Contents

[1 Tìm hiểu về đạo văn 5](#_Toc59627690)

[1.1 Lời cam kết 5](#_Toc59627691)

[2 Giới thiệu thư viện SKLEARN 6](#_Toc59627692)

[2.1 Giới thiệu 6](#_Toc59627693)

[2.2 Tại sao sử dụng sklearn 6](#_Toc59627694)

[2.3 Cài đặt sklearn 6](#_Toc59627695)

[2.4 Ưu điểm của thư viện 6](#_Toc59627696)

[3 Machine Learning 7](#_Toc59627697)

[3.1 Đôi nét về Machine Learning 7](#_Toc59627698)

[3.1.1 Machine Learning là gì ? 7](#_Toc59627699)

[3.1.2 Tại sao sử dụng Machine Learning 7](#_Toc59627700)

[3.1.3 Phân Loại 8](#_Toc59627701)

[3.1.4 Những thách thức trong Machine Learning 14](#_Toc59627702)

[3.2 Testing and Validating 16](#_Toc59627703)

[3.2.1 Testing 16](#_Toc59627704)

[3.2.2 Validating 17](#_Toc59627705)

# Tìm hiểu về đạo văn

Đạo văn không chỉ đơn thuần là việc copy thành quả của một ai đó hay vay mượn các ý tưởng gốc, mà còn rất nhiều biến hóa khác của đạo văn. Tại các quốc gia phát triển họ coi ý tưởng hay sản phẩm trí thức là một tài sản được Pháp luật bảo vệ nên có thể nói đạo văn là một " **tội** ".

Để hiểu rõ về đạo văn một cách rõ ràng hơn ta sẽ tìm hiểu những hình thức đạo văn khác nhau :

“The Ghost Writer”: Sử dụng toàn bộ công trình của người khác thành của mình

“The Photocopy”: Sao chép bố cục của các đoạn văn từ một nguồn duy nhất, không hề sửa đổi lại.

“The Potluck Paper”: Cố gắng “trá hình” việc đạo văn của mình bằng cách sao chép từ nhiều nguồn khác nhau, biên tập lại các câu sao cho nội dung thật hợp lí.

“The Poor Disguise”: Giữ lại các nội dung quan trọng của nguồn, nhưng người đó vẫn sửa lại một chút về “diện mạo” của bài viết đó bằng cách thay đổi từ khóa hay câu cú.

“The Labor of Laziness”: Chú giải các nguồn khác nhau và nối chúng lại với nhau, thay vì dành nỗ lực tương tự cho công việc của mình.

“The Self-Stealer”: Sao chép đáng kể các thành quả trước đó của chính mình để phục vụ cho bài viết/nghiên cứu mới.

Vậy để tránh việc đạo văn ta có những cách sau :

Tìm hiểu kỹ về vấn đề mà bạn đang muốn nói tới

Diễn đạt lại nhiều lần bằng các cách khác nhau.

Nắm được những gì không cần trích dẫn.

Nêu tên tác giả của ý tưởng đó.

Trích dẫn nguồn bất cứ khi nào sử dụng lời trích, chú giải một cách chi tiết và cụ thể.

## Lời cam kết

“Chúng tôi xin cam đoan đồ án này do chính chúng tôi thực hiện. Chúng tôi không sao chép, sử dụng bất kỳ tài liệu, mã nguồn của người khác mà không ghi rõ nguồn gốc. Chúng tôi xin chịu hoàn toàn trách nhiệm nếu vi phạm.”

# Giới thiệu thư viện SKLEARN

## Giới thiệu

Scikit-learn (sklearn) là thư viện mã nguồn mở về machine learning phổ biến nhất trong cộng đồng Python

Sklearn chứa các thuật toán machine learning từ cơ bản cho đến phức tạp , ta chỉ cần khai báo vừa đưa dữ liệu và train thì sẽ có kết quả của thuật toán không cần cài đặt thuật toán một cách thủ công.

## Tại sao sử dụng sklearn

Độ chính xác thuật toán của Sklearn đem lại.

Sklearn hỗ trợ các thuật toán và doc của sklearn đầy đủ dễ hiểu để ta có thể sử dụng các thuật toán một các dễ dàng ngắn ngọn hiểu quả cao.

