questions/42599/what-is-the-relationship-between-the-accuracy-and-the-loss-in-deep-learning

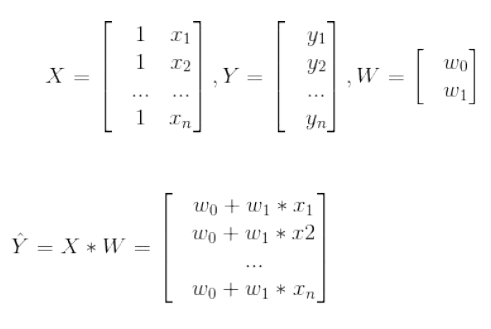
1. What about (y-o)^2

(y-o)^2 = (o-y)^2 nó là mean square error, em hiểu như bình thường :). Một số tài liệu thêm ½ ở trước để phép tính đạo hàm cho đẹp, hoàn toàn ko ảnh hưởng đến kết quả bài toán.

Mục đích đưa hàm loss về dạng (y-o)^2 là vì hàm này khi được biểu diễn trên đồ thị có dạng parapol và giúp cho việc tìm min của hàm loss trở nên dễ dàng hơn

1. What about x = [b, w]

x = [b, w], chỗ này em không hiểu. Nếu biểu diễn theo vector như bên dưới thì em hiểu, X, Y là dữ liệu input, mình cần tối ưu W

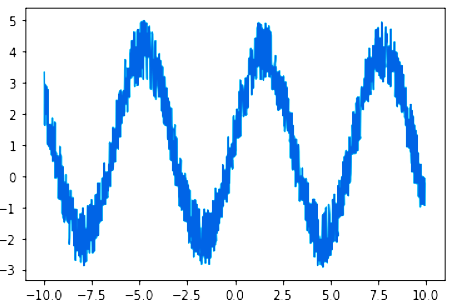


<Vinh> Ở trên anh đặt câu hỏi sai rồi. Nó nên là theta=[b,w]

Trong code hiện tại anh đang dùng theta=[w,b], việc dùng ngược lại thì đổi code một tí thôi.

|  |
| --- |
| **Quang** |

1. Giải 2 bài toán sau bằng Linear Regression



**<Quang>**

Gợi ý:

với:

data & code: [Linear\_sin](https://colab.research.google.com/drive/1FCZw1w1KJLyuxTGRrDQRnrQr1ppMtudu?usp=sharing)

Code tạo data: [code](https://colab.research.google.com/drive/1ZdqhXXzLSHJ7glBuKTmb9QfFsW4N3FrR?usp=sharing)

1. Cho data: [Moore.csv](https://drive.google.com/file/d/1r65kNy2-a1rtOtiNc8wFmLEM9NSWuTP4/view?usp=sharing)

Phân tích data và đem về dạng linear.

**<Quang>**

code: [Linear Moore](https://colab.research.google.com/drive/18bjSekirW7pbaTUO1N-3S2gQlh4KC7Vi?usp=sharing)

**<Vinh>** Hay đó Quang. Cái này có thể làm bài tập cho lớp làm thêm. [Thumbs up].

**<Thân>** Anh giải thích về mặt toán học, anh sửa lại phép biến đổi f(x) xíu

với:

Sau khi mình tối ưu A, B, C. Rồi chúng ta sẽ tiếp tục đi giải hệ pt để tìm w0, w1, w2.

**<Quang>** Cảm ơn anh đã bổ sung ạ

|  |
| --- |
| **Tiềm** |

1. Em chưa hiểu 1 sample và n sample khác nhau như thế nào ạ? Sample có phải là testing data không ạ?

**<Vinh>** Dữ liệu dùng khi 1 lần train nhằm tính giá trị đạo hàm cho mỗi weight. Nếu tập training có noise (vài sample không label chuẩn chẳng hạn) thì dùng 1 sample giá trị loss sẽ zigzag (không giảm đều và mượt). Khi trung n sample và giá trị trung bình của các giá trị đạo hàm của các sample sẽ cho làm cho quá trình tối ưu hóa chạy ổn định hơn.  
  
Cái này giống tính chất trong statistics, khi có nhiều sample và tính trung bình sẽ giúp giảm ảnh hưởng của các sample noise.

**<Tiềm>** Dạ em hiểu rồi ạ, em cảm ơn anh :))

1. Em xin góp ý về phần lí thuyết ạ(Chỉ là ý kiến cá nhân của em, ad có thể lưu ý ạ :))):

Khái niệm học máy Ad có thể dùng khái niệm của ông Mitchell không? Em thấy khá đầy đủ, mình có thể cho ví dụ để mọi người xác định T,E,P

Phần tiếp theo em nghĩ nên phân loại học máy thành 2 loại theo chức năng và theo phương thức học, Phương thức học: Học có giám sát, không giám sát, bán giám sát, tăng cường + ví dụ. Theo chức năng thì là cái sơ đồ cây của ad.

<Vinh> Ok Tiềm. Anh sẽ bổ sung phần này

<Trang 4> Em nghĩ ad nên thêm thông tin về chia dữ liệu thành 3 phần train, test, validation và 1 số phương pháp chia tập train và validation như cross-validation, kinh nghiệm của ad.

**<Vinh>** Mấy demo vừa rồi dùng data đơn giản. Anh sẽ làm cho bài có data nhiều hơn và chia ra cho đúng chuẩn.

Ad có thể đề cập thêm về hiện tượng overfitting và 1 số phương pháp tránh overfitting

**<Vinh>** Overfitting thường xảy ra khi chúng ta dùng model lớn như multilayer perceptron. Đến lesson 5-6 anh có nói 1 chương chuyên về cái này luôn (trong bài model generalization). Khi chúng ta xây dựng model, đầu tiền chúng ta cố gắng model đạt đến điểm overfitting, rồi mới giải quyết overfitting; chứ mình không thiết kế model để tránh overfitting ngay từ đầu.

Phần thuật toán em nghĩ nên có tổng kết sự khác nhau giữa 1 sample và nhiều sample ạ :))

**<Vinh>** Ok Tiềm. Anh sẽ làm thêm 1 slide về cái này.

|  |
| --- |
| **Nhi** |

|  |
| --- |
| **Annie** |

|  |
| --- |
| **Sơn** |

|  |
| --- |
| **Tuấn Anh** |