

Short Course in Artificial Intelligence

October 2023

Lecture 2: AI Applications

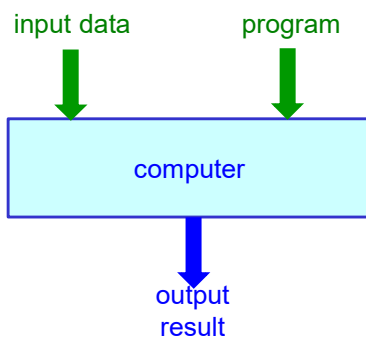
PNV 2023

1

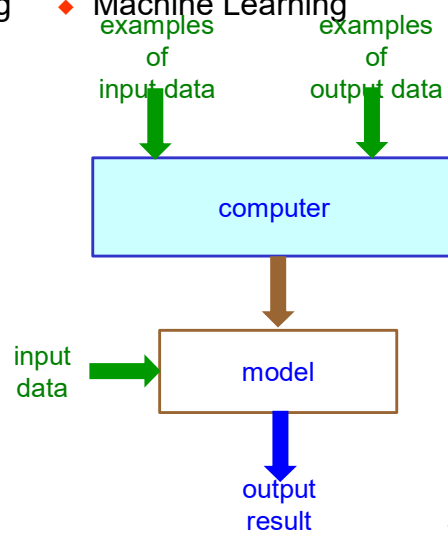
1

Traditional Programming and Machine Learning

♦ Traditional Programming



♦ Machine Learning



PNV 2023

2

2

Traditional Programming and Machine Learning

◆ Traditional Programming

Find the square of a value

Python:

```
>>> x = int(input())
2
>>> print('the square
is',x*x)
the square is 4
```

◆ Machine Learning

Find the square of a value

Provide examples:

[2,4],[3,9],[4,16],[5,25]...

Use ML to build the **model**

Use the **model** to process new data

PNV 2023

3

3

Traditional Programming and Machine Learning

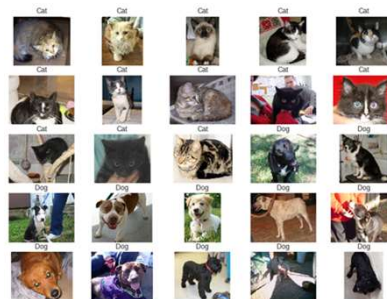
◆ Why is Machine Learning so useful ?

◆ Because we often have problems for which we do not know a **formula**

◆ Example: Is an image a **cat** or **dog** ?



cat or dog ?



PNV 2023

4

4

RECAP IN VIETNAMESE

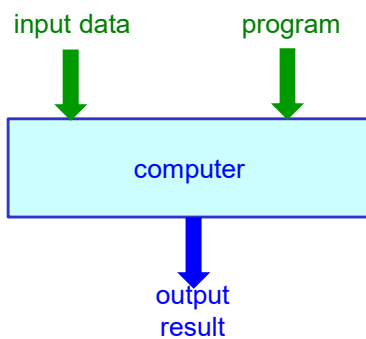
PNV 2023

5

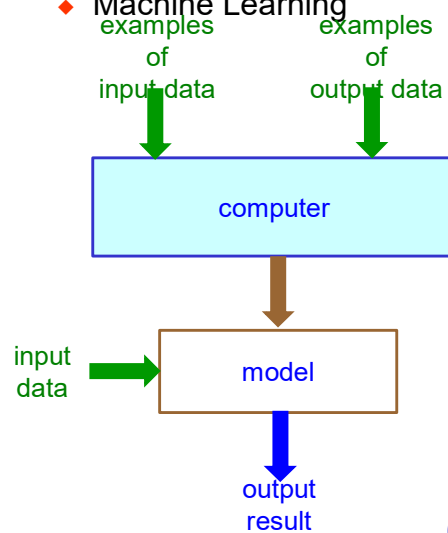
5

Lập trình truyền thống và Học máy (ML)

♦ Lập trình truyền thống



♦ Machine Learning



PNV 2023

6

6

Lập trình truyền thống và Machine Learning

♦ Lập trình truyền thống

Find the square of a value

Python:

```
>>> x = int(input())
2
>>> print('the square
is',x*x)
the square is 4
```

♦ Machine Learning

Find the square of a value

Provide examples:

[2,4],[3,9],[4,16],[5,25]...

Use ML to build the **model**

Use the **model** to process new data

PNV 2023

7

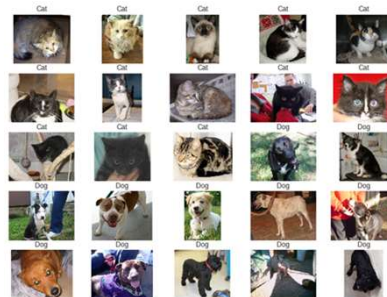
7

Lập trình truyền thống và Machine Learning

- ♦ ML rất hữu ích, ML cho phép chúng ta giải quyết các vấn đề khi mà chúng ta không biết công thức toán/thuật toán cụ thể.
- ♦ Ví dụ: Xác định xem một hình ảnh có chứa một **con mèo** hay một **con chó**?



cat or dog ?



PNV 2023

8

8

Artificial Intelligence – Machine Learning

◆ Learn from **data**

- Data = set of objects $\{x\}$ $\{images\}$
- 1. Unsupervised learning
 - Groups the objects by similarity
- 2. Supervised learning $\{(image_1, cat), (image_2, dog)\}$
 - Objects have labels $y: \{(x, y)\}$
 - From training data, build a model to predict the label

$$y = f(x)$$
- 3. Learning by reinforcement
 - Learn based on reward

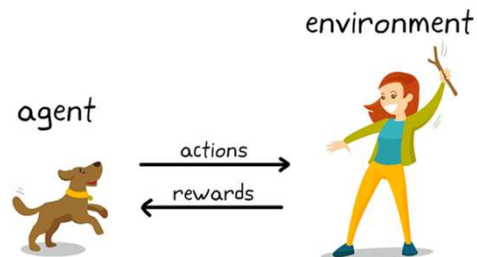
PNV 2023

9

9

Reinforcement Learning

◆ Data is **action**, action get **reward**



- ◆ Learning by adjusting **rewards**
- ◆ Typical example: games, moves are rewarded based on **winning** or **losing** game