Scikit-learn còn là một trong những lựa chọn hàng đầu của các researchers và deverlopers.

Trang chủ của thư viện: <http://scikit-learn.org/>

## Cài đặt sklearn

Để phục vụ trong quá trình học hoặc ta làm việc vs sklearn ta cần cài những thư viện liên quan và đặc biệt là ngôn ngữ sử dụng là Python.

B1: Trước tiên máy các bạn phải được cài đặt Python ( phiên bản mình đang sử dụng là Python 3.7 ) nếu máy tính các bạn đã có Python có thể bỏ qua bước này.

B2: Để cài đủ thư viện sklearn ta cần sử dụng lệnh này:

**$ pip install numpy scipy matplotlib scikit-learn pandas**

## Ưu điểm của thư viện

Hàm ngắn ngọn, khai báo dễ dàng, dễ sử dụng.

Xử lý được lượng dữ liệu lớn.

Độ chính xác và độ tin cậy cao do cộng đồng sử dụng nhiều.

Tốc độ tính toán nhanh.

Hỗ trợ nhiều các thuật toán Machine Learning đơn giản và phức tạp.

# Machine Learning

## Đôi nét về Machine Learning

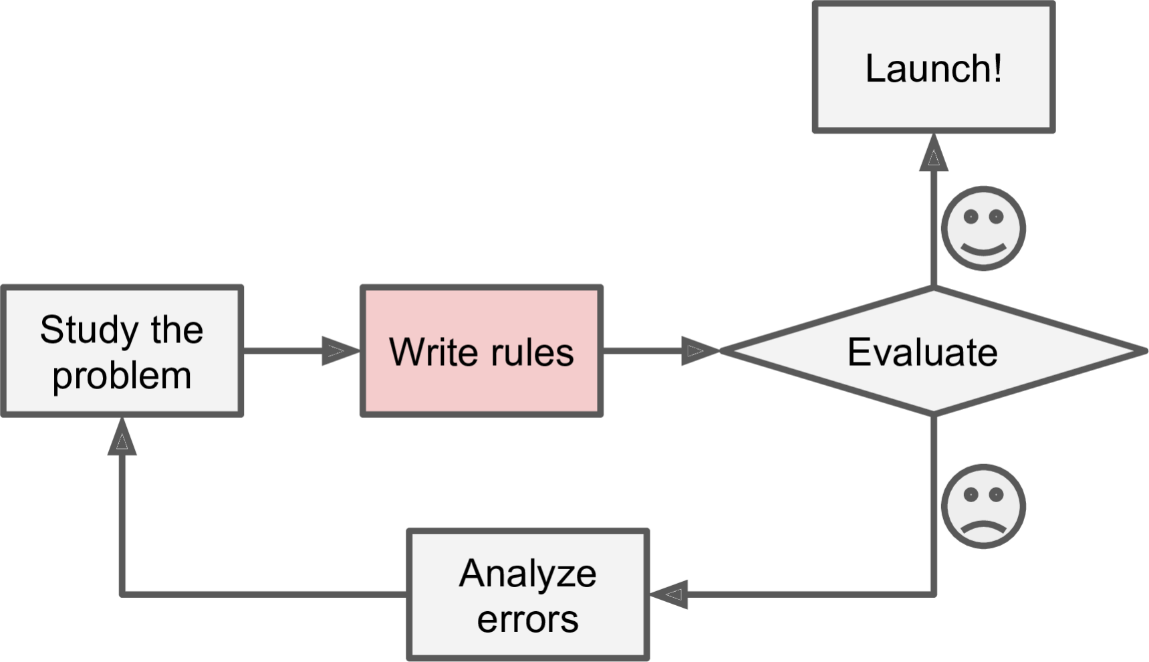
### Machine Learning là gì ?

