



Introduction to Machine Learning

Prof. Chia-Yu Lin
Yuan Ze University
2022 Spring

Thanks to the slides of Prof. P. Domingos from Washington University, Prof. H.-T. Lin and Prof. Lee Hung-Yi Lee from NTU.

Outline

- Introduction to Machine Learning
 - Supervised Learning
 - Unsupervised Learning
 - Semi-supervised Learning
 - Reinforcement Learning
 - Generative Adversarial Network
 - 生成對抗網路

AI is so Easy

Like gardening

- **Seeds** = Algorithms
- **Nutrients** = Data
- **Gardener** = You
- **Plants** = Programs

You can do it!



Data Mining = AI Algorithm?

- Dig out invisible but invaluable minerals.
- Data mining is dig out **meaningful information** from meaningless data.

- Walmart



- Association rule
- =>Find 「pattern」 but no intelligence

Learning is to find a function

- Voice Recognition

$f($



$) =$ 鳥叫聲

- Speech Recognition

$f($



$) =$ “我不知道你說什麼”

- Image recognition

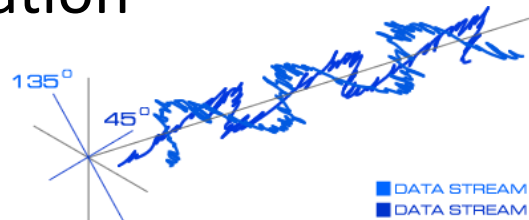
$f($



$) =$ “Seafood”

- Channel estimation

$f($



$) =$ Channel parameters

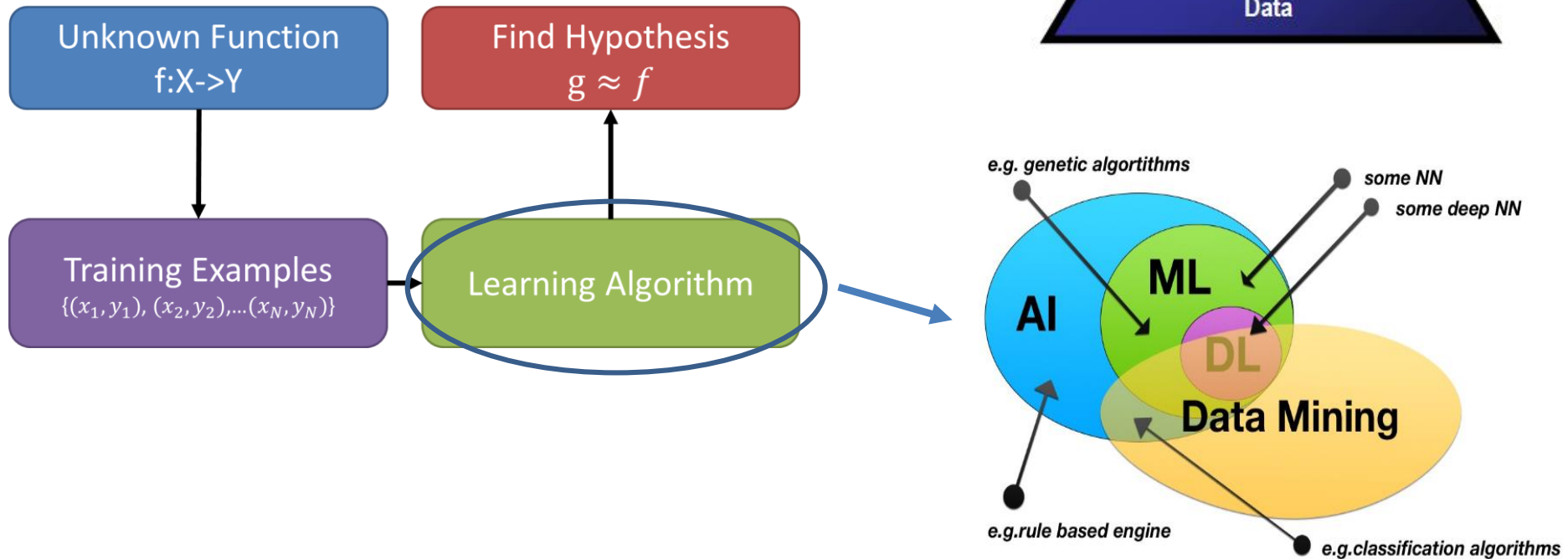
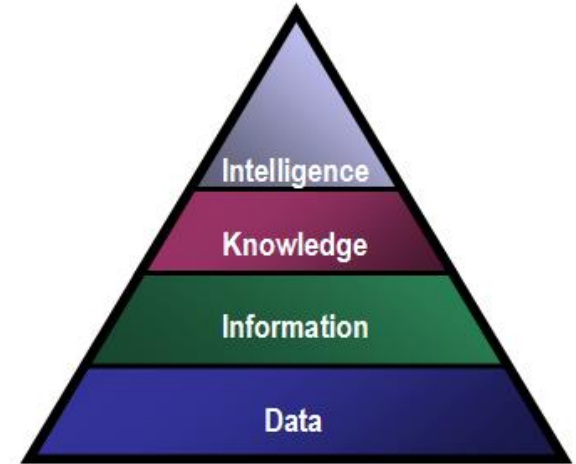
http://cdn1.itpro.co.uk/sites/itpro/files/images/dir_176/it_photo_88225.jpg

http://www.berkeleywellness.com/sites/default/files/field/image/ThinkstockPhotos-520490716_field_img_hero_988_380.jpg

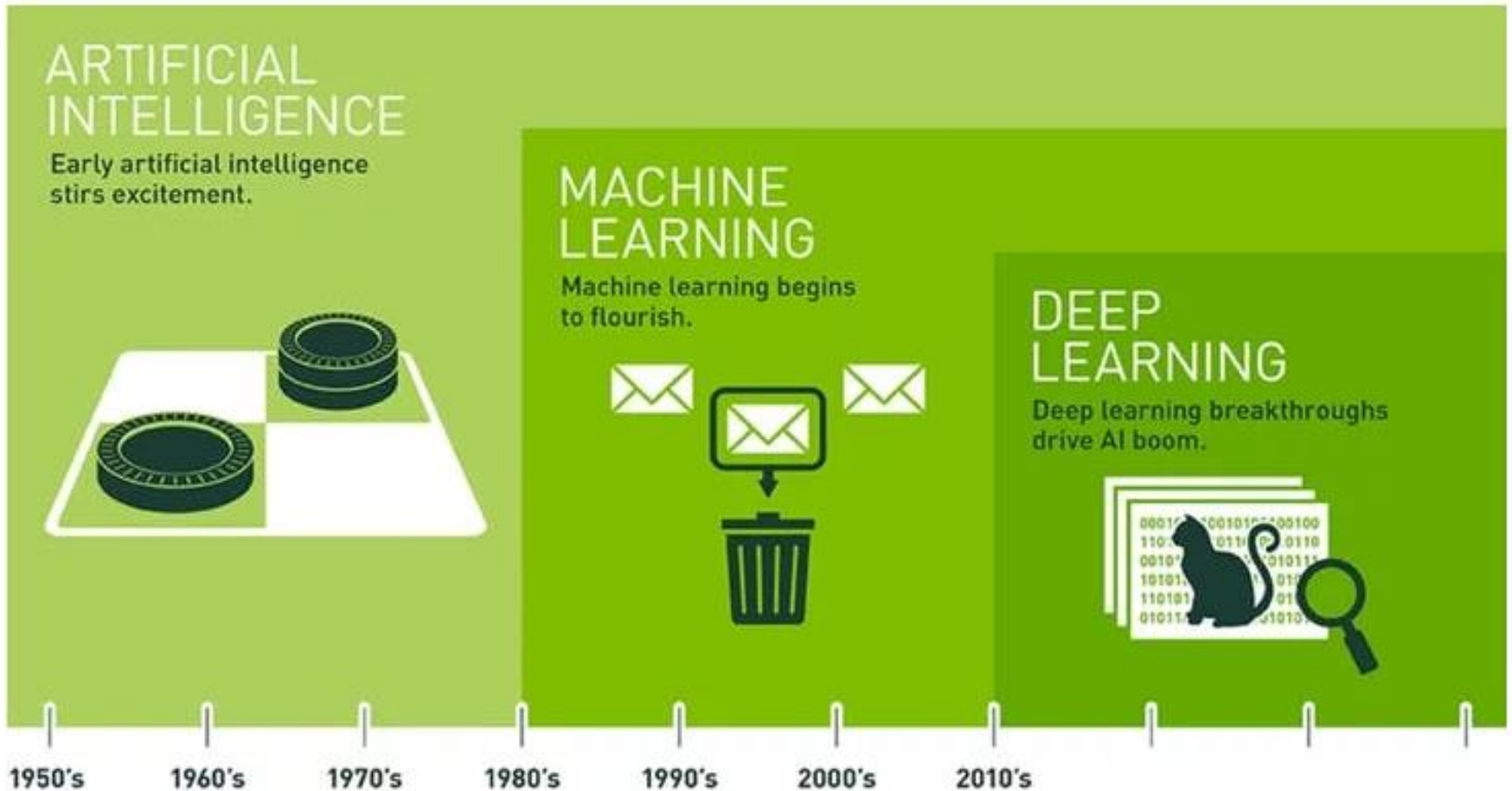
<https://telcoantennas.com.au/site/sites/default/files/images/4G-cross-polarisation-low-signal-areas.png>