PNV 2023

10

10

RECAP IN VIETNAMESE

PNV 2023

11

11

Học máy (Machine Learning)

- ♦ Máy tính sẽ học từ dữ liệu được cung cấp
 - thường được biểu diễn dưới dạng tập hợp các đối tượng (objects).
- ♦ Không giám sát (Unsupervised)
 - nhóm các đối tượng lại với nhau dựa trên sự tương đồng hoặc cấu trúc trong dữ liệu
- ♦ Giám sát (Supervised)
 - mỗi đối tượng trong tập dữ liệu huấn luyện được gắn kết với một nhãn hay đầu ra mong muốn
 - xây dựng một mô hình dự đoán đúng nhãn cho các đối tượng mới chưa được gán nhãn

12

12

Machine Learning (Học máy)

◆ Tăng cường (Reinforced)

- hình học thông qua tương tác với một môi trường và nhận phần thưởng (reward) dựa trên hành động của nó
- tìm hiểu cách chọn hành động tốt nhất trong một tình huống để tối đa hóa phần thưởng tích lũy

PNV 2023

13

13

Học tăng cường (Reinforcement Learning)

- ◆ Hành động được thực hiện dựa trên môi trường và nhận được phần thưởng dựa trên kết quả của hành động đó.
- ◆ Môi trường: bàn cờ và quân cờ
- ◆ Hành động là các nước đi được thực hiện.
- ◆ Mục tiêu: tìm ra cách chơi tối ưu để đạt được kết quả tốt nhất, được đánh giá thông qua các phần thưởng như thắng hoặc thua trò chơi.

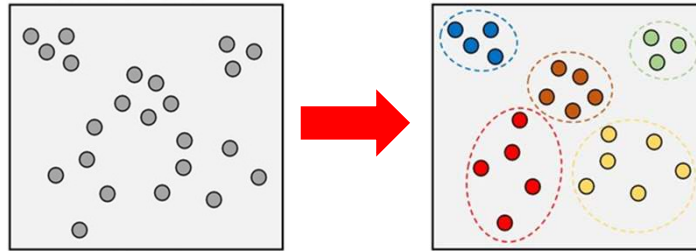


PNV 2023

14

Unsupervised Machine Learning

- ◆ Data = set of objects $\{x\}$
- ◆ Cluster similar objects together into groups



- ◆ Typical application: marketing, group customers who are likely to buy the same products

PNV 2023

15

15

RECAP IN VIETNAMESE

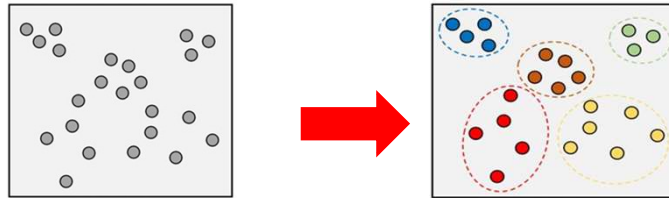
PNV 2023

16

16

Học không giám sát (Unsupervised)

- ◆ Dữ liệu = tập hợp các đối tượng
- ◆ Mục đích: Gom những đối tượng có điểm tương đồng thành 1 nhóm



- ◆ Ứng dụng phổ biến: Marketing - xác định những nhóm khách hàng có xu hướng mua những sản phẩm tương tự

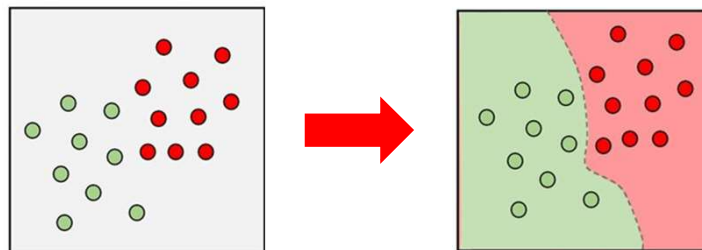
PNV 2023

17

17

Supervised Machine Learning

- ◆ Data = set of objects **with label** $\{(x, y)\}$
- ◆ Find a **model** to guess the label **y** from the value **x**



- ◆ Typical application: **prediction** of a value

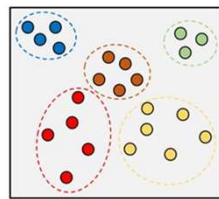
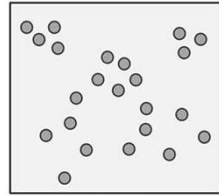
PNV 2023

18

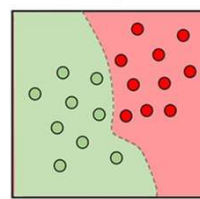
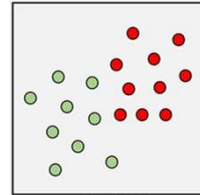
18

Machine Learning

◆ Unsupervised



◆ Supervised



PNV 2023

19

19

Supervised Machine Learning

◆ Which data ?

- Image → Computer Vision CV
- Text → Natural Language Processing NLP
- Data → Example: shopping user data

◆ Which applications ?

- Retail: predict what customers will buy
- Finance: predict future stock value
- Health: automatic diagnostic
- Spam detection: fraudulent emails or sms
- Weather forecast: predict future rain, storm, typhoon
- Vision: face recognition
- Natural Language: chatbots
- ...

PNV 2023

20

20

Supervised Machine Learning

◆ Data:

- Object: $x = (x_1, x_2, \dots, x_n)$
- Training data: set $\{x_i\} = \{(x_{i1}, x_{i2}, \dots, x_{in})\}$
- Each object x_i has a label y_i :
 $\{(x_i, y_i)\} = \{(x_{i1}, x_{i2}, \dots, x_{in}, y_i)\}$

◆ Example: student s has grade x at course C

- We predict grade at C_3 from grades at (C_1, C_2)

Student			
	16	17	14
	9	12	7
	7	5	13
...			

