視覺機器學習模型

授課老師: 楊景明

同意書 & 問卷Time

- 同意與否皆不會影響上課權益
- 學習記錄在未來有機會用於資料分析與系統改善(皆會去識別化)
- 同意參與以及完成兩次問卷填寫會有小禮物
- 聯絡電話可以填電話或email
- 正本會由老師保存, 同學若有需要可以拍照留存
- 問卷結果不會影響成績
- 問卷連結
  :https://docs.google.com/forms/d/e/1FAIpQLSfH9LsFxtko9HM-dFU1IYGS75EZ0tv4ZH3FabETnMcfOZeHgA/viewform?usp=sf_link

為什麼電腦視覺很難？

為什麼電腦視覺很難？

物件

類別

深度

…

相機和感測器性能

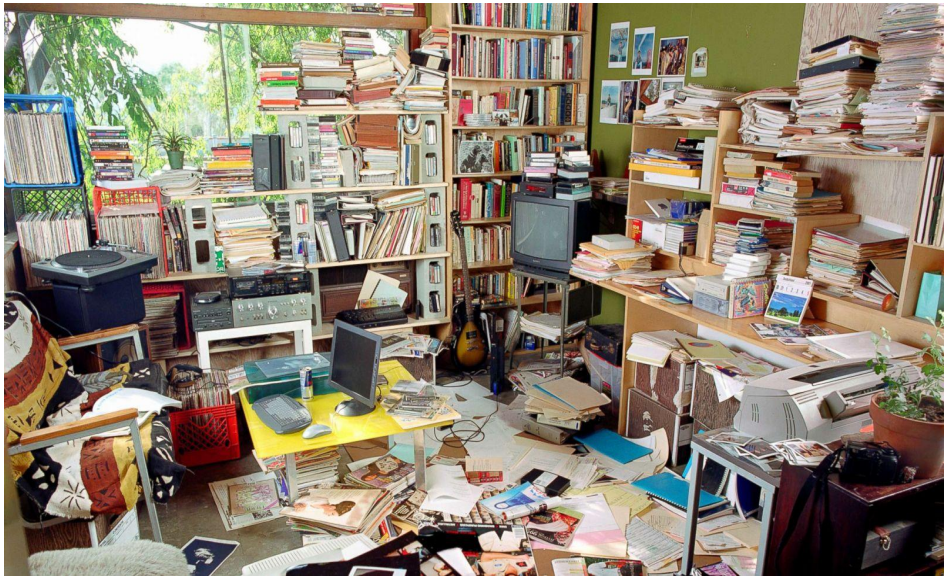
角度

亮度

規模

動作

雜亂(Clutter)

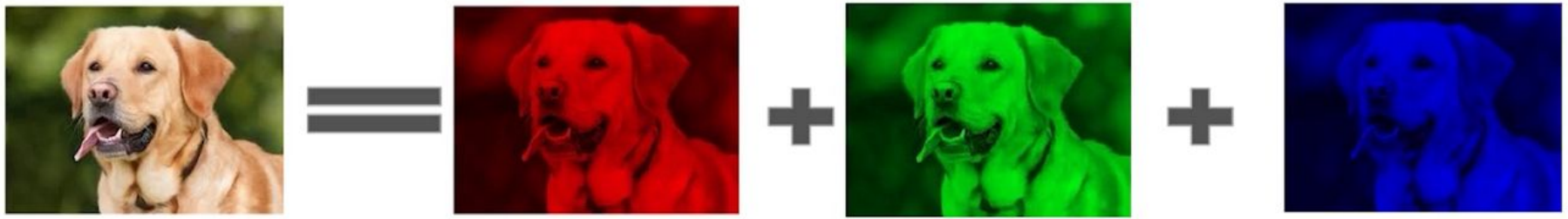


型態

視錯覺 (optical illusions)