There are many methods to learn!

- Learning is to find a function!!
- How do we find the function?
- There are many learning algorithms.



AI, ML, DL



圖片來源：NVIDIA

Machine Learning

- Machine Learning
 - 通過演算法來分析數據、從中學習找到一個函式，來判斷或預測現實世界裡的某些事
 - 並非手動編寫帶有特定指令的軟體程序來完成某個特殊任務
 - 是使用大量的資料和演算法來「訓練」機器，讓它學習如何執行任務。
- Tens of thousands of machine learning algorithms
- Hundreds new every year
- Every machine learning algorithm has three components:
 - **Representation**
 - **Evaluation**
 - **Optimization**



Representation

- Decision trees
- Sets of rules / Logic programs
- Instances
- Graphical models (Bayes/Markov nets)
- Neural networks
- Support vector machines
- Model ensembles
- Etc.

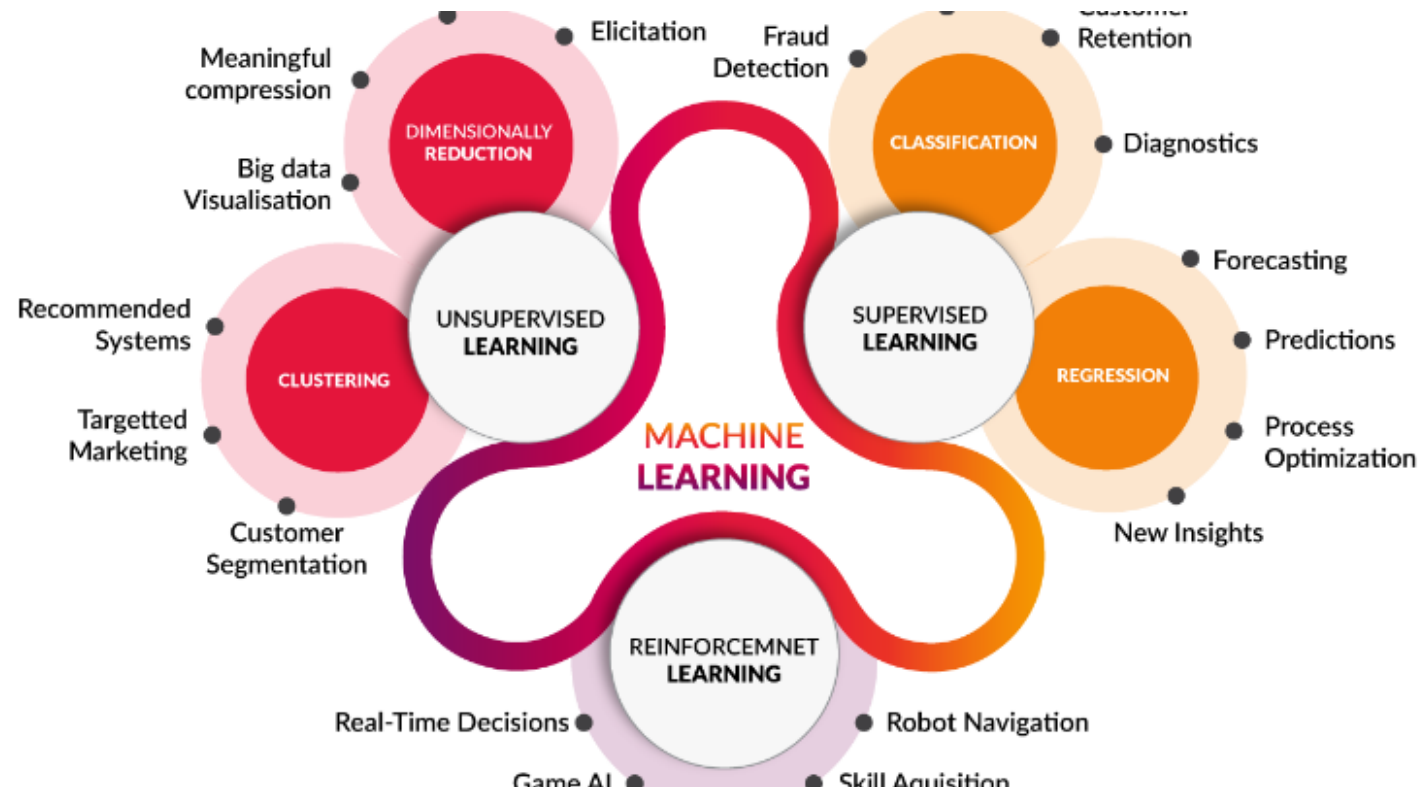
Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- Etc.

Optimization

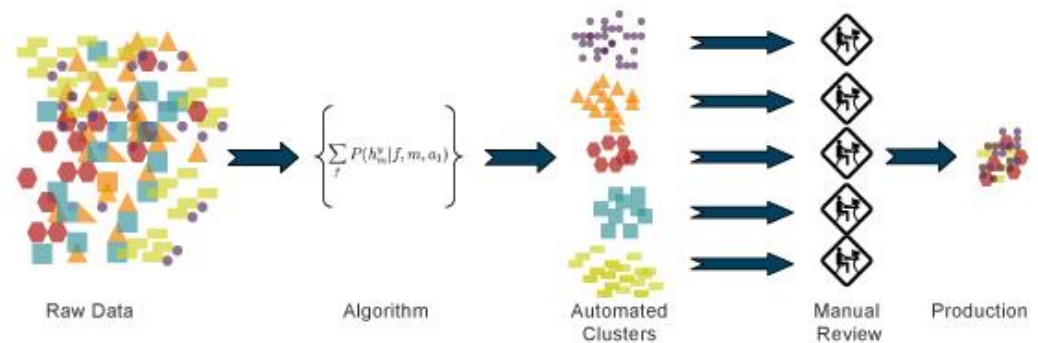
- Combinatorial optimization
 - E.g.: Greedy search
- Convex optimization
 - E.g.: Gradient descent
- Constrained optimization
 - E.g.: Linear programming