PNV 2023

21

21

Supervised Machine Learning

- ### ◆ To make it simpler, we only predict if the student **succeeds** or **fails** in C_3

◆ Data:

Student				
	S	16	17	14
	F	9	12	7
	F	7	5	13
...				

- ### ◆ For easier mathematics, we note:

- Success: **S** $y_1 = +1$
- Failure : **F** $y_2 = -1$ $y_3 = -1$

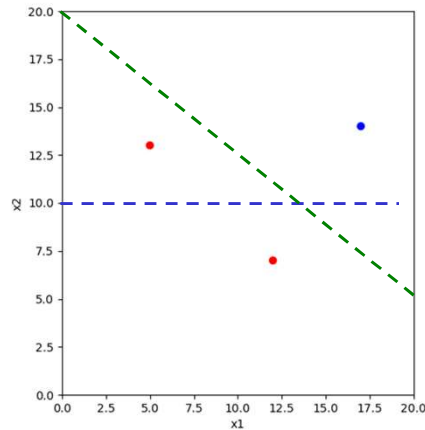
PNV 2023

22

22

Supervised Machine Learning

- ◆ Linear Classifier: $w_1x_1 + w_2x_2 + w_0 = 0$



PNV 2023

23

23

Supervised Machine Learning

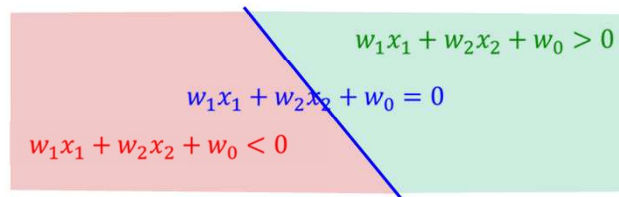
- ◆ The boundary of the linear classifier is the line:

$$w_1x_1 + w_2x_2 + w_0 = 0$$

- ◆ The line splits the space into two half-spaces

- ◆ $w_1x_1 + w_2x_2 + w_0 > 0$

- ◆ $w_1x_1 + w_2x_2 + w_0 < 0$



PNV 2023

24

24

Supervised Machine Learning

- ◆ How to use a linear classifier ?
- ◆ Given a set of points, we define an error function and we find the values of (w_1, w_2, w_0) that minimize this function
 - The common minimization technique is called "Gradient Descent"
- ◆ With the values of (w_1, w_2, w_0) we can classify any point (x_1, x_2) :
 - we compute $w_1x_1 + w_2x_2 + w_0$
 - if $w_1x_1 + w_2x_2 + w_0 > 0$, the classifier predicts $y = +1$
 - if $w_1x_1 + w_2x_2 + w_0 < 0$, the classifier predicts $y = -1$

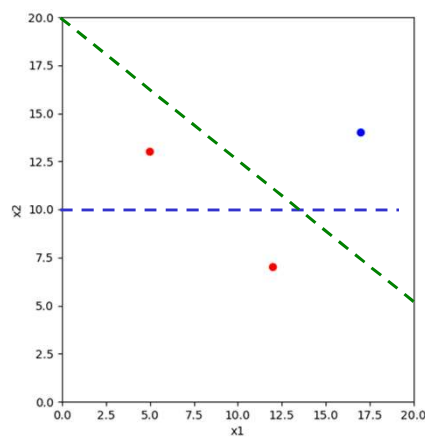
PNV 2023

25

25

Supervised Machine Learning

- ◆ Linear Classifier: $w_1x_1 + w_2x_2 + w_0 = 0$



1 error	

0 error	

PNV 2023

26

26

RECAP IN VIETNAMESE

PNV 2023

27

27

Học giám sát (Supervised)

◆ Dữ liệu: hình ảnh, văn bản hoặc các dữ liệu khác.

◆ Ứng dụng

- Bán lẻ (Retail): Dự đoán mua sắm của khách hàng trong tương lai.
- Tài chính (Finance): Dự đoán giá trị cổ phiếu trong tương lai.
- Y tế (Health): Tự động chẩn đoán bệnh và dự đoán kết quả điều trị.
- Phát hiện thư rác (Spam detection): Phát hiện email hay tin nhắn gian lận.
- Dự báo thời tiết (Weather forecast): Dự đoán thời tiết trong tương lai.
- Thị giác máy tính (Computer Vision): Nhận diện khuôn mặt và đối tượng.
- Xử lý ngôn ngữ tự nhiên (Natural Language Processing): Xây dựng chatbot và phân loại văn bản.

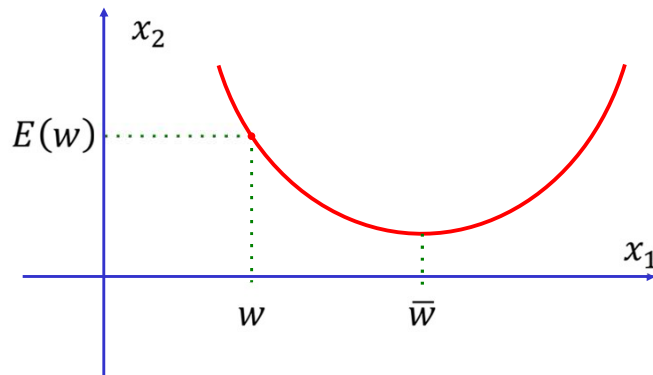
PNV 2023

28

28

Intuition for Gradient Descent

- ◆ We want to minimize an Error function $E(w)$



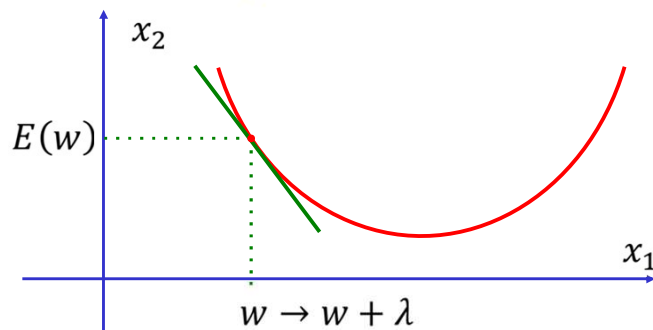
PNV 2023

29

29

Intuition for Gradient Descent

- ◆ We use the derivative to reduce the value $E(w)$
 - if the derivative is negative, we increase $w \rightarrow w + \lambda$
 - if the derivative is positive, we decrease $w \rightarrow w - \lambda$