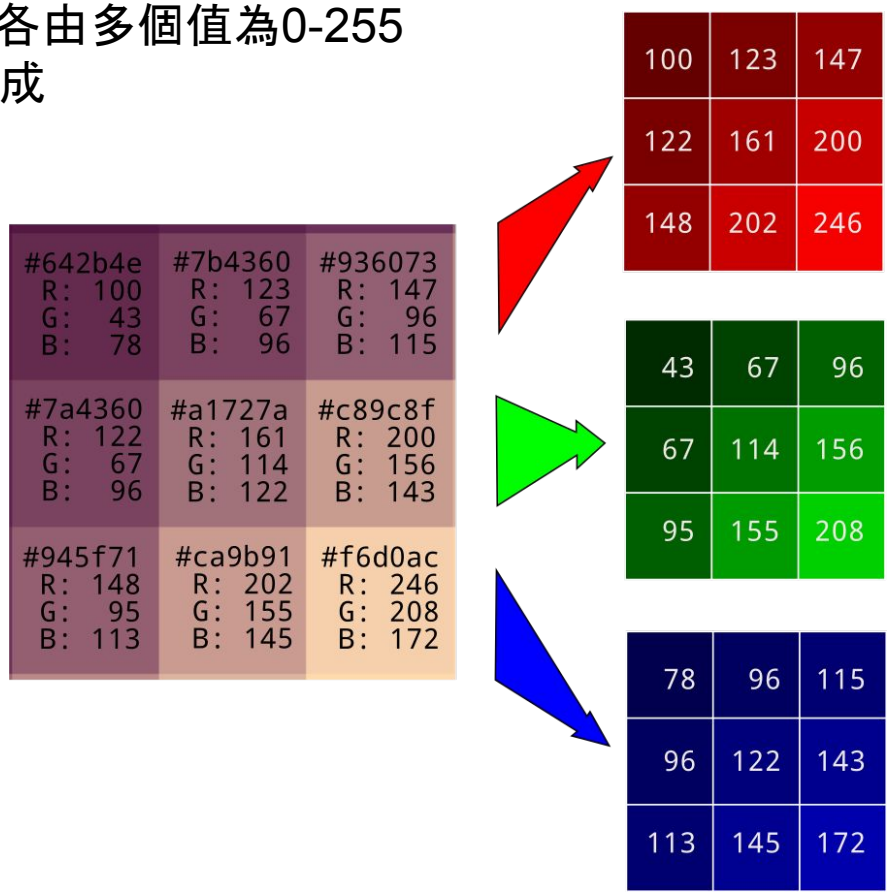
什麼是image？

電腦眼中的圖



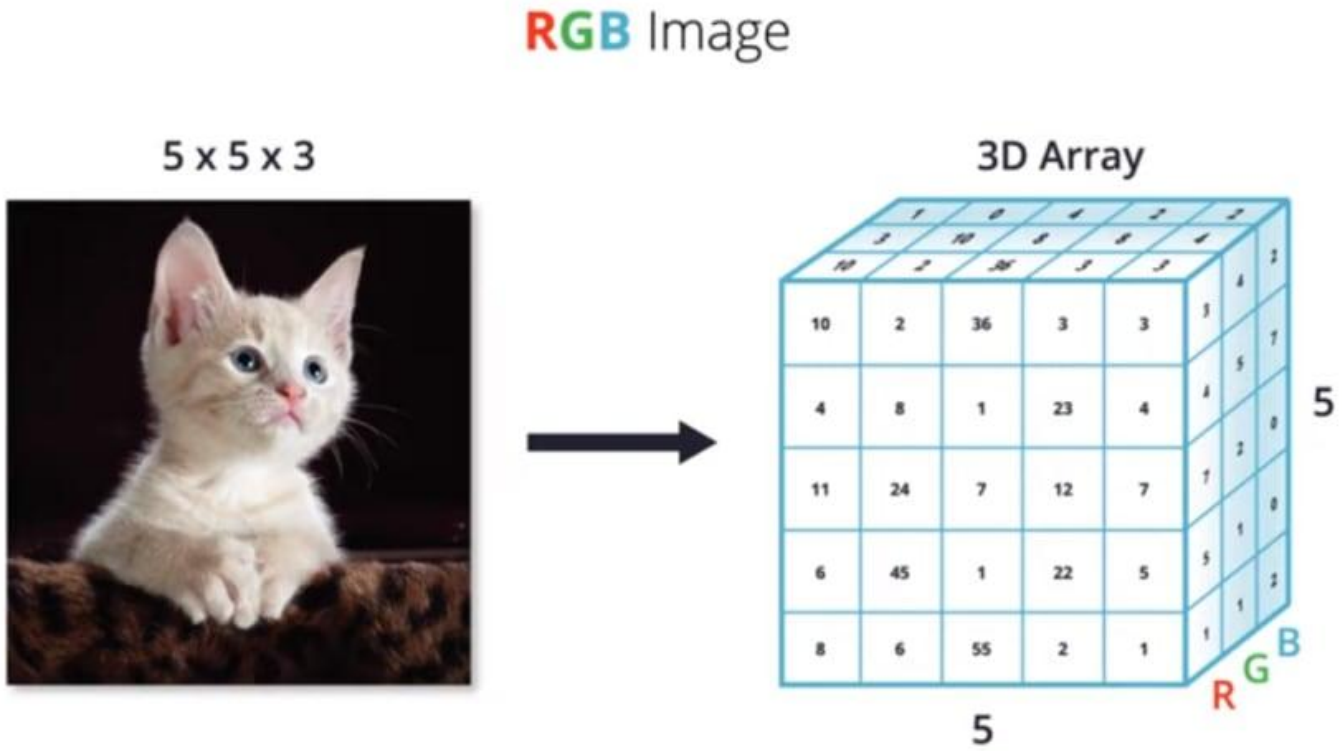
Source: https://www.analyticsvidhya.com/blog/2021/03/grayscale-and-rgb-format-for-storing-images/

RGB頻道(channel) 各由多個值為0-255之間的像素(pixel)組成



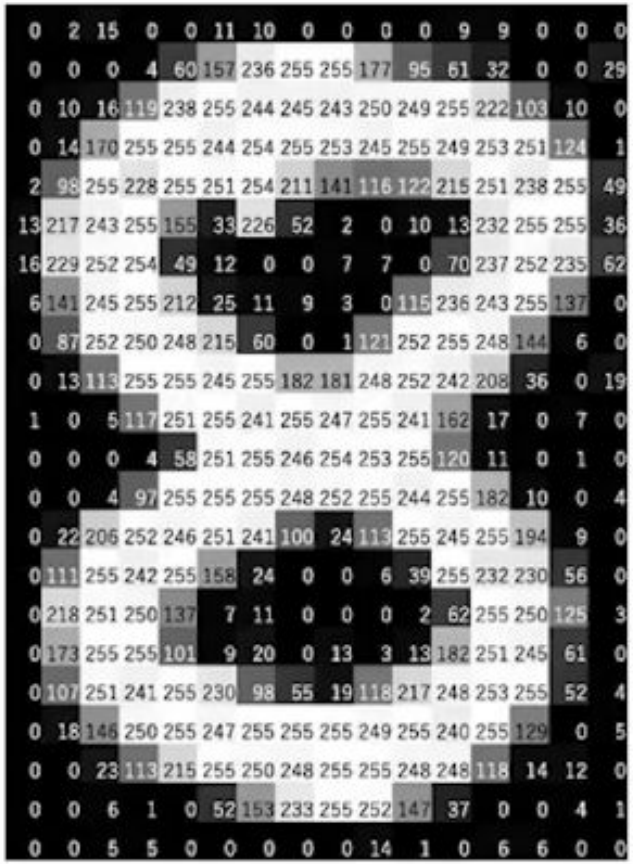
Source: https://www.kdnuggets.com/2020/01/convert-picture-numbers.html

圖像格式



Source: https://dev.to/sandeepbalachandran/machine-learning-going-furthur-with-cnn-part-2-41km