#### Khái niệm

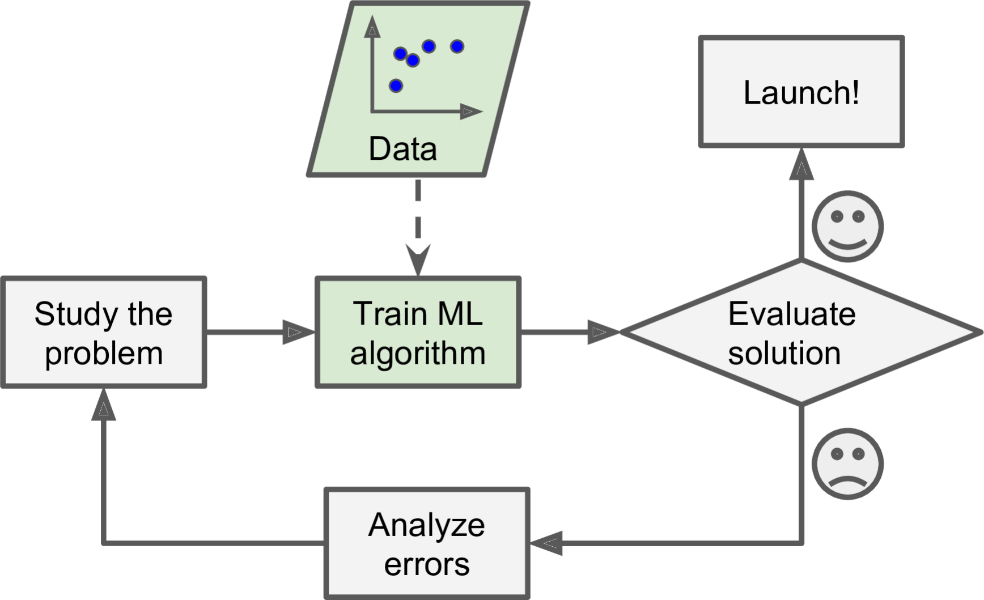
Theo Arthur Samuel đưa ra năm 1959 Maching learning là một ngành học thuộc khoa học máy tính, giúp máy tính có khả năng tự học mà không phải lập trình một cách rõ ràng

Theo Tom Mitchell Tom Mitchell coi Machine Learning như 1 chương trình, nhiệm vụ của nó là thưc hiện 1 task T nào đó, khi thực hiện xong, ta thu được experience E. Nhờ vào việc học hỏi experience E, ta có thể thay đổi (hoặc không) để tiến tới thực hiện task T+1, và nhằm cải thiện hiệu suất P

### Tại sao sử dụng Machine Learning

**Lập trình truyền thống**

**Lập trình có Machine Learning**



Lập trình truyền thống: Mình cần phải Tự học rất các vấn đề và viết ra các luật cho máy tính khi chạy chương trình mình có thể sẽ sinh ra lỗi vì mình sẽ thiếu một số trường hợp máy tính không xử lí được và mình tiếp tục tự học vấn đề và thử tiếp. Còn lập trình có xử dụng Machine Learning thì mình sẽ đưa ra vấn đề cho máy tính mà máy tính tự train cho máy tính và xử lý đưa ra kết quả.

Machine Learning có sự gắn bó chặt chẽ với khá nhiều ngành khác:

Big Data, AI, Statistics Learnig

Ứng dụng sâu rộng vào cuộc sống hàng ngày: trí tuệ nhân tạo AlphaGo, nhận diện khuôn mặt, gợi ý bạn bè từ faceboook, phân loại spam email từ google mail, chuẩn đoán y khoa, phát hiện thẻ tín dụng giả, phân tích thị trường chứng khoán, dự đoán kết quả trận đấu, nhận dạng giọng nói, phân loại các chuẩn DNA

### Phân Loại

Machine learning Algorithm được chia làm 2 loại chính là: Supervised Learning (Học có giám sát) và Unsupervised Learning (Học không giám sát).

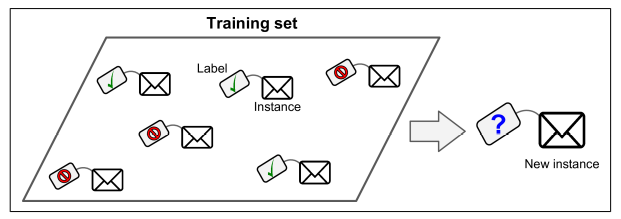
Ngoài ra còn 1 vài loại khác như SemiSupervised Learning, Reinforcement Learning, Learning to Learn, Developmental Learning, ...