PNV 2023

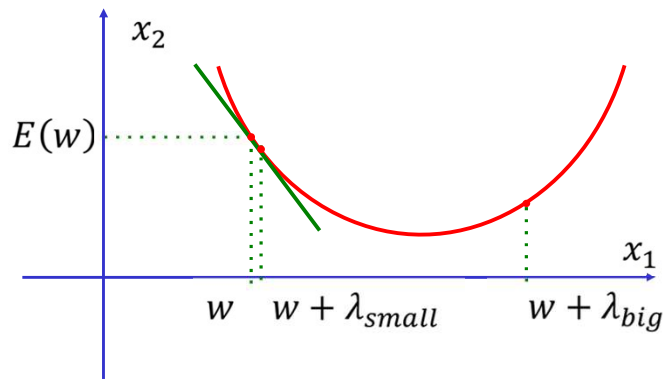
30

30

Intuition for Gradient Descent

◆ Problems:

- if λ is too small, it takes very long to reach \bar{w}
- if λ is too big, we miss \bar{w} and we need to come back



PNV 2023

31

31

Intuition for Gradient Descent



◆ Gradient Descent procedure:

- start with an initial value of w
- compute the derivative
- use it to modify the value of w by an amount that depends on the learning rate λ
- check the error to make sure that λ is not too big
- iterate until convergence
 - the iterations are called "epoch"

PNV 2023

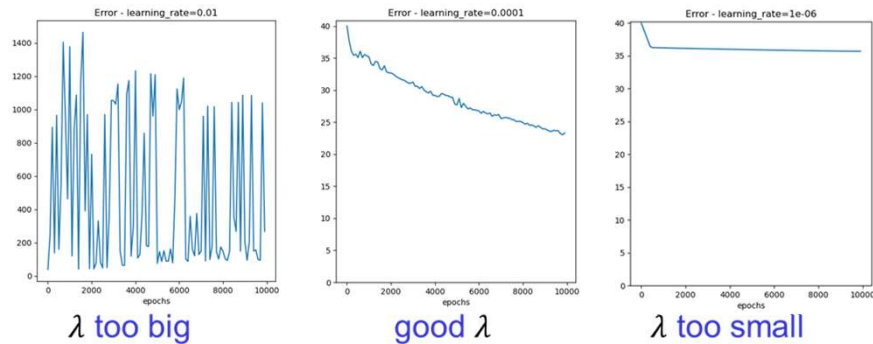
32

32

Intuition for Gradient Descent

◆ Gradient Descent procedure:

- to find a good learning rate, watch how the value of the error changes with the number of epochs



PNV 2023

33

33

Intuition for Gradient Descent

- ◆ A good value for the learning rate may be different for each function
- ◆ There are many algorithms to change the value of the learning rate during the epochs:
 - Start with a high learning rate → move fast
 - Later reduce the learning rate → be accurate

PNV 2023

34

34

RECAP IN VIETNAMESE

PNV 2023

35

35

Triết lý của Gradient Descent (GD)

- ◆ Learning rate (tỷ lệ học) là một tham số quan trọng trong huấn luyện mô hình học máy.
- ◆ Giá trị tốt cho learning rate có thể khác nhau cho từng hàm số.
- ◆ Learning rate phù hợp -> kết quả chính xác
- ◆ Có nhiều thuật toán để thay đổi learning rate trong quá trình huấn luyện.
 - Learning rate cao -> di chuyển nhanh
 - Giảm learning rate -> đạt độ chính xác cao hơn.

PNV 2023

36

36

AI Applications

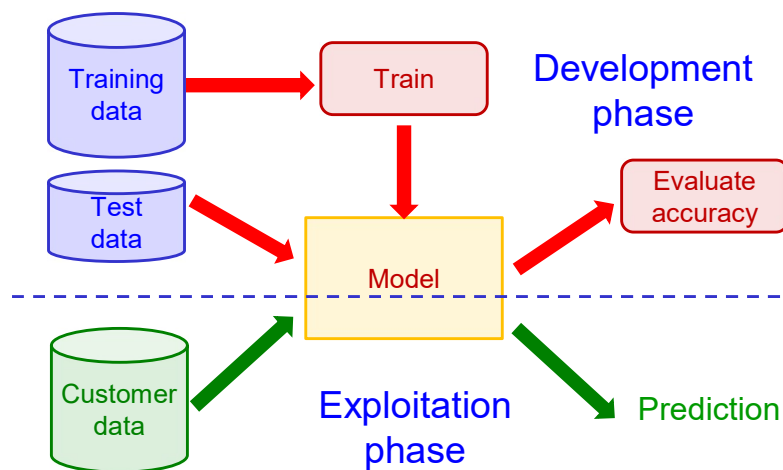
- ◆ How to build a AI application ?
- ◆ Steps:
 1. Collect data
 2. Prepare the data (into suitable format)
 - cleaning, labelling, ...
 3. Choose a model
 4. Train the model
 - find the best values of the parameters
 5. Evaluate the model
 - on data that was not used to train
 6. Make predictions
 - on new data