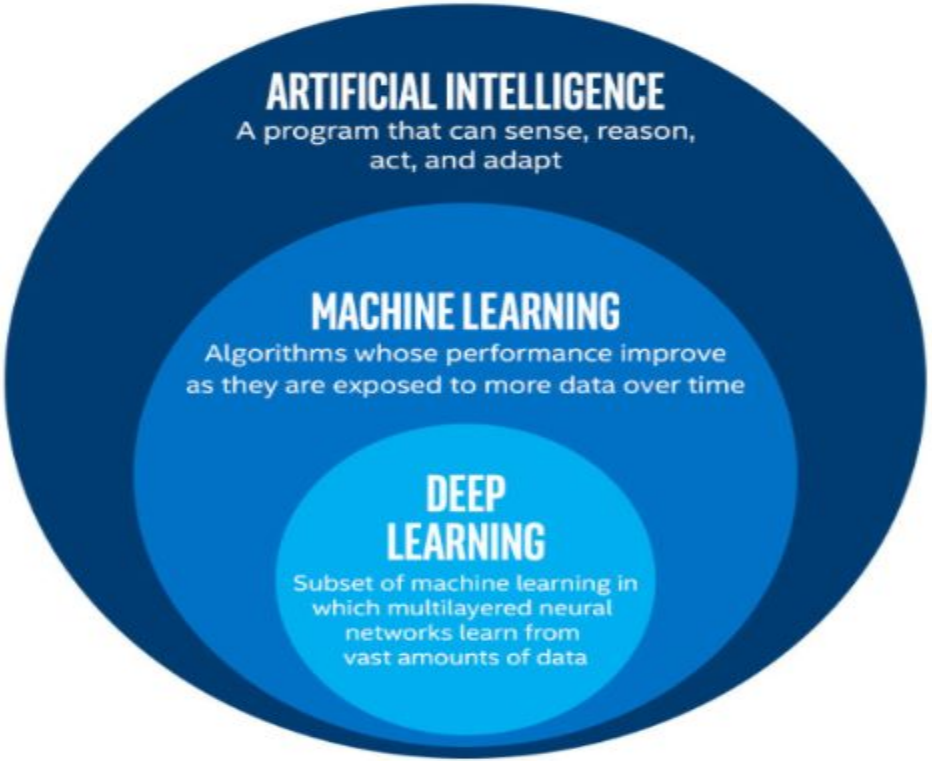
灰階圖片



Source: https://www.analyticsvidhya.com/blog/2021/03/grayscale-and-rgb-format-for-storing-images/

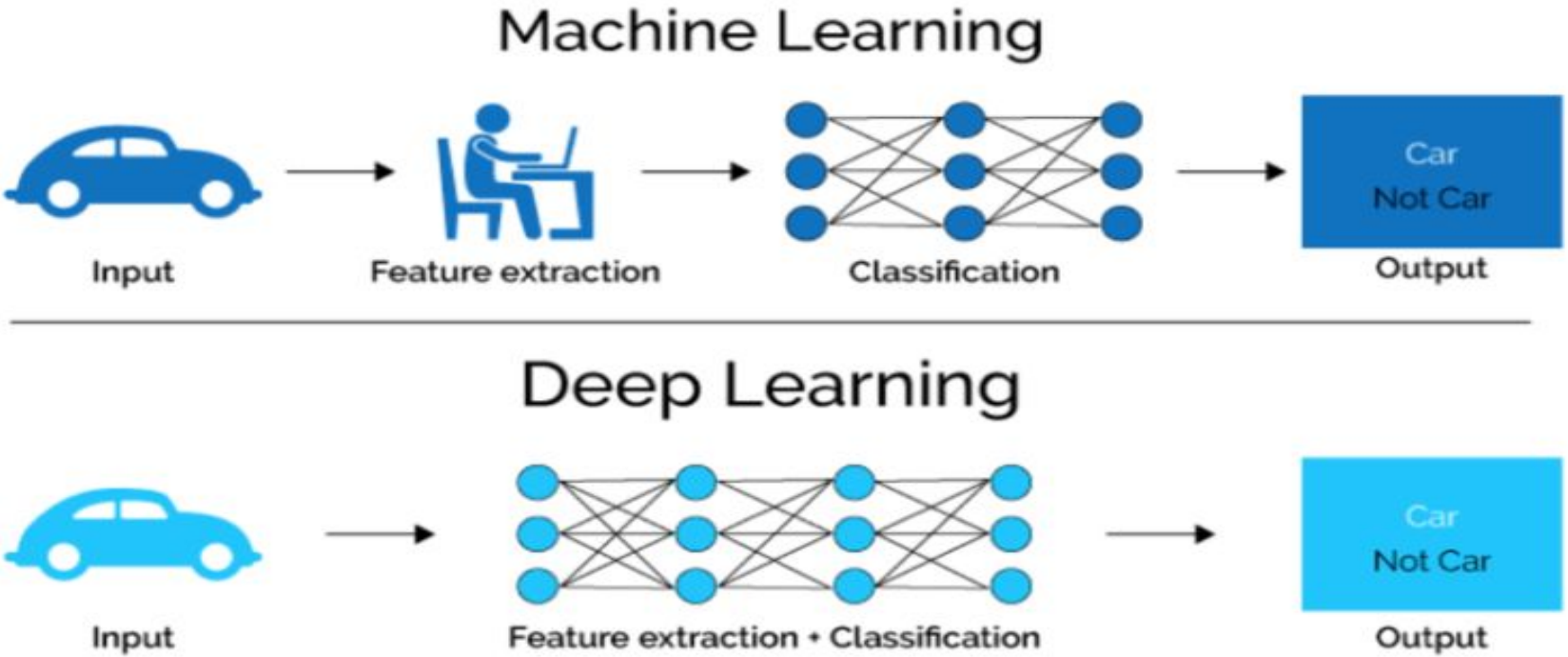
Machine Learning x Computer Vision

What is Machine Learning?