Representation



Unsupervised Learning (1/2)

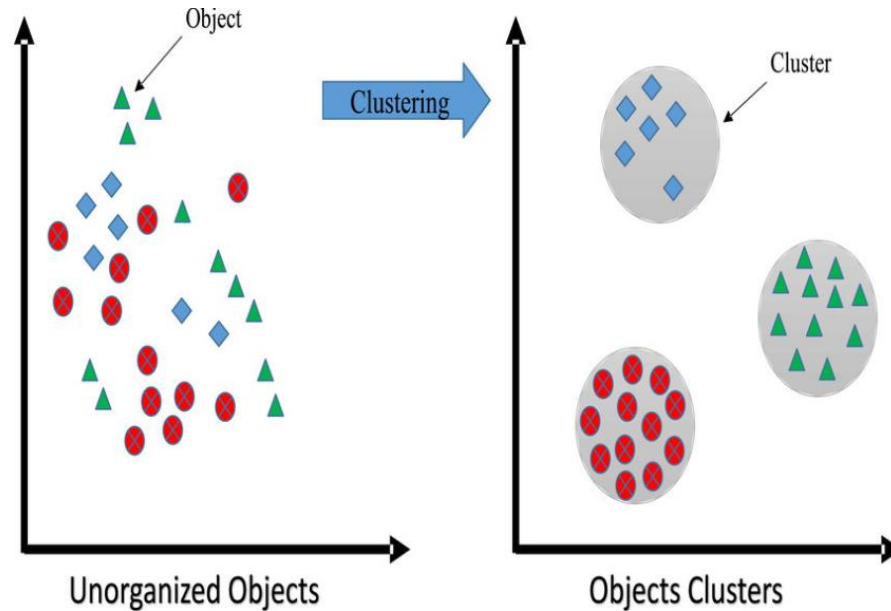
- Unsupervised Learning is the second type of machine learning, in which **unlabeled data are used** to train the algorithm, which means it used against data that has **no historical labels**.
- The purpose is to explore the data and find some structure within.
 - Clustering
 - Anomaly Detection
 - Association Rule
 - Autoencoder



Unsupervised Learning (2/2)



- Clustering



- Association Rule

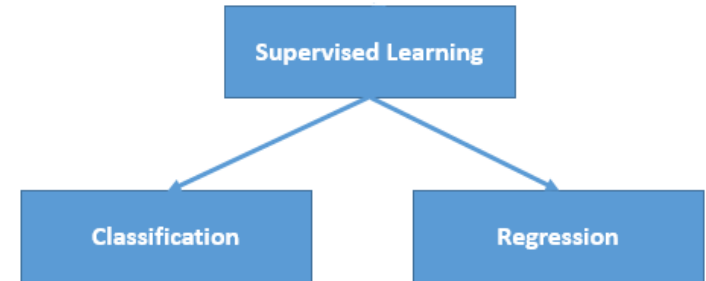


摩根大通應用非監督去降低業務流程



- 早期創業家尋求銀行貸款，第一印象通常是最重要的，但是也**因為資訊的缺乏**造成銀行職員誤判的結果。
- 為了使日常工作自動化並減少分析業務往來所需的時間及錯誤，摩根大通開發了一種專有的ML算法，稱為Contract Intelligence或COiN。
- COiN的工作就是**自動執行特定類別的文件審查，採用圖像辨識來識別文件中的模式**，而背後的演算法即使用了無監督學習。
- 演算法通過分析各個銀行契約中的數據，**並抓出既定模式**，COiN可以將條款細分為150種不同的信用貸款契約，並從中擷取重要的訊息。
- COiN不僅可以在幾秒內處理12,000個信貸協議，更是讓銀行省下了360,000個工時。

Supervised Learning

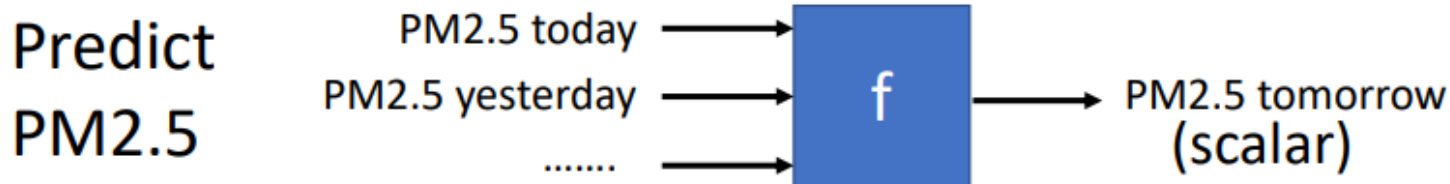


- With labeled data
- ***Regression***
 - The type of Supervised Learning in which labelled data used, and this data is used to make predictions in a **continuous** form.
 - The output of the input is always ongoing, and the graph is linear.
 - EX: House Price Prediction
- **Classification**
 - The type of Supervised Learning in which labelled data can use, and this data is used to make predictions in a **non-continuous** form.
 - The output of the information is not always continuous, and the graph is non-linear.
 - EX: check whether the email is spam or not spam.

Regression Example

Regression

The output of the target function f is “scalar”.



Training Data:

Input:

9/01 PM2.5 = 63 9/02 PM2.5 = 65

Input:

9/12 PM2.5 = 30 9/13 PM2.5 = 25

Output:

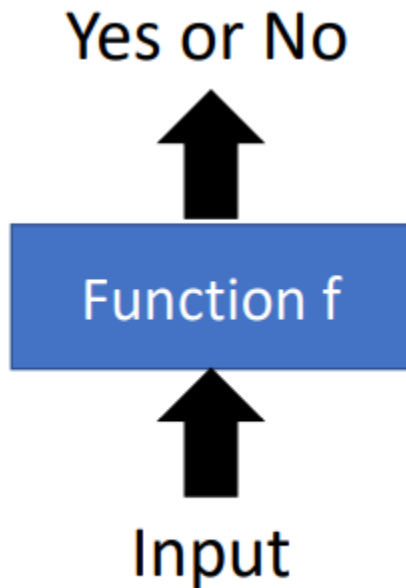
9/03 PM2.5 = 100

Output:

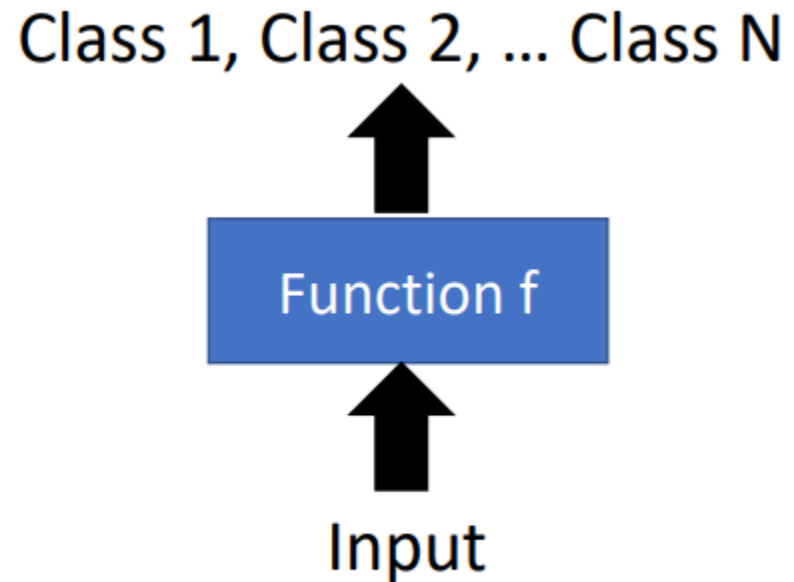
9/14 PM2.5 = 20

Classification Example

Binary Classification



Multi-class Classification



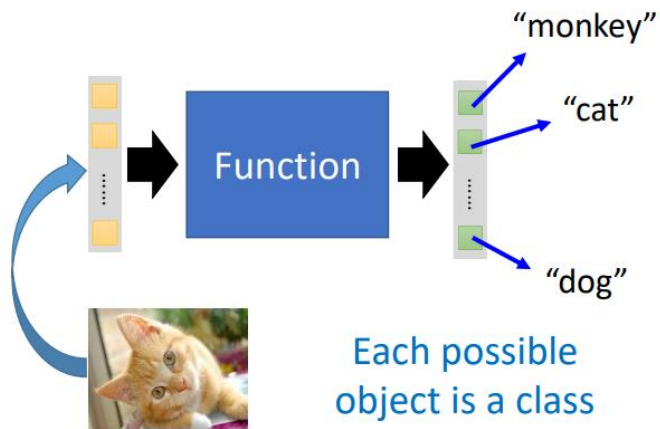
Binary Classification Example

Spam
filtering

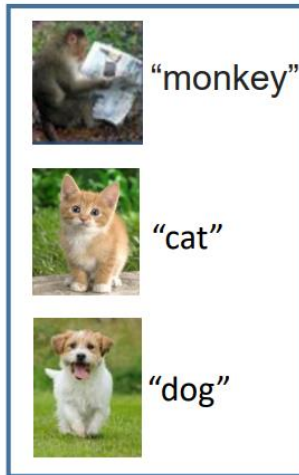


Multi-Classification Example

Image Recognition



Training Data

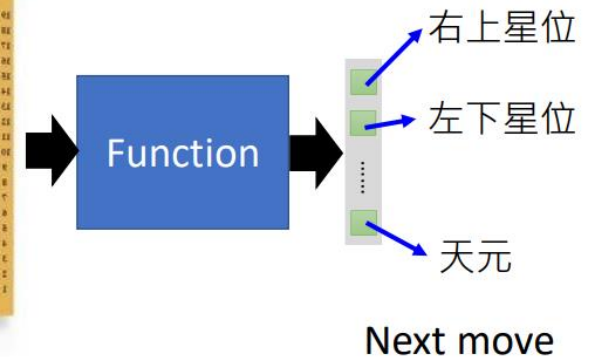


Playing GO



Each position
is a class

(19 x 19 classes)



Object Detection

- Object detection of multiple classes

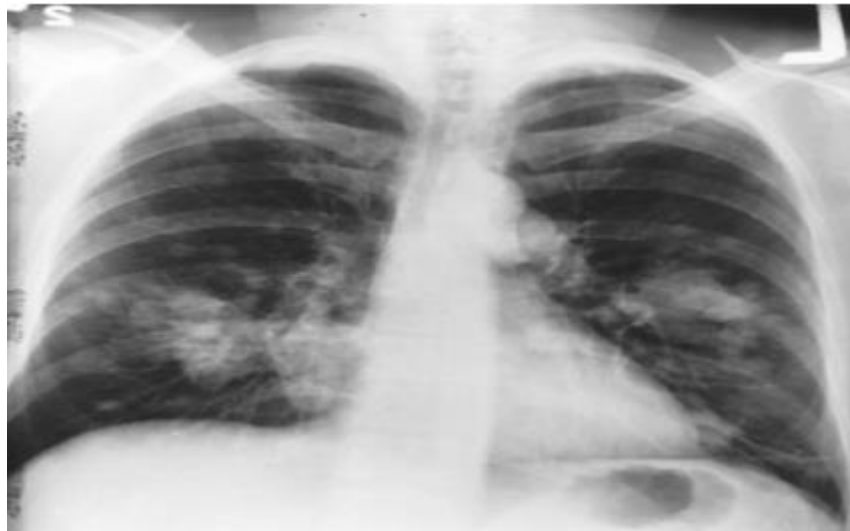
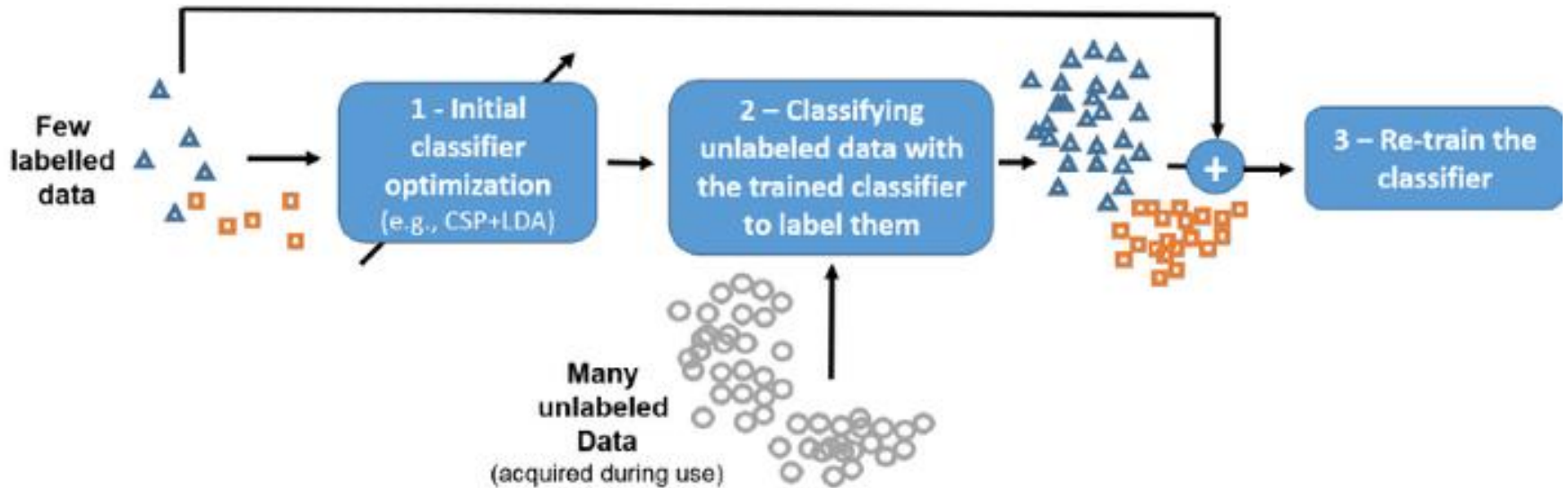


Semi-Supervised Learning (1/2)



- Both types of raw data used.
- Hybrid of supervised and unsupervised machine learning.
- The Semi-supervised employs both labelled and unlabeled data for training typically **a small amount** of labelled data with **a significant amount** of unlabeled data.
- Labelling massive amounts of data
 - Time-consuming
 - Expensive.
 - Human biases on the model
- Include lots of unlabeled data
 - Improve the accuracy of the final model while reducing the time and cost spent building it.