#### Supervised Learning (Học có giám sát)

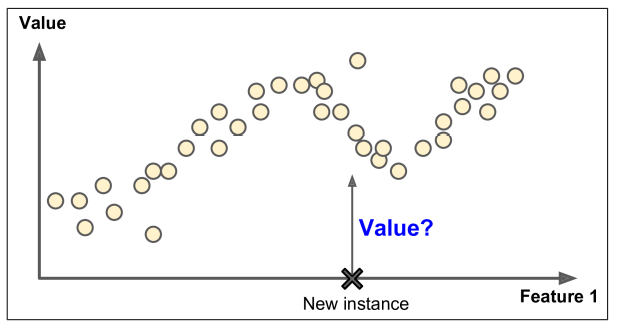
Supervised learning là thuật toán dự đoán đầu ra (outcome) của một dữ liệu mới (new input) dựa trên các cặp (input, outcome) đã biết từ trước.

Thuật toán supervised learning còn được tiếp tục chia nhỏ ra thành hai loại chính:

**Classification (Phân loại)**

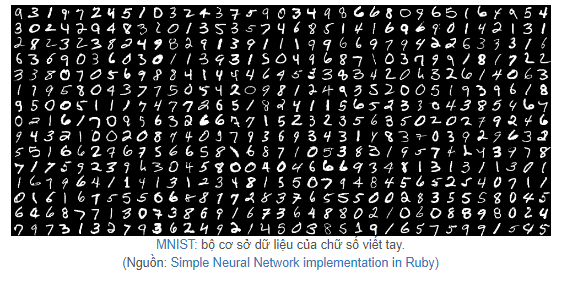
Một bài toán được gọi là classification nếu các label của input data được chia thành một số hữu hạn nhóm. Ví dụ: Gmail xác định xem một email có phải là spam hay không; các hãng tín dụng xác định xem một khách hàng có khả năng thanh toán nợ hay không

**Regression (Hồi quy)**

Nếu label không được chia thành các nhóm mà là một giá trị thực cụ thể. Ví dụ: một căn nhà rộng x m2x m2, có yy phòng ngủ và cách trung tâm thành phố z kmz km sẽ có giá là bao nhiêu?

Gần đây [**Microsoft có một ứng dụng dự đoán giới tính và tuổi dựa trên khuôn mặt**](https://how-old.net/). Phần dự đoán giới tính có thể coi là thuật toán **Classification**, phần dự đoán tuổi có thể coi là thuật toán **Regression**.

**Chú ý rằng phần dự đoán tuổi cũng có thể coi là Classification nếu ta coi tuổi là một số nguyên dương không lớn hơn 150, chúng ta sẽ có 150 class (lớp) khác nhau.**

VD: Trong nhận dạng chữ viết tay, ta có ảnh của hàng nghìn ví dụ của mỗi chữ số được viết bởi nhiều người khác nhau. Chúng ta đưa các bức ảnh này vào trong một thuật toán và chỉ cho nó biết mỗi bức ảnh tương ứng với chữ số nào. Sau khi thuật toán tạo ra một mô hình, tức một hàm số mà đầu vào là một bức ảnh và đầu ra là một chữ số, khi nhận được một bức ảnh mới mà mô hình **chưa nhìn thấy bao giờ**, nó sẽ dự đoán bức ảnh đó chứa chữ số nào.

Một số thuật toán phổ biến trong Supervised Learning

k-Nearest Neighbors

Linear Regression

Logistic Regression

Support Vector Machines (SVMs)

Decision Trees and Random Forests

#### Unsupervised Learning (Học không giám sát)

Trong thuật toán này, chúng ta không biết được outcome hay nhãn mà chỉ có dữ liệu đầu vào. Thuật toán unsupervised learning sẽ dựa vào cấu trúc của dữ liệu để thực hiện một công việc nào đó, ví dụ như phân nhóm (clustering) hoặc giảm số chiều của dữ liệu (dimension reduction) để thuận tiện trong việc lưu trữ và tính toán.

Một cách toán học, Unsupervised learning là khi chúng ta chỉ có dữ liệu vào XX mà không biết nhãn YY tương ứng.