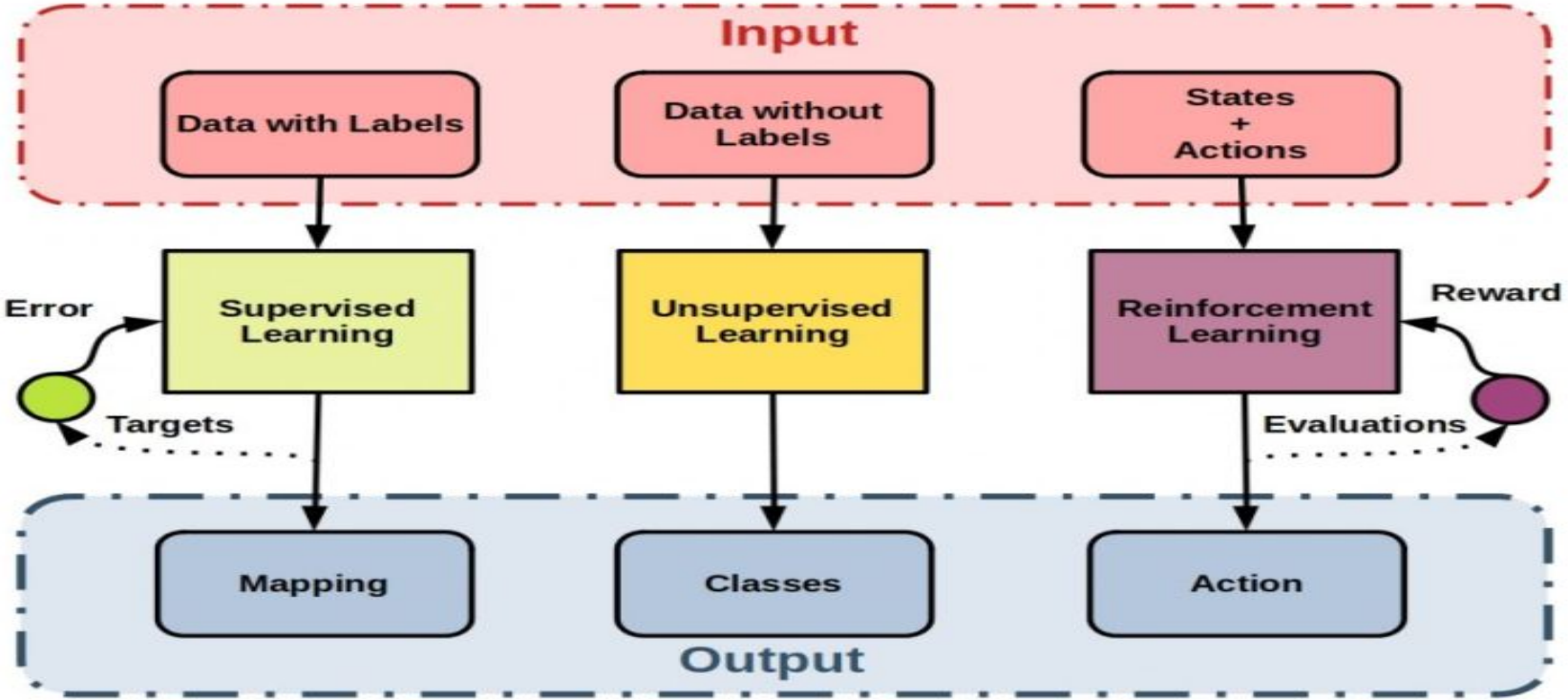
source:https://www.aifintechtw.com/article?article=40

Difference between ML and DL



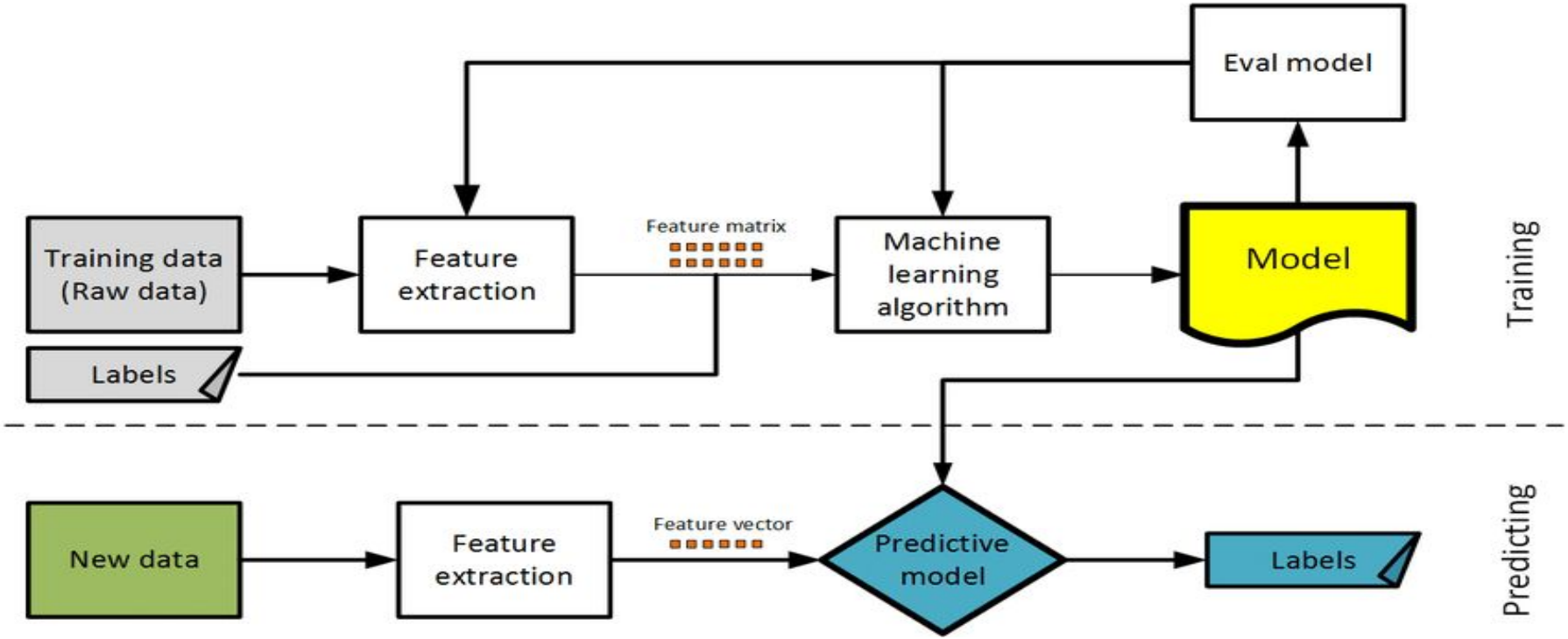
source:https://tw.alphacamp.co/blog/machine-learning-and-deep-learning-intro