PNV 2023

37

37

AI Applications

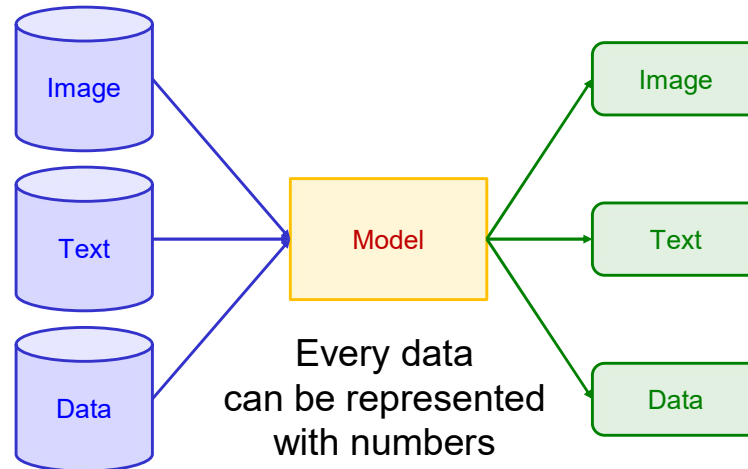


PNV 2023

38

38

AI Applications



PNV 2023

39

39

AI Applications

- ◆ Image → Data
- ◆ Face Recognition
- ◆ Emotion detection

- ◆ Self-driving cars



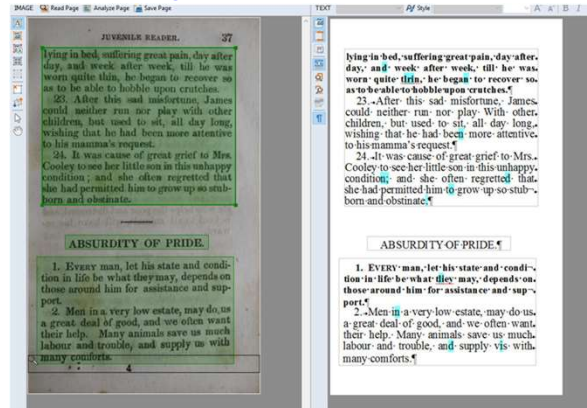
PNV 2023

40

40

AI Applications

- ◆ Image → Text
- ◆ Character Recognition



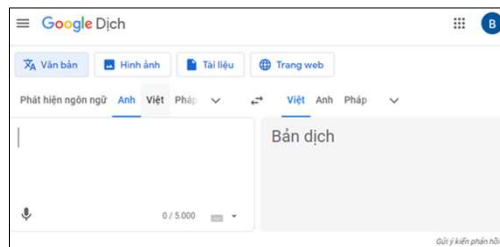
PNV 2023

41

41

AI Applications

- ◆ Text → Text
- ◆ Automatic Translation



- ◆ Summarization
- ◆ Information search

PNV 2023

42

42

AI Applications

- ◆ Speech Data → Text
- ◆ Speech Recognition
- ◆ Language Learning



- ◆ Text → Speech
- ◆ Speech Synthesis



PNV 2023

43

43

AI Applications

- ◆ Marketing: targeted advertising
- ◆ Is the customer **X** willing to buy product **Y**
- ◆ Data:
 - customer age
 - salary
 - family
 - address
 - products already bought
 - web pages visited
 - ...
- ◆ Prediction: the customer will buy **Y** ? (yes/no)



PNV 2023

44

44

AI Applications

- ◆ Common question to AI engineers from users:
 - how much data do you need ?
- ◆ Common answer:
 - as much as possible
- ◆ But **quality** of data is important

DATA = \$\$
Personal Data = \$\$\$

PNV 2023

45

45

RECAP IN VIETNAMESE

PNV 2023

46

46

Cách xây dựng một ứng dụng AI

◆ Bước 1: Thu thập dữ liệu

- Thu thập dữ liệu liên quan đến vấn đề mà bạn muốn giải quyết bằng trí tuệ nhân tạo.

◆ Bước 2: Chuẩn bị dữ liệu

- Chuẩn bị dữ liệu để nó phù hợp với định dạng phù hợp cho mô hình học máy.
- Tiến hành làm sạch, gán nhãn và xử lý dữ liệu để loại bỏ nhiễu và chuẩn hóa dữ liệu.

◆ Bước 3: Chọn một mô hình

- Chọn một mô hình học máy phù hợp với vấn đề của bạn.
- Có thể là một mô hình học máy có giám sát (supervised), không giám sát (unsupervised) hoặc học tăng cường (reinforcement learning).

PNV 2023

47

47

Cách xây dựng một ứng dụng AI

◆ Bước 4: Huấn luyện mô hình

- Sử dụng dữ liệu đã chuẩn bị để huấn luyện mô hình.
- Tìm kiếm các giá trị tối ưu của các tham số trong mô hình.

◆ Bước 5: Đánh giá mô hình

- Đánh giá hiệu suất của mô hình bằng cách sử dụng dữ liệu mà không được sử dụng trong quá trình huấn luyện.
- Đánh giá các độ đo như độ chính xác, độ phân loại chính xác, hay độ đo F1.

◆ Bước 6: Tiến hành dự đoán

- Sử dụng mô hình đã huấn luyện để thực hiện dự đoán trên dữ liệu mới.
- Tiến hành dự đoán cho các trường hợp mà mô hình chưa từng thấy trong quá trình huấn luyện.

PNV 2023

48

48

Practice 2: Learning Linear Classifier

- ◆ You are provided with a Python notebook that already contains the code to train and visualize
- ◆ There are 3 datasets, from easy to more difficult
- ◆ You should experiment with several **learning rates** and the **number of epochs** to achieve a good classifier
- ◆ Note: not all values for the learning_rate will lead to a successful training

PNV 2023

49

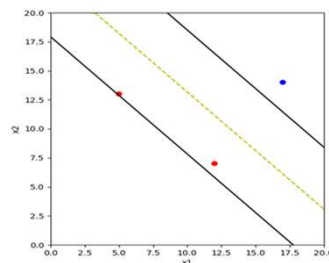
49

Practice 2: Learning Linear Classifier

- ◆ Data : student and grades

Student	Grade1	Grade2	Success
S1	17	14	1
S2	12	7	-1
S3	5	13	-1

- ◆ Goal: build **linear classifier** to **predict** success



PNV 2023

50

50

RECAP IN VIETNAMESE

PNV 2023

51

51

Luyện tập 2: Học phân loại tuyến tính

- ◆ Sử dụng code Python đã cho để huấn luyện và trực quan hóa dữ liệu.
- ◆ Bộ dữ liệu gồm 3 loại, từ dễ đến khó hơn.
- ◆ Thử nghiệm với tốc độ học và số epochs khác nhau để tìm ra bộ phân loại tốt.
- ◆ Tốc độ học quyết định tốc độ cập nhật trọng số trong quá trình huấn luyện.
- ◆ Số epochs là số lần mô hình được huấn luyện trên toàn bộ dữ liệu.
- ◆ Cần thử nghiệm để tìm giá trị tốc độ học và số epochs tối ưu.
- ◆ *Lưu ý không phải tất cả các giá trị tốc độ học đều thành công.*

PNV 2023

52

52

Introduction to Deep Learning

PNV 2023

53

53

A Quick Course On Deep Networks

- ◆ Deep Learning is a particular type of models based on the **combination of linear classifiers**
- ◆ Those models are the basis for most of the recent progress in AI/ML
- ◆ They can be applied to (almost) any type of data (text, image, speech, data...), and can be extremely efficient for very diverse tasks.
- ◆ The models can be huge (billions of parameters) and be trained on huge training sets.