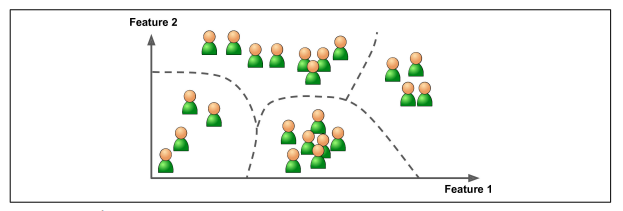
Những thuật toán loại này được gọi là Unsupervised learning vì không giống như Supervised learning, chúng ta không biết câu trả lời chính xác cho mỗi dữ liệu đầu vào.

Unsupervised learning được tiếp tục chia nhỏ thành hai loại:

Clustering (phân nhóm)

Một bài toán phân nhóm toàn bộ dữ liệu X thành các nhóm nhỏ dựa trên sự liên quan giữa các dữ liệu trong mỗi nhóm

VD: tam giác, vuông, tròn với màu xanh và đỏ, sau đó yêu cầu trẻ phân chúng thành từng nhóm. Mặc dù không cho trẻ biết mảnh nào tương ứng với hình nào hoặc màu nào, nhiều khả năng chúng vẫn có thể phân loại các mảnh ghép theo màu hoặc hình dạng.



Association

Là bài toán khi chúng ta muốn khám phá ra một quy luật dựa trên nhiều dữ liệu cho trước.

Ví dụ: những khách hàng nam mua quần áo thường có xu hướng mua thêm đồng hồ hoặc thắt lưng; những khán giả xem phim Spider Man thường có xu hướng xem thêm phim Bat Man, dựa vào đó tạo ra một hệ thống gợi ý khách hàng (Recommendation System), thúc đẩy nhu cầu mua sắm

#### Semisupervised learning (Học bán giám sát )

Các bài toán khi chúng ta có một lượng lớn dữ liệu XX nhưng chỉ một phần trong chúng được gán nhãn được gọi là Semi-Supervised Learning. Những bài toán thuộc nhóm này nằm giữa hai nhóm được nêu bên trên.

#### Online learning

Online learning là: Hệ thống có khả năng học hỏi tưng bước bằng cách cho nó dữ liệu tuần tự. Mỗi bước học nhanh chóng và hệ thống có thể tìm hiểu về dữ liệu mới. Máy sẽ học rất nhanh những dữ liệu đó và giải phóng dữ liệu để tránh tiêu tốn tài nguyên. Thích hợp với những hệ thống có nguồn tài nguyên hạn chế và dữ liệu liên tục cập nhật và đổi mới.

VD: Chương trình đánh giá “Giá cổ phiếu”.

#### Batch learning

Batch learning là: Hệ thống không có khả năng học hỏi từng bước. Đưa dữ liệu đầu vào (data training ) máy sẽ học tất cả dữ liệu và được vận dụng giải quyết các bài toán theo dữ liệu có sẵn. Muốn xử lý dữ liệu mới thì máy tính học lại tập dữ liệu cũ và thêm dữ liệu mới sau đó vận hành lại, quá trình này sẽ mất rất nhiều thời gian và tài nguyên máy tính.