Difference tasks of ML



source:https://tw.alphacamp.co/blog/machine-learning-and-deep-learning-intro

ML flowchart

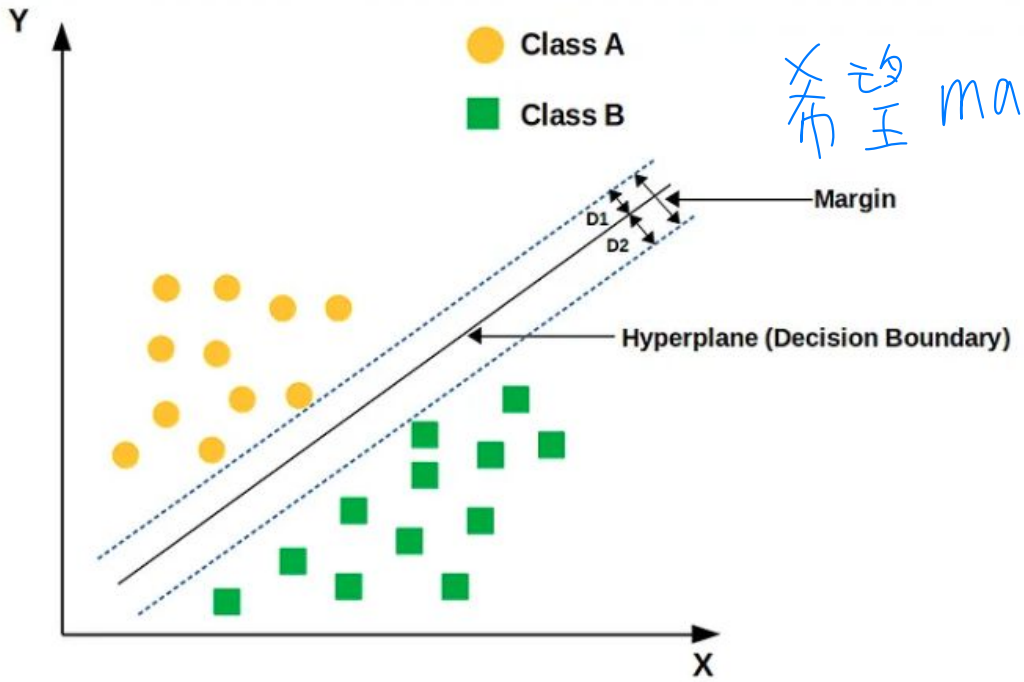


source:source:https://tw.alphacamp.co/blog/machine-learning-and-deep-learning-intro

1.SVM Support Vector machine

- 一種監督式機器學習演算法
- 主要用於分類任務
- 目標是找到一條"最佳超平面"來區分不同類別的數據
- 同時最大化不同類別之間的數據點距離超平面的間隔(Margin)，確保加入新的資料後的維持更高的準確率

二元分類

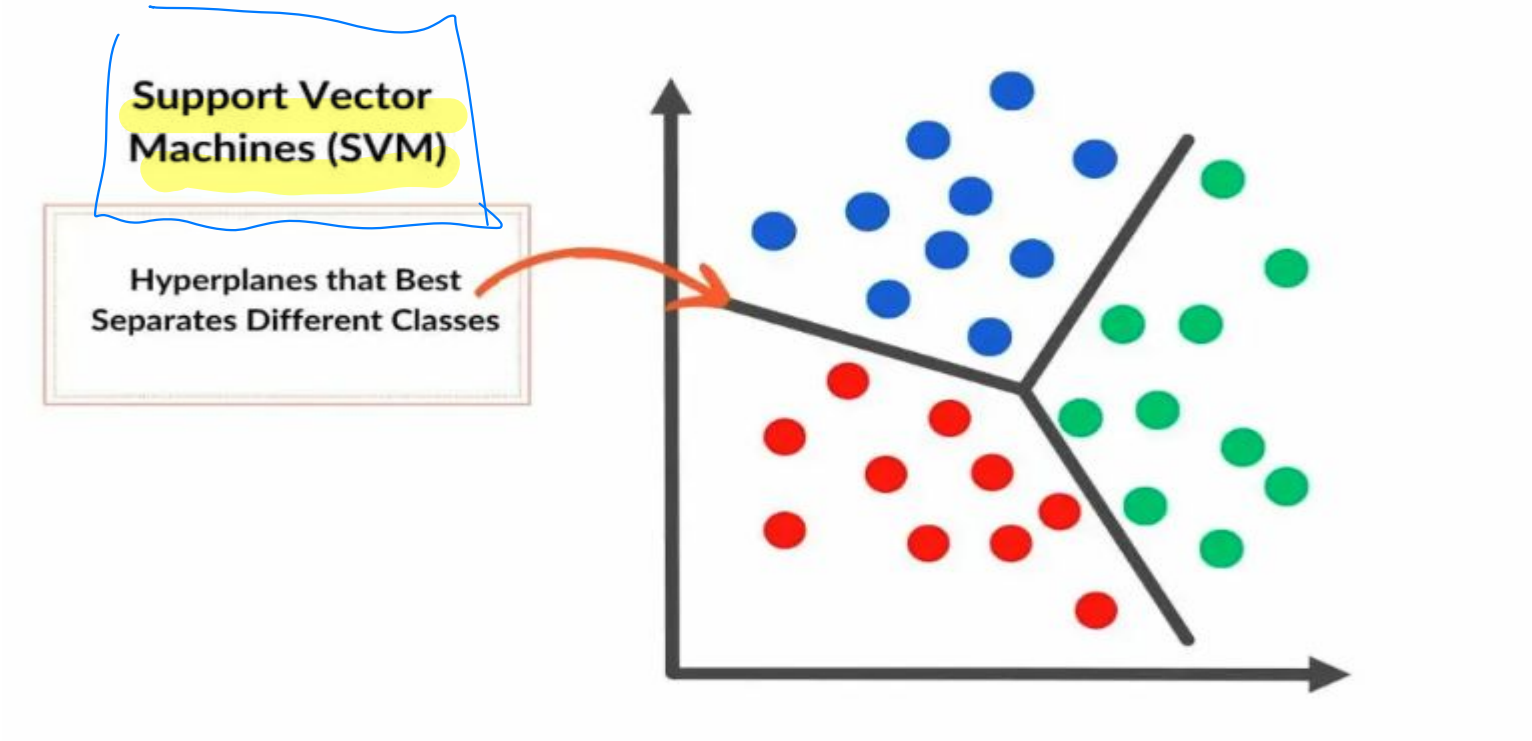


間隔

希望 margin 越大越好

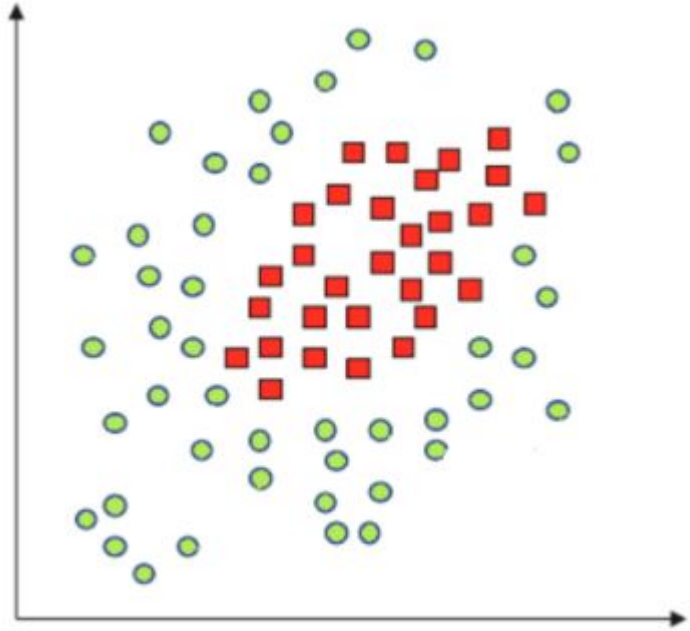
source:https://ashish-mehta.medium.com/support-vector-machine-svm-algorithm-for-machine-learning-350fe9139a52