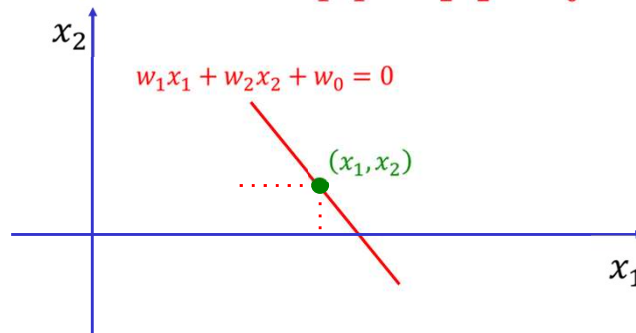
PNV 2023

54

54

A Quick Course On Deep Networks

- ◆ The real plane \mathbb{R}^2
- ◆ Points (vectors) (x_1, x_2)
- ◆ Lines $w_1x_1 + w_2x_2 + w_0 = 0$



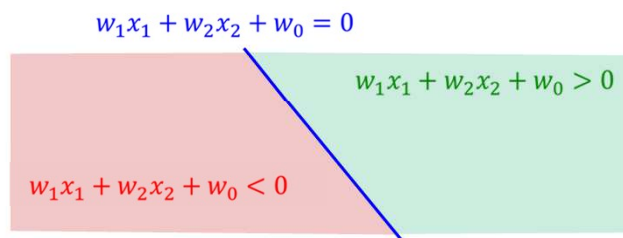
PNV 2023

55

55

A Quick Course On Deep Networks

- ◆ Lines $w_1x_1 + w_2x_2 + w_0 = 0$
- ◆ Half-spaces
 - ◆ $w_1x_1 + w_2x_2 + w_0 > 0$
 - ◆ $w_1x_1 + w_2x_2 + w_0 < 0$



PNV 2023

56

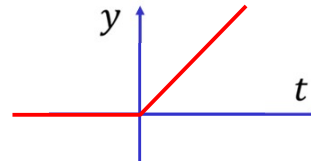
56

A Quick Course On Deep Networks

◆ The ReLU Function (Rectified Linear Unit)

$$\text{ReLU}(t) = \max(t, 0)$$

$$\text{ReLU}(t) = \begin{cases} t & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}$$



◆ ReLU on the real plane \mathbb{R}^2

$$\begin{aligned} \text{ReLU}(x_1, x_2) &= \text{ReLU}(w_1 x_1 + w_2 x_2 + w_0) \\ &= \max(w_1 x_1 + w_2 x_2 + w_0, 0) \end{aligned}$$

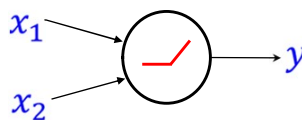
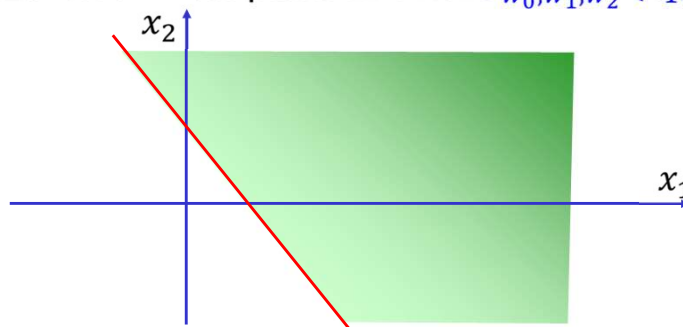
PNV 2023