Semi-Supervised Learning (2/2)



小結

監督式學習

- 需要有標記(Labeled)的數據
- 可用於迴歸分析 (數值的預測)
- 可用於分群分析 (分類)

- 線性回歸 (Linear regression)
- 單純貝式 (Naive bayes)
- 決策樹 (Decision Tree)



- 優化定價策略、預估價格彈性、以及市場動態

非監督式學習

- 需要無標記(Labeled)的數據
- 可用於分群分析 (分類)
- 用於在數據中的尋找既定模式的情況

- 集群分析 (K-means clustering)
- 混和模型 (Gaussian mixture)



- 判斷信用交易、保險金融等活動是否異常 (詐欺)

半監督式學習

- 結合有標記與未標記的數據
- 出於有標記數據的獲取困難或成本較高
- 可以有效提升模型的正確率

- 先利用監督式學習訓練，之後在加上無標記的數據。



- 可用於自然語言處理，以強化準確性

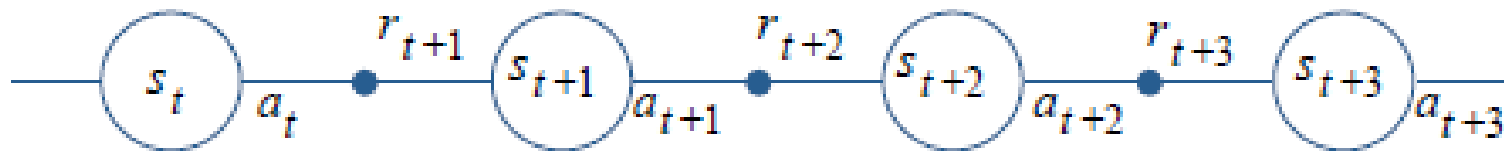
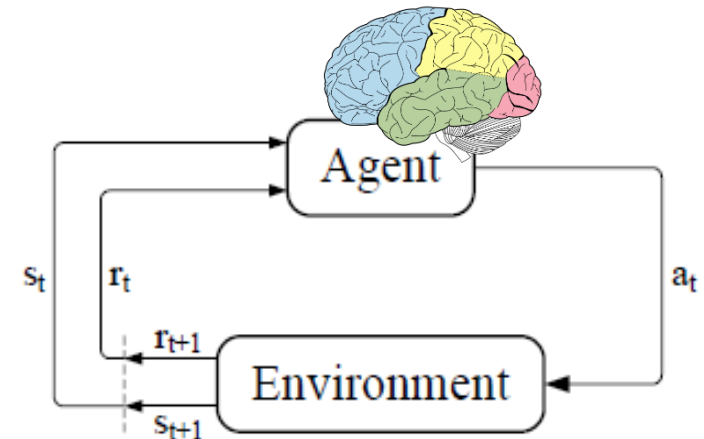
Reinforcement Learning

- No raw data is given as input.
- Reinforcement learning algorithm have to figures out the situation on their own.
- Frequently used for robotics, gaming, and navigation.
- With reinforcement learning, the algorithm discovers through **trial and error** which actions **yield the most significant rewards**.
- Three components:
 - **Agent**: the learner or decision maker.
 - **Environment**: everything the agent interacts with.
 - **Actions**: what the agent can do.

Interaction between Agent and Environment



- Agent interacts at discrete time steps $t = 0, 1, 2, \dots$
- Observe state $s_t \in S$
- Selects action $a_t \in A(s_t)$
- Obtains immediate reward $r_{t+1} \in R$
- Observe resulting state s_{t+1}



Train an agent which can adapt to the environment.

How To Choose Action?

- Agent adopts “strategy” to choose actions.
- Two methods:
 - Exploration(探索): choose actions randomly
 - Exploitation(利用): choose best actions from past experience (Q-Table)

Q-table initialised at zero

	UP	DOWN	LEFT	RIGHT
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0

After few episodes

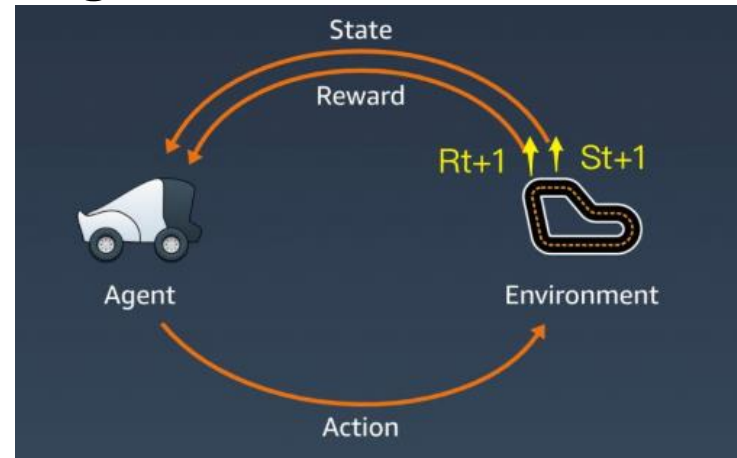
	UP	DOWN	LEFT	RIGHT
0	0	0	0	0
1	0	0	0	0
2	0	2.25	2.25	0
3	0	0	5	0
4	0	0	0	0
5	0	0	0	0
6	0	5	0	0
7	0	0	2.25	0
8	0	0	0	0

Eventually

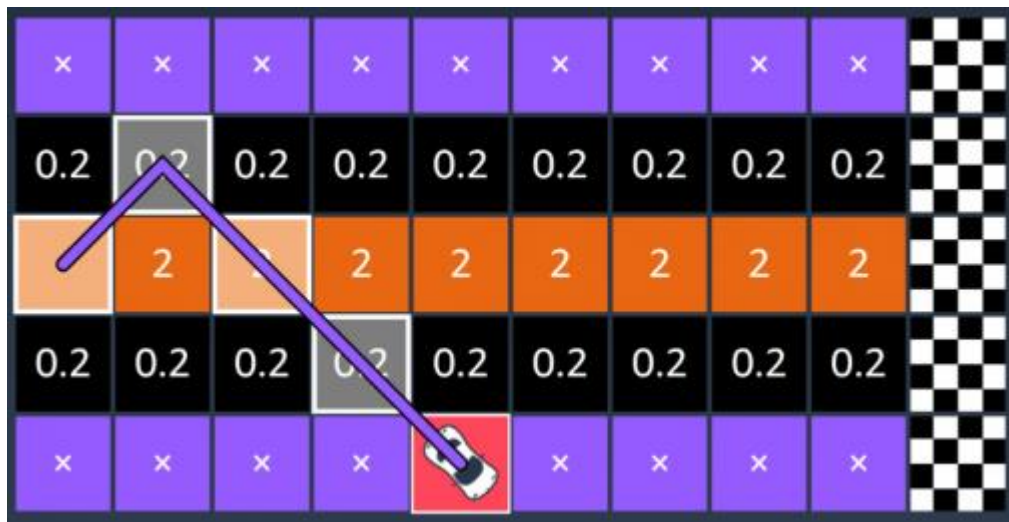
	UP	DOWN	LEFT	RIGHT
0	0	0	0.45	0
1	0	1.01	0	0
2	0	2.25	2.25	0
3	0	0	5	0
4	0	0	0	0
5	0	0	0	0
6	0	5	0	0
7	0	0	2.25	0
8	0	0	0	0

Reinforcement Learning Example

- Mission: Train the racing car to get used to the track.
 - Racing car: agent
 - Track: Environment
 - Policy: Action
- Maximize reward function.



First Try:



-
- The diagram illustrates the Reinforcement Learning loop. On the left is the **Agent**, represented by a small white car icon. On the right is the **Environment**, represented by a track icon. Three curved orange arrows show the flow of information: a top arrow labeled **State** from Environment to Agent, a middle arrow labeled **Reward** from Environment to Agent, and a bottom arrow labeled **Action** from Agent to Environment. To the right of the Environment icon, the text R_{t+1} and S_{t+1} is shown with two upward-pointing yellow arrows, indicating the next state and reward.