多類別分類



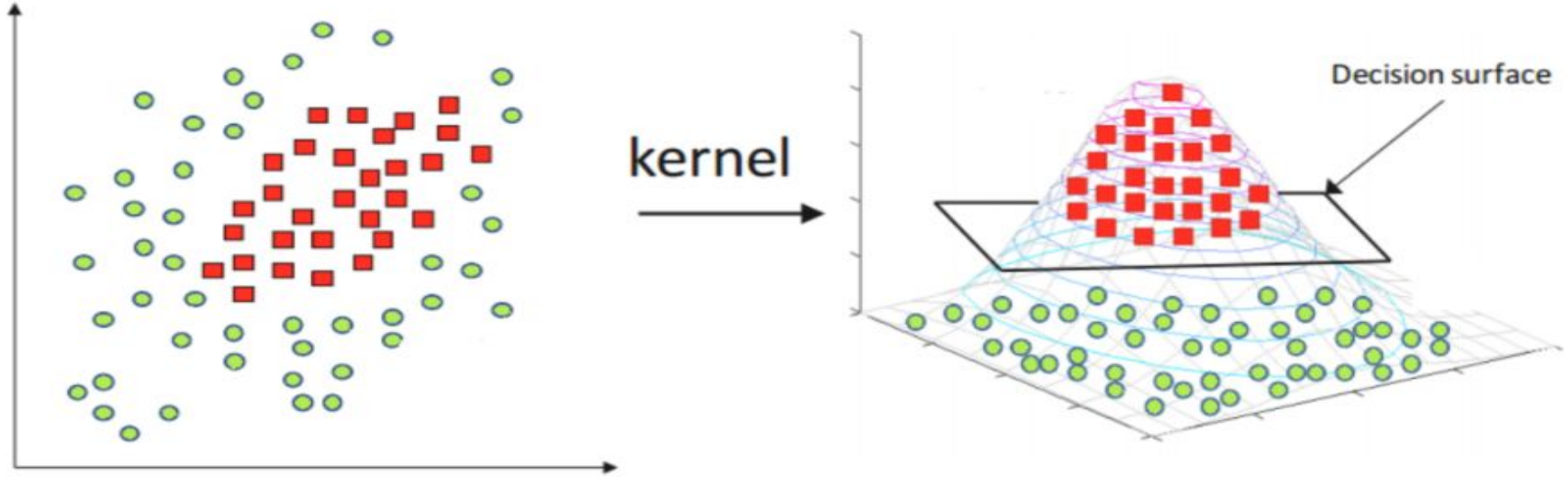
source:https://spotintelligence.com/2024/05/06/support-vector-machines-svm/

若資料無法用一條線切開呢?



source:https://tangshusen.me/2018/10/27/SVM/

透過Kernel Trick將資料投影到更高維的空間



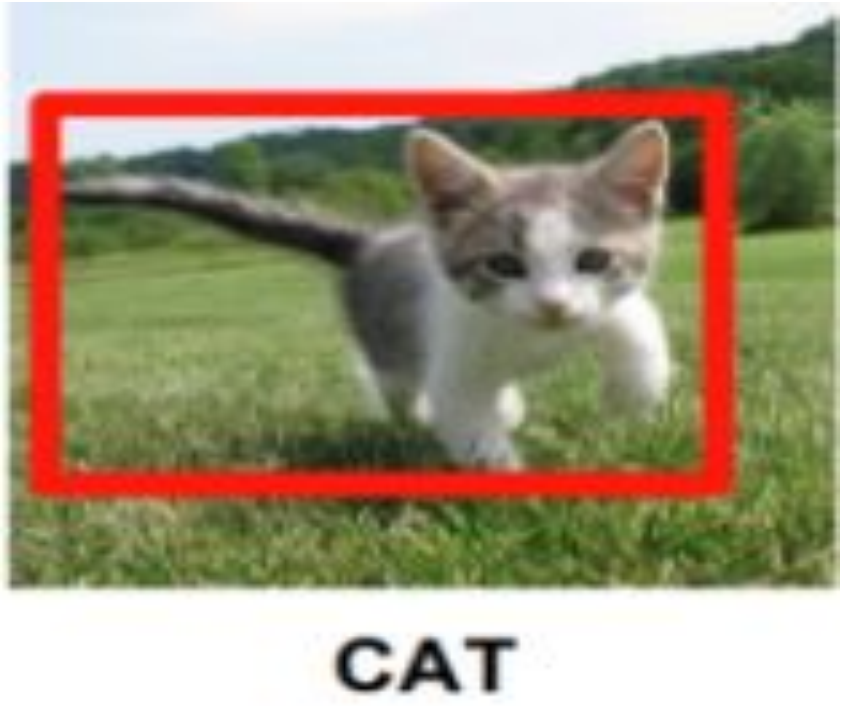
source:https://tangshusen.me/2018/10/27/SVM/