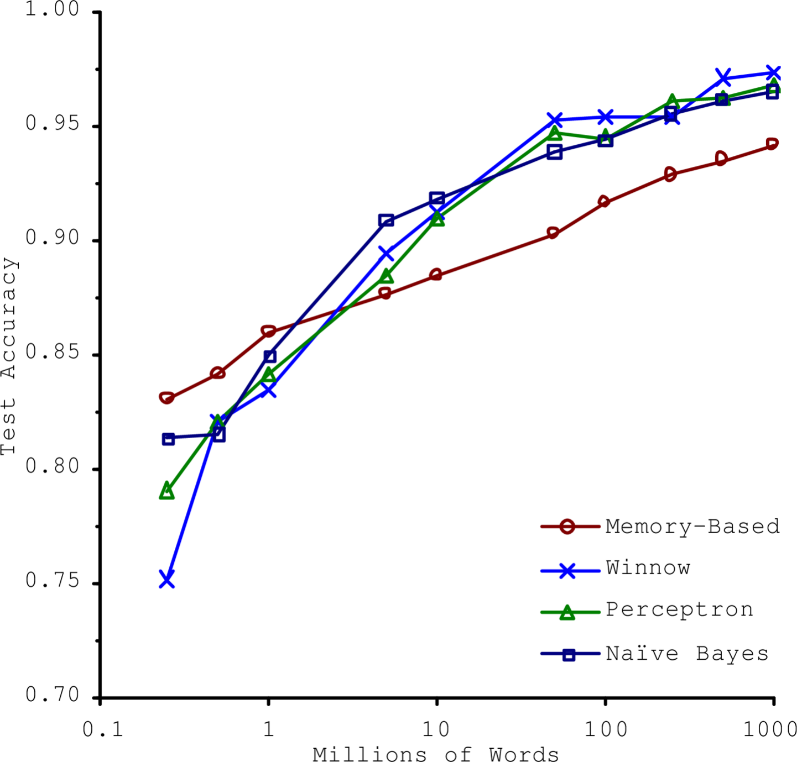
VD: Trò chơi game cờ vua

#### Instance-based learning

#### Model-based learning

### Những thách thức trong Machine Learning

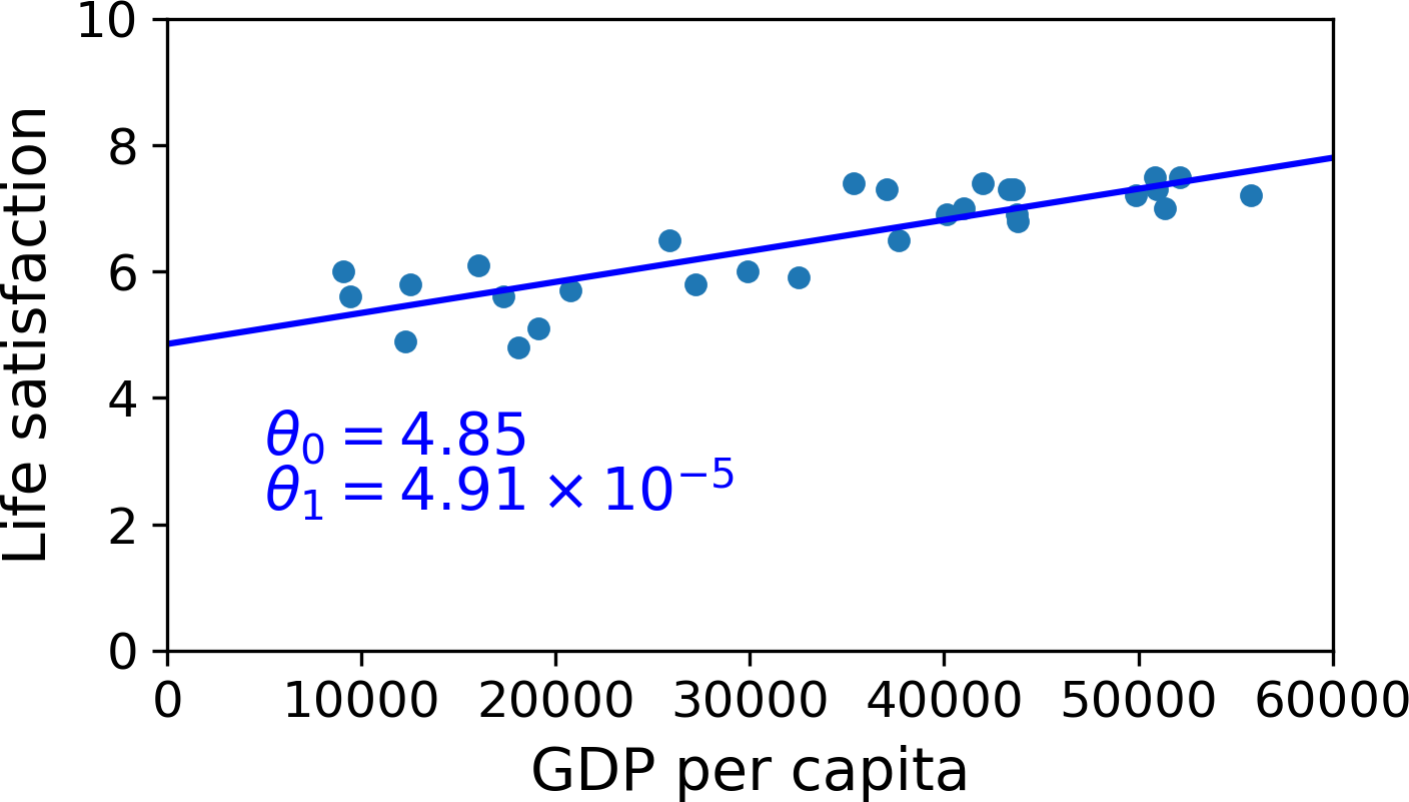
#### Bad Data

**Insufficient Quantity of Training Data**

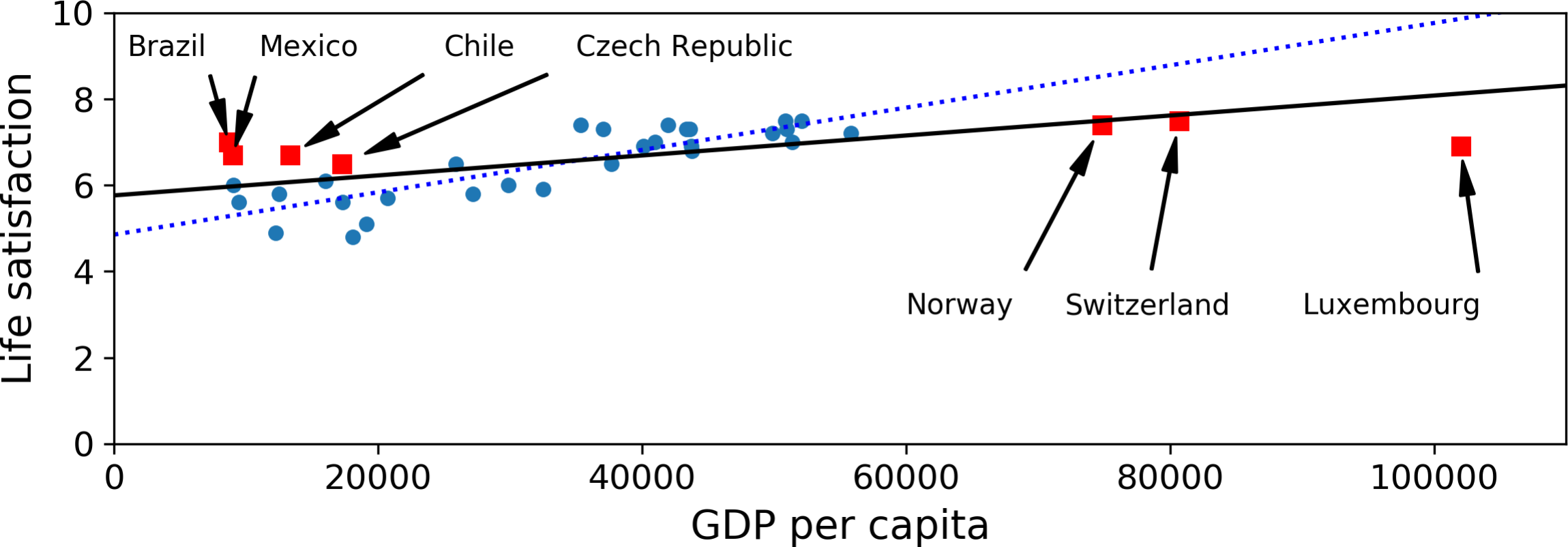
Thu thập training data quá ít không đủ để cho máy học và sẽ có một số trường hợp hay một dữ liệu nào đó mà máy sẽ đánh giá và đưa ra kết quả không được tốt hoặc có thể không xử lý được

**Nonrepresentative Training Data**

Dữ liệu không bị nhiễu



Dữ liệu bị nhiễu dư thừa



Muốn thu thập dữ liệu về một số người mắc covid mà ta chỉ thu thập dữ liệu số người mắc bệnh ở các quốc gia như Trung Quốc, Mỹ thì máy sẽ không thể đánh giá mức độ mắc bệnh của một quốc gia khác như Việt Nam.