[illegible]

Reinforcement Learning Example

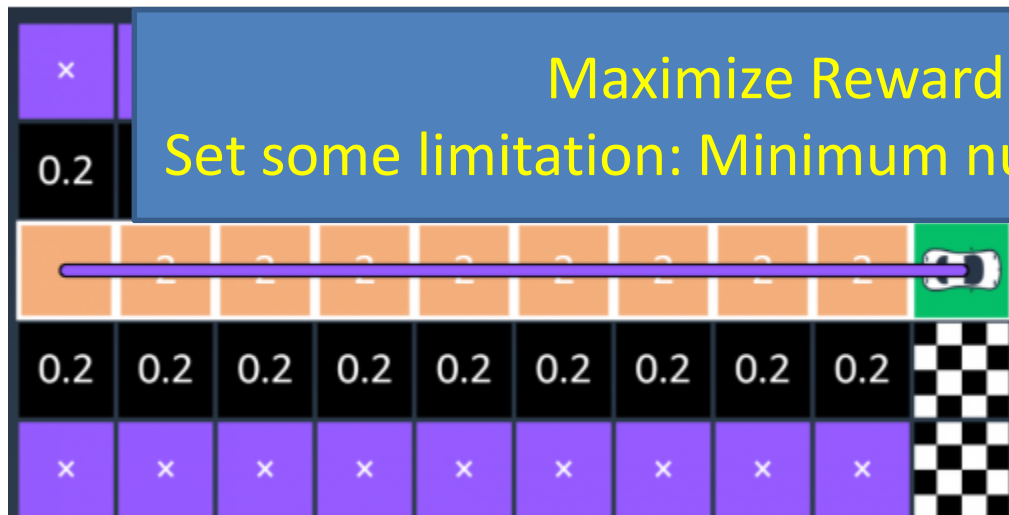
- Mission: Train the racing car to get used to the track.
 - Racing car: agent
 - Track: Environment
 - Policy: Action



- Maximize

Maximize Reward:
Try to go every lattice. =>Waste too much time.

Third

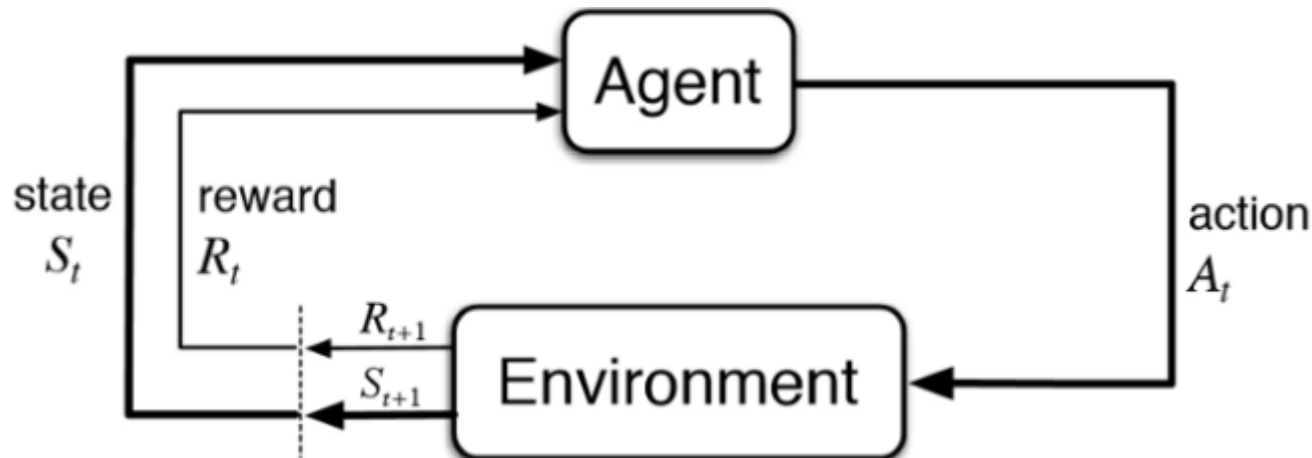


Maximize Reward:
Set some limitation: Minimum number of steps.



Objective

- The agent take actions that **maximize the expected reward** over a given measure of time.
- The agent will reach the goal much quicker by following a good policy.
- The purpose of reinforcement learning is to **learn the best plan**.



Generation

- Generate structured and complex data ex: image data

Image Generation

<https://arxiv.org/abs/1809.11096>

<https://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>

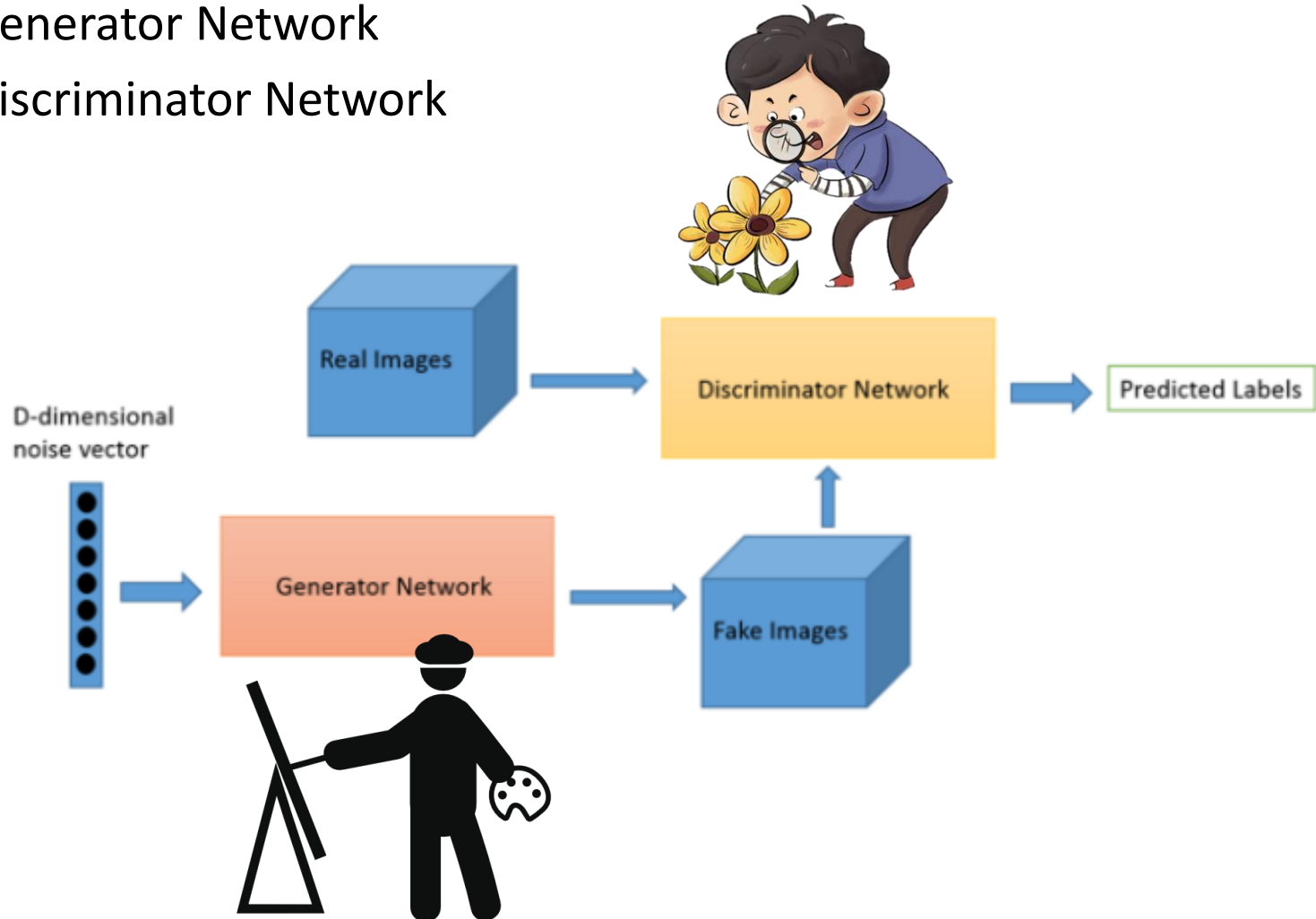


Generative Models: variational auto-encoder (VAE), generative adversarial network (GAN), Flow-based generative model, etc.