SVM X Computer Vision

- 圖像分類
- 物件偵測
- 語義分割

SVM X 物件偵測

- SVM適合分類任務
- 每個格子或偵測匡分成是該物體或不是

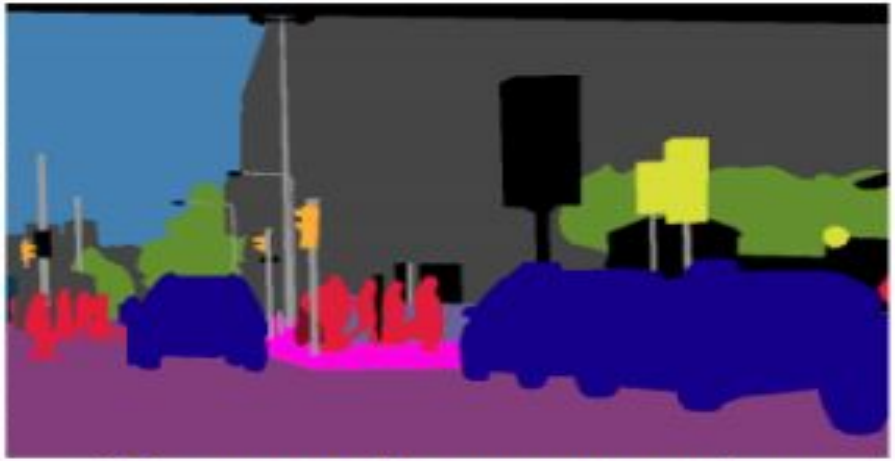


source:https://ithelp.ithome.com.tw/articles/10192950

SVM X 語義分割

同理, SVM也適合把圖相似的部分做切分

(a) image



(b) semantic segmentation

source:https://medium.com/ching-i/%E5%BD%B1%E5%83%8F%E5%88%86%E5%89%B2-image-segmentation-%E8%AA%9E%E7%BE%A9%E5%88%86%E5%89%B2-semantic-segmentation-1-53a1dde9ed92

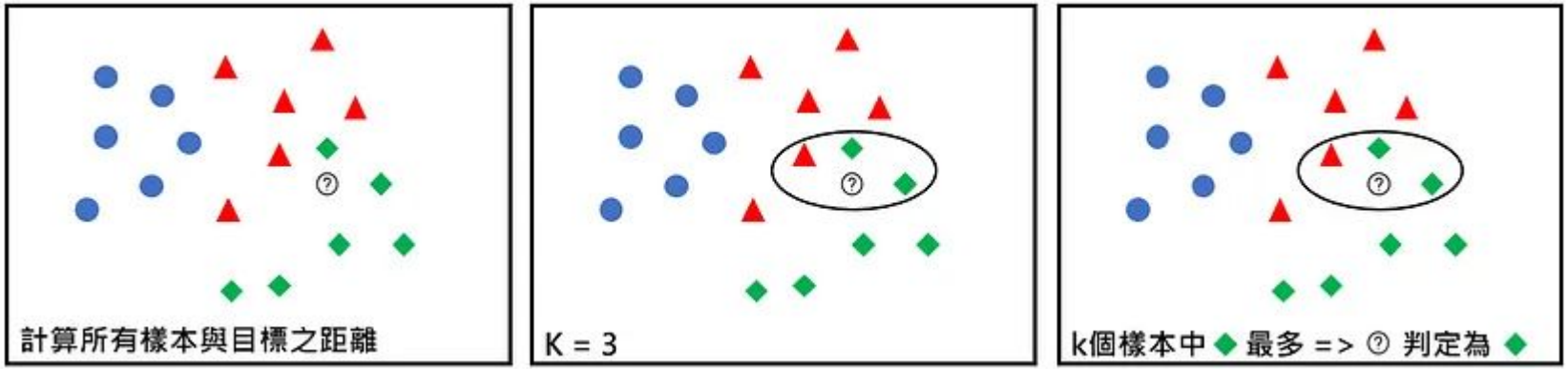
Disadvantages of SVM

- 不適合大資料集
  - SVM 訓練過程在大型資料集上非常耗時
  - 因為它需要花很多時間計算最佳超平面該如何切分
- 對非線性資料的處理有限
  - 雖然Kernel Trick能夠擴展 SVM 處理非線性問題的能力
  - 但選擇合適的Kernel function以及調整參數可能非常困難
- 對於多分類問題較複雜
  - SVM 原本是為二分類問題設計的 ∵ SVM 是為 二分類 設計的
  - 當應用於多分類問題時，需要進行“一對一”或“一對多”的轉換
  - 增加了計算的複雜性和成本

2.KNN

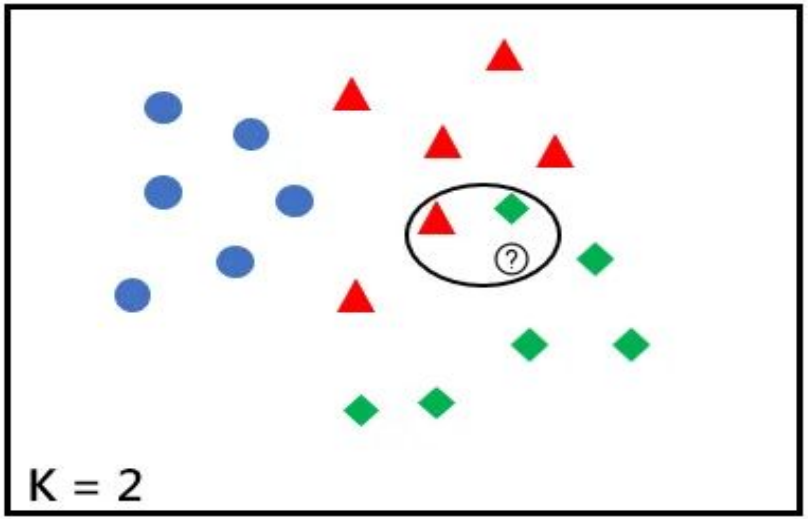
- 一種監督式學習演算法
- 主要用於分類任務
- 核心概念是基於距離來判斷資料的類別

基本三步驟



source:https://medium.com/@SCU.Datascientist/python%E5%AD%B8%E7%BF%92%E7%AD%86%E8%A8%98-knn-k-nearest-neighbor-531a95336f71

若結果為各半怎麼辦?



為避免這個情況,K通常會設為奇數

source:https://medium.com/@SCU.Datascientist/python%E5%AD%B8%E7%BF%92%E7%AD%86%E8%A8%98-knn-k-nearest-neighbor-531a95336f71

如何計算距離?