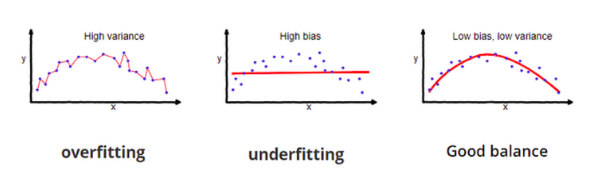
**Missing Data**

Muốn thu thập dữ liệu thống kê giá nhà ở Việt Nam. Nhưng chúng ta thu thập lại thiếu các cột dữ liệu như (số phòng, độ an ninh khu vực, …) thì máy sẽ không thể đưa ra đánh giá chính xác về giá nhà ở khu An Ninh hay nhà có nhiều phòng.

**Irrelevant Features**

Thu thập nhiều thông tin không cần thiết. Muốn thu thập dữ liệu chiều cao mà thu thập them các dữ liệu về nơi ở, tôn giáo, dân tộc, …

#### Bad algorithms



**Overfitting**

Overfitting là mô hình quá khít với dữ liệu, nó sẽ đúng trên tập training nhưng trên tập test (kiểm thử ) thì kết quả rất tệ. Mô hình này thường có bias nhỏ và variance lớn là mô hình quá khít với dữ liệu, nó sẽ đúng trên tập training nhưng trên tập test (kiểm thử ) thì kết quả rất tệ. Mô hình này thường có bias nhỏ và variance lớn.

Giải pháp: Thu thập thêm dữ liệu để tăng độ đa dạng, phong phú của dữ liệu. Dừng train model phù hợp để tránh overfitting để đạt được model tốt nhất (sẽ được đề cập các cách tránh overfitting trong các phần sau Gather more data, Simple model, Use Regularization,Early stoping ).

**Underfitting**

Underfitting là mô hình mà high bias và variance low. Hiện tượng này xảy ra khi lượng dữ liệu quá ít hoặc ta cố gắng mô tả các dữ liệu phức tạp bằng các mô hình tuyến tính đơn giản như hồi quy.

Giải pháp: Thu thập thêm dữ liệu và tăng độ phức tạp của model.

## Testing and Validating

### Testing

Test set là một tập dữ liệu dùng để kiểm tra độ chính xác của model được train từ training data.

Test set là bộ dữ liệu mới không có nằm trong training set máy phải sử dụng bộ data training để học mà xử lý những dữ liệu mới chưa được train để giải quyết vấn để nhằm đánh giá khách quan tăng độ chính xác cao. Còn training set là tập dữ liệu được đưa vào để huấn luyện cho máy khi đó nếu ta đưa dữ liệu này vào thì máy sẽ đánh giá đưa ra kết quả tốt và mình không biết dữ liệu khác thì model đó có đánh giá có tốt hay không nên training set không dùng để training test để kiểm tra thử tính hiểu quả hay tính chính xác của model đem lại.

### Validating

Hyperprameter tuning là lựa chọn một tập hợp training set tối ưu cho máy học. Một Hyperprameter là một tham số có giá trị được thiết lập trước khi quá trình học tập và bắt đầu.

Validation set là một tập dữ liệu được trích ra từ tập training set để đánh giá mức độ chính xác của model. Giải pháp này thường hoạt động khá tốt. Tuy nhiên nếu bộ dữ liệu test quá nhỏ thì sẽ đánh giá mô hình không được chính xác và ngược lại nếu bộ dữ liệu test quá lớn thì bộ training còn lại ít sẽ dẫn đễn hiệu quả của máy học.

Ta cần phải dùng validation set vì nếu chỉ cho máy học trên training set và đưa ra hoạt động thì chắc chắn model đó sẽ hoạt động không như ý mình mong muốn nhằm tăng độ chính xác của model chúng ta cần phải test model trước khi sử dụng. Nếu bạn chỉ học lý thuyết mà không thực hành thì cũng không thể đánh giá được năng lực của bạn và máy cũng thế. Nhằm mục đích cho model hoạt động tốt nhất thì phải cần dùng validation set để kiểm chứng lại quá trình model học có chính xác hay không. Từ đó có thể điều chỉnh lại tập hợp training set hoặc xử lý lại model bằng thuật toán khác nhằm mục đích giúp cho model hoạt động hiệu quả nhất có thể.

## End to end machine learning project

### Look at the big picture.

### Get the data.

### Discover and visualize the data to gain insights.

### Prepare the data for machine learning algorithms.

### Train and evaluate models.

### Fine-tune your models.

### Analyze and test your solution.

### Launch, monitor, and maintain your system.