Generative Adversarial Network



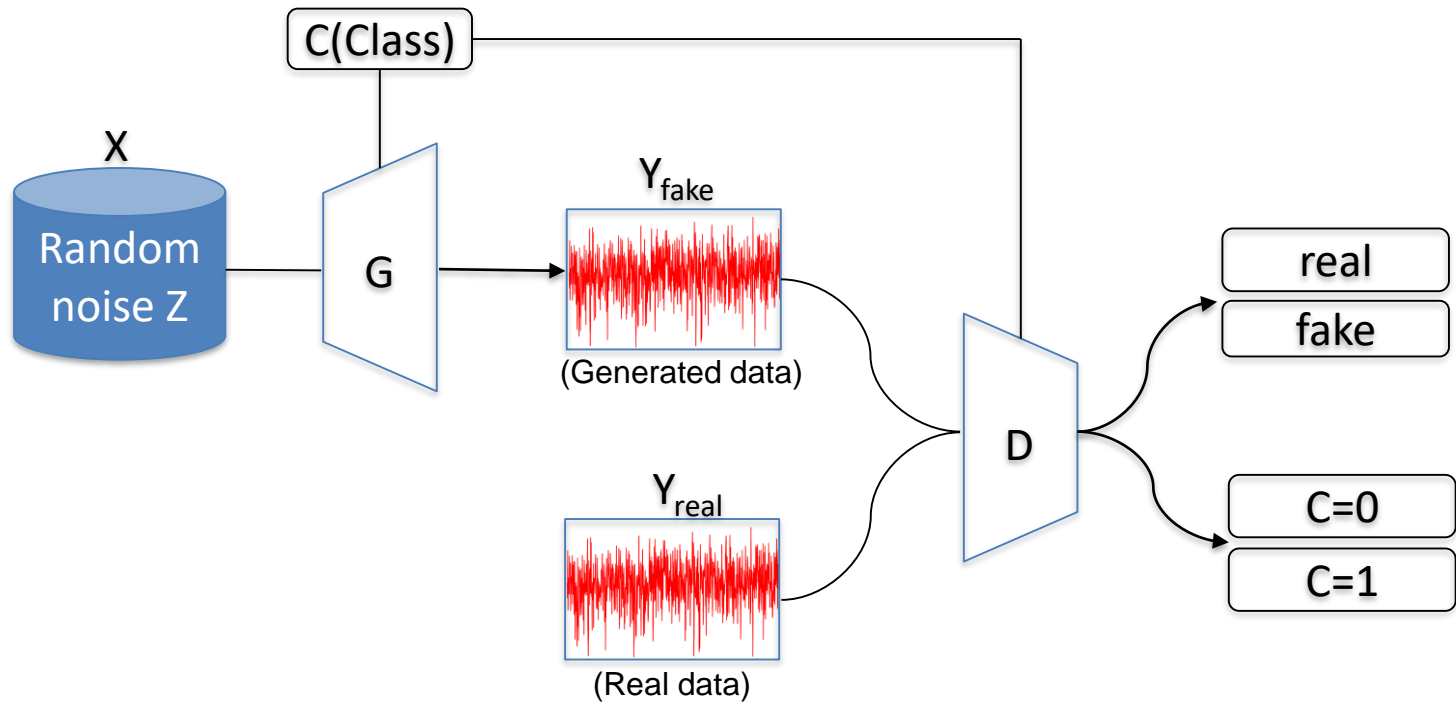
- GAN:
 - Generator Network
 - Discriminator Network



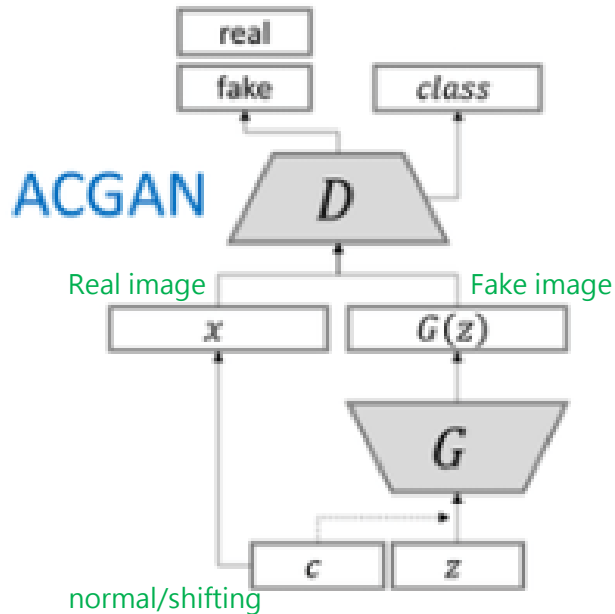
Auxiliary Classifier GAN (1/2)



- ACGAN: add classification into GAN.



Auxiliary Classifier GAN (2/2)



The normal data generated by ACGAN.

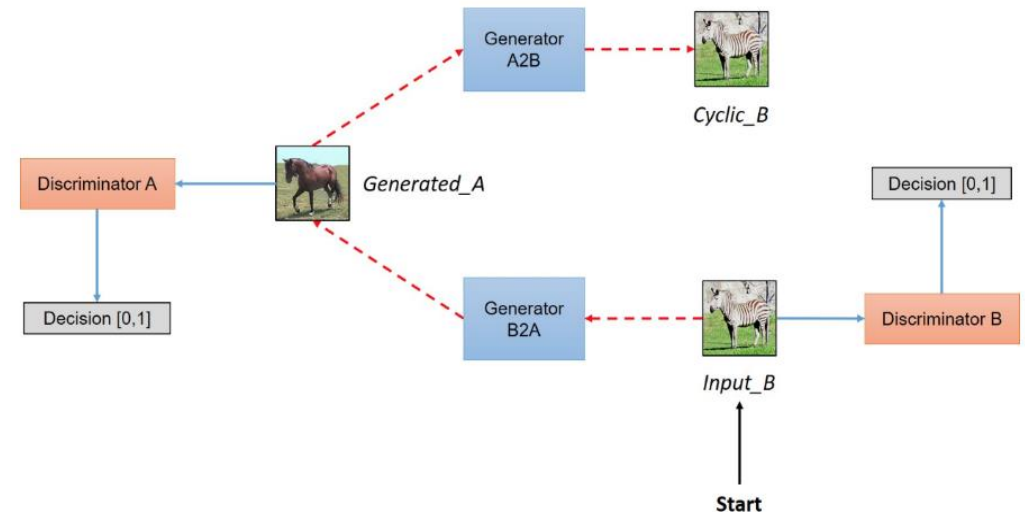
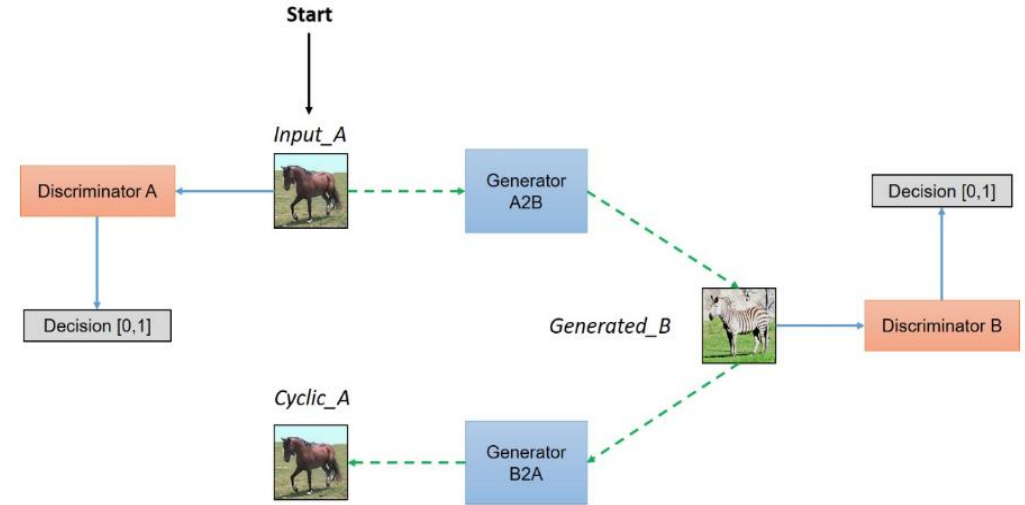


The defect data generated by ACGAN.