- 歐基里德距離 (EUCLIDEAN DISTANCE)
- 曼哈頓距離 (MANHATTAN DISTANCE)
- 明氏距離 (MINKOWSKI DISTANCE)

$$D = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

$$D = |x_1 - y_1| + |x_2 - y_2| + \ldots + |x_n - y_n|$$

$$= \sum_{i=1}^{n} |x_i - y_i|$$

$$D = \left(\sum_{i=1} n|x_i - y_i|^p\right)^{\frac{1}{p}}$$

P = 1 → 曼哈頓距離（L1 norm）

P = 2 → 歐幾里德距離 （L2 norm）

source:https://iaic.nccu.edu.tw/column-articles/26

KNN X Computer Vision

- 圖像分類
- 物件偵測
- 圖像分割

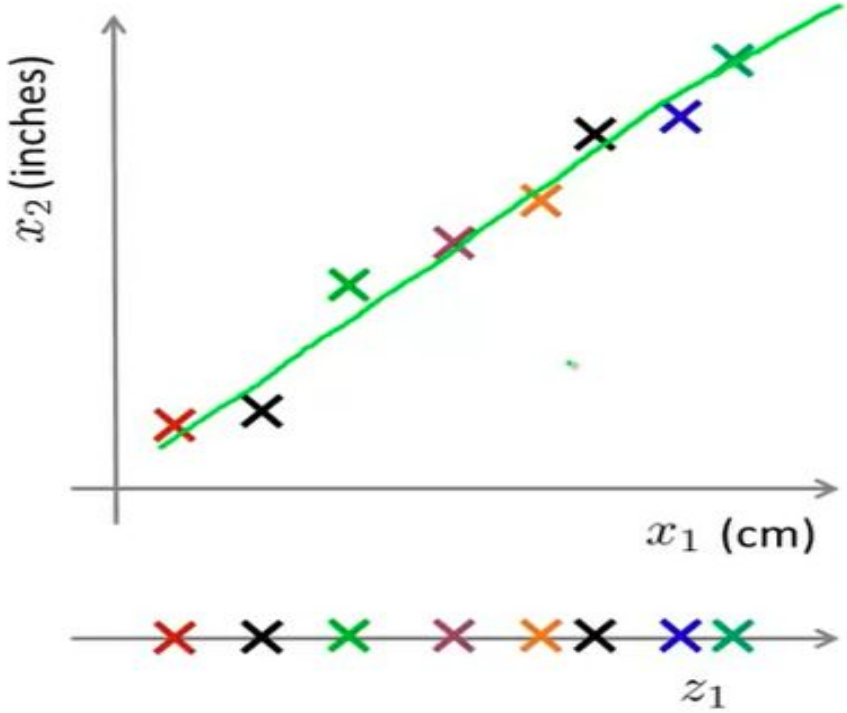
Disadvantages of KNN

- 計算效率低
  - k-NN 在進行預測時需要計算新樣本與所有樣本之間的距離
  - 這在大型資料集會非常慢，隨著資料集繼續增大，計算成本會顯著增加
- 在高維度資料下表現不佳
  - 因為高維度空間中距離的概念變得不再有效，這導致分類效果下降
- 無法處理非線性關係
  - k-NN 根據距離來進行判斷，但在許多圖像應用中，圖像之間的關聯往往是非線性的

PCA

- 是一種降維技術
- 主要用於在保持資料主要信息的同時，減少特徵數量
- 核心思想是將高維資料投影到較低維度的空間，並且這些新空間(稱為主成分)是資料中變異性最大的方向

將二維資料降成一維



Video

https://leemeng.tw/images/pca/DotProductAsProjectTo1D.mp4

PCA X Computer Vision

1.圖像降維
2.圖像去噪

PCA X 圖像降維

- 某些任務不需要考慮圖像色彩
  ex.邊緣偵測
- 直接用灰階圖像可以減少計算量，加快訓練速度

PCA X 圖像去噪

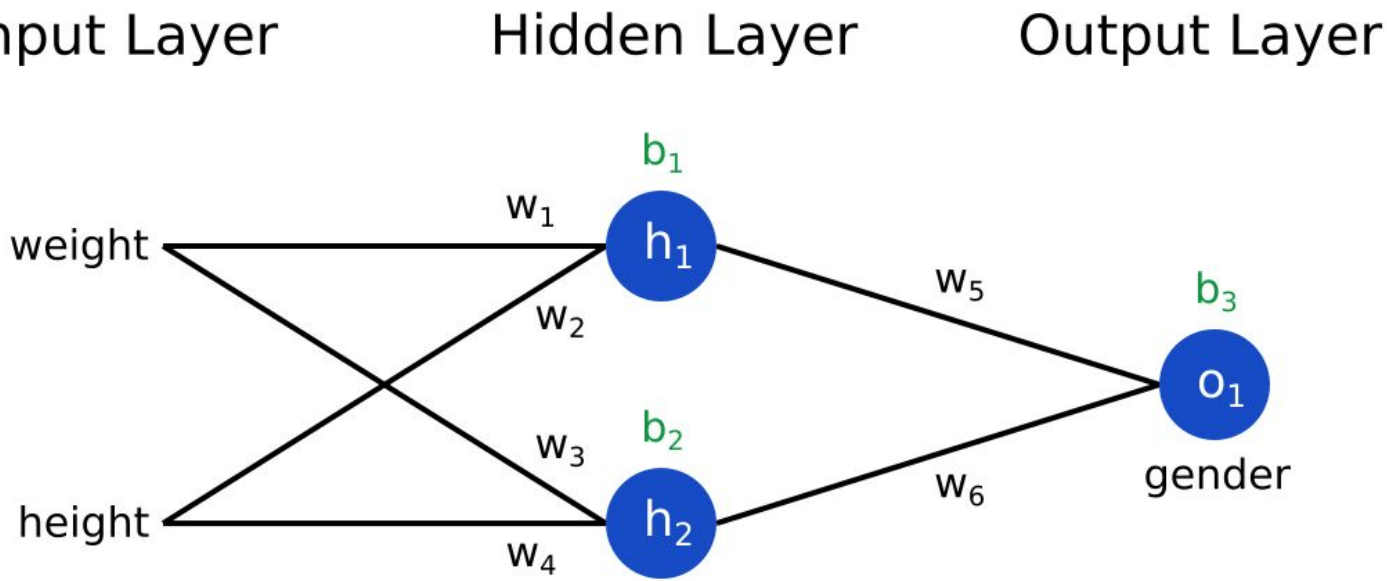
夜間視野不佳或是畫質不好的畫面可以透過圖像降噪來得到清晰的圖像

Disadvantages of PCA

- 對非線性資料的處理有限
  - PCA 是一種線性降維技術，假設資料中的變異性主要沿著直線方向分佈
  - 因此，對於那些具有非線性結構的數據，PCA 可能無法有效捕捉數據的關鍵特徵
- 訊息遺失
  - PCA 只保留了資料中變異性最大的方向
  - 雖然在降維時減少了資料的維度，但可能會損失一些重要的細節與特徵
  - 可能在某些需要細節判斷的任務表現不佳