57

57

A Quick Course On Deep Networks

◆ ReLU on the real plane \mathbb{R}^2 : $\text{ReLU}_{w_0, w_1, w_2}(x_1, x_2)$



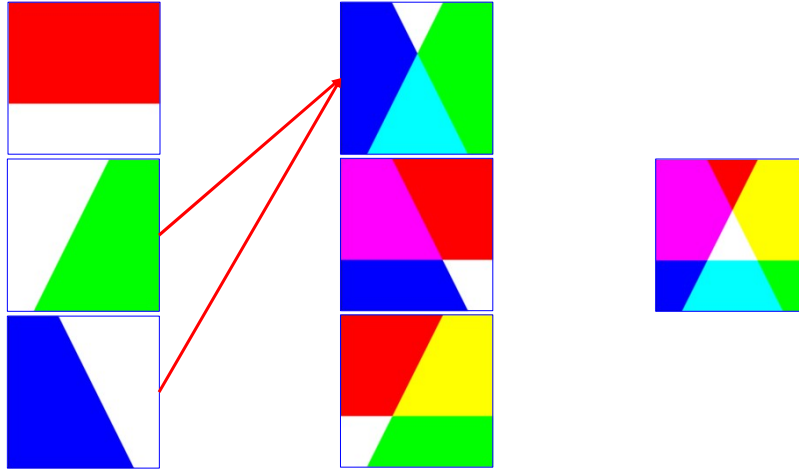
PNV 2023

58

58

A Quick Course On Deep Networks

♦ Combining linear classifiers



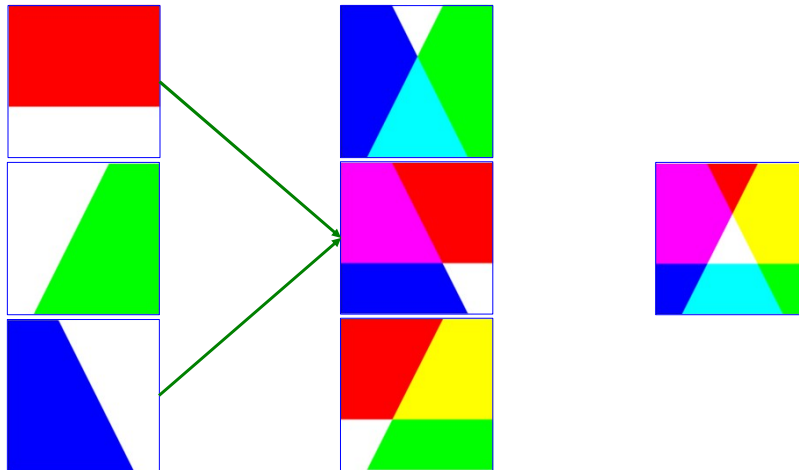
PNV 2023

59

59

A Quick Course On Deep Networks

♦ Combining linear classifiers



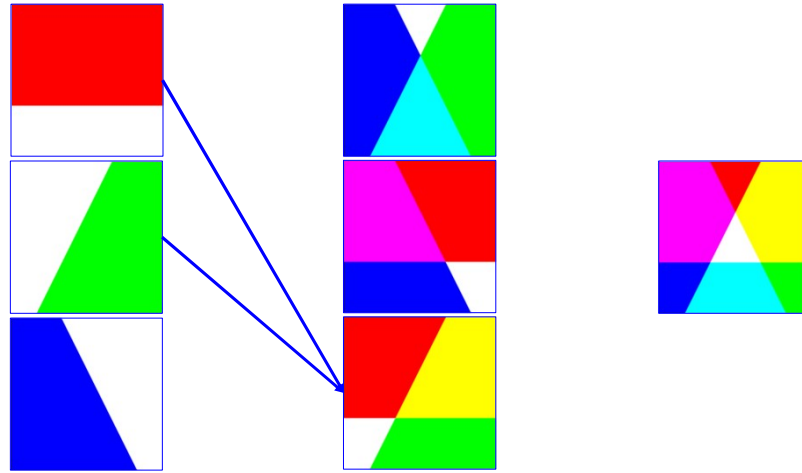
PNV 2023

60

60

A Quick Course On Deep Networks

♦ Combining linear classifiers



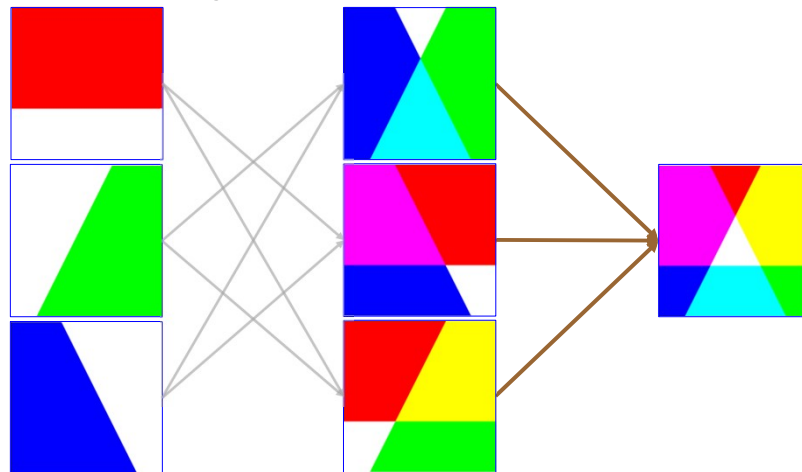
PNV 2023

61

61

A Quick Course On Deep Networks

♦ Combining linear classifiers



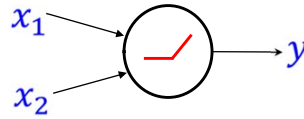
PNV 2023

62

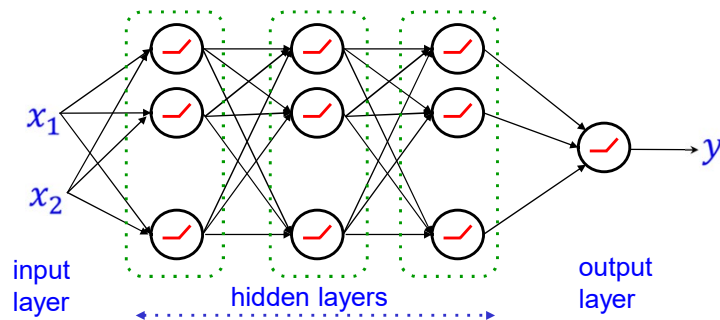
62

A Quick Course On Deep Networks

◆ From ReLu



◆ To Deep Networks

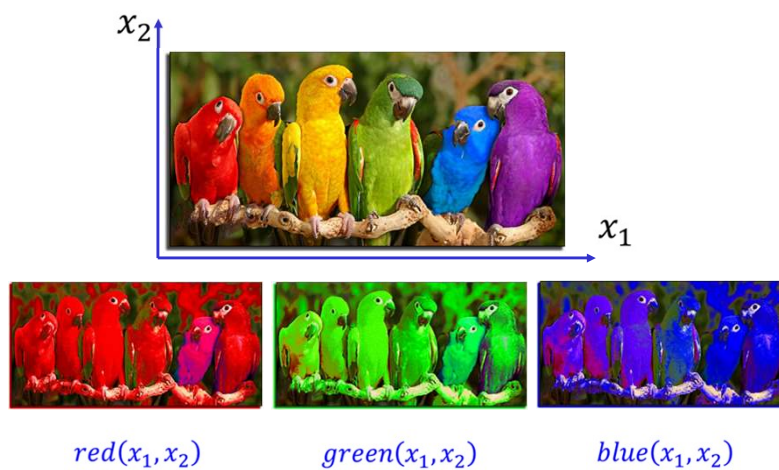


PNV 2023

63

63

A Quick Deep Network

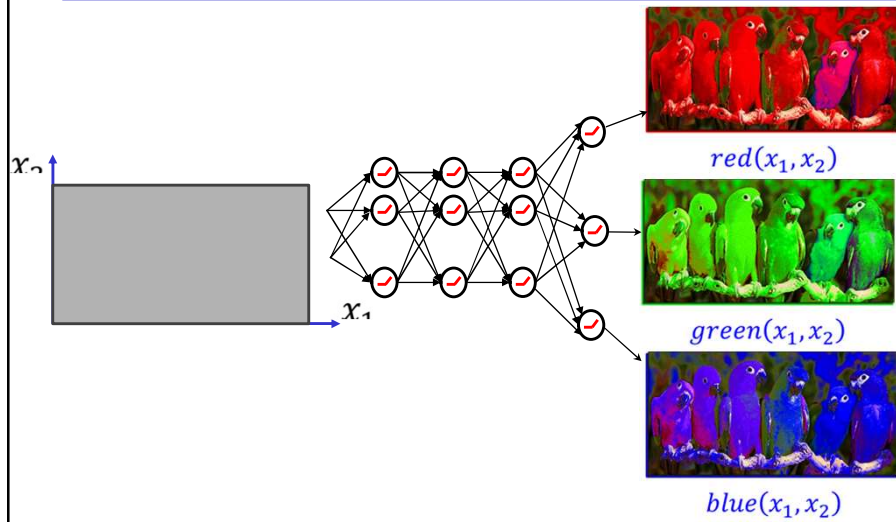


PNV 2023

64

64

A Quick Deep Network



PNV 2023

65

65

RECAP IN VIETNAMESE

PNV 2023

66

66

Giới thiệu về Deep Learning

- ◆ **Deep Learning**: loại mô hình đặc biệt dựa trên sự kết hợp của các bộ phân loại tuyến tính.
- ◆ Áp dụng cho hầu hết các loại dữ liệu (văn bản, âm thanh, hình ảnh, ...)
- ◆ Có thể được huấn luyện với các tập dữ liệu rất lớn.
- ◆ Học được các đặc trưng phức tạp và thực hiện các nhiệm vụ phân loại, nhận dạng, dịch thuật, và nhiều tác vụ khác với hiệu suất cao và khả năng tổng quát hóa tốt.

PNV 2023

67

67

Practice 3: Learning Deep Networks

- ◆ You are given a Javascript code that trains a Deep Network to represent a color image
- ◆ You can see how the network evolves during training
- ◆ You can modify the **number of hidden layers** (just cut and paste one line) to see how deeper networks can better represent the image

PNV 2023

68

68

Practice 3: Learning Deep Networks



◆ Data :

- One image 200x200