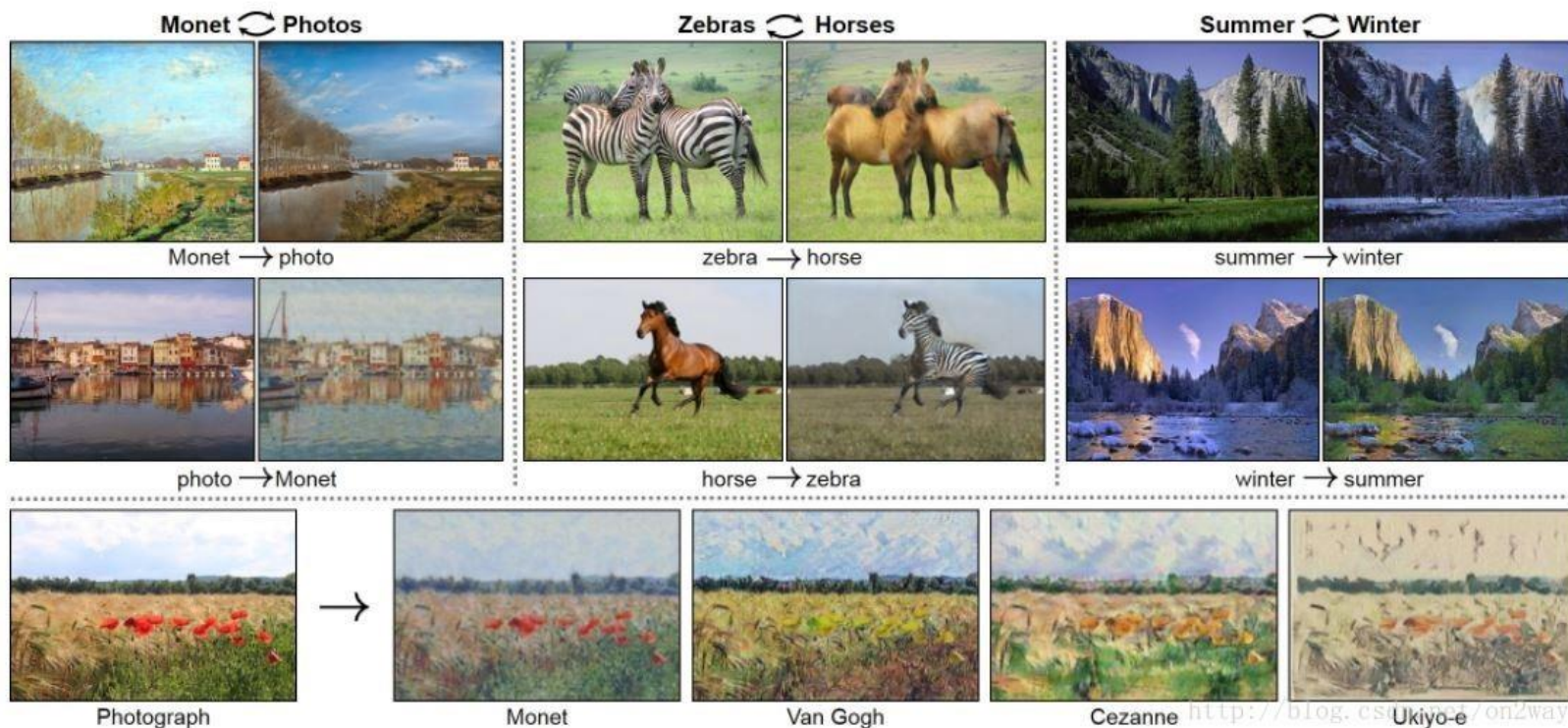


Cycle GAN (1/2)

- Cycle GAN is a neural network that can be used to generate different style/material transformations. Applied to image-to-image transformation
- Horse \Leftrightarrow Zebra

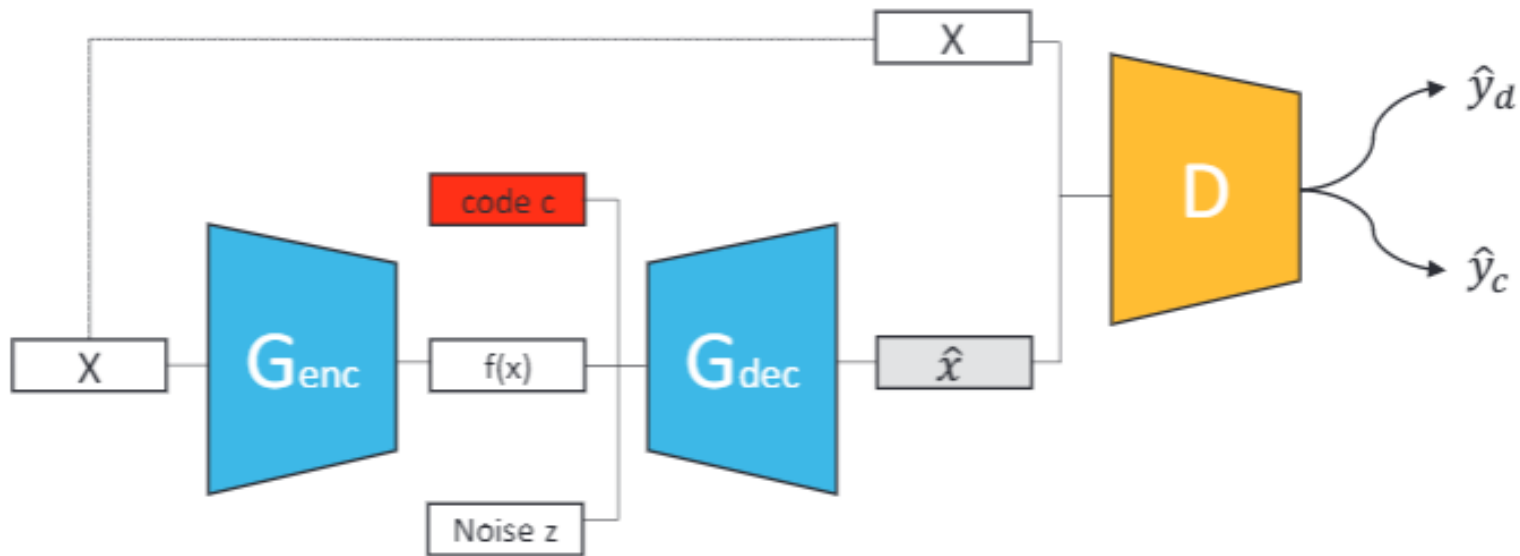


Cycle GAN (2/2)



DRGAN (1/2)

- Disentangled Representation Learning GAN (DR-GAN)
- Add pose code in GAN.
- DRGAN can merge two faces or turn the face away.



DRGAN (2/2)