◆ Find models of the color functions:

- $red(x_1, x_2)$
- $green(x_1, x_2)$
- $blue(x_1, x_2)$

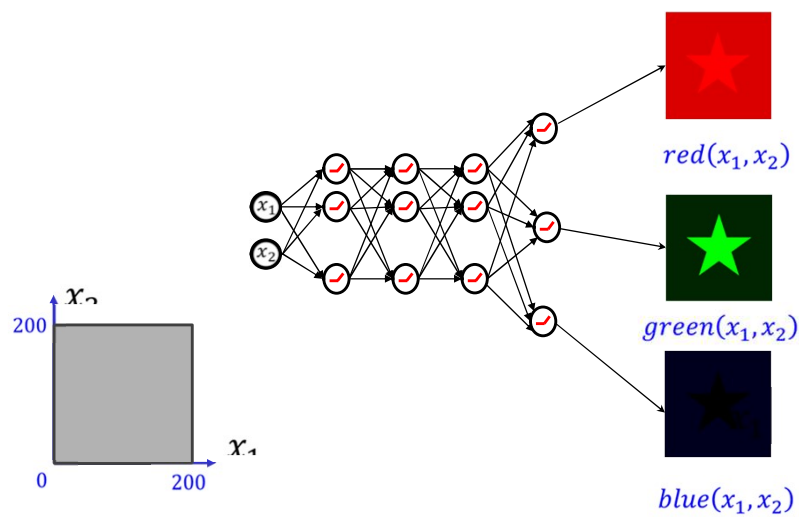


PNV 2023

69

69

Practice 3: Learning Deep Networks



PNV 2023

70

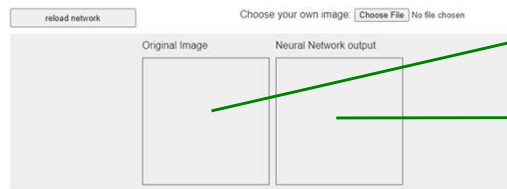
70

Practice 3: Learning Deep Networks

ConvnetJS demo: Image "Painting"

Credits to [Andrei Karpathy](#)

```
layer_defs = []
// input layer
layer_def.push({type:'input', out_size1, out_size2, out_depth2}); // 2 inputs: x, y
// fully connected hidden layer(s)
layer_def.push({type:'fc', num_neurons:20, activation:'relu'});
// output layer
layer_def.push({type:'regression', num_neurons:3}); // 3 outputs: r,g,b
net = new convnetjs.Net();
net.makeLayers(layer_defs);
trainer = new convnetjs.SGDTrainer(net, {learning_rate:0.01, momentum:0.9, batch_size:5, l2_decay:0.0});
```



Learning rate: 0.01
The learning rate should probably be decreased over time (slide left) to let the network better overfit the training data. It's nice to not have to worry about overfitting.

input layer (x_1, x_2)

hidden layer
20 neurons

output layer (r, g, b)

training image

model image

learning rate

PNV 2023

71

71

Practice 3: Learning Deep Networks

ConvnetJS demo: Image "Painting"

Credits to [Andrei Karpathy](#)

```
layer_defs = []
// input layer
layer_def.push({type:'input', out_size1, out_size2, out_depth2}); // 2 inputs: x, y
// fully connected hidden layer(s)
layer_def.push({type:'fc', num_neurons:20, activation:'relu'});
// output layer
layer_def.push({type:'regression', num_neurons:3}); // 3 outputs: r,g,b
```

hidden layer
20 neurons

Duplicating this line will add new hidden layers to the model



Learning rate: 0.01
The learning rate should probably be decreased over time (slide left) to let the network better overfit the training data. It's nice to not have to worry about overfitting.

PNV 2023

72

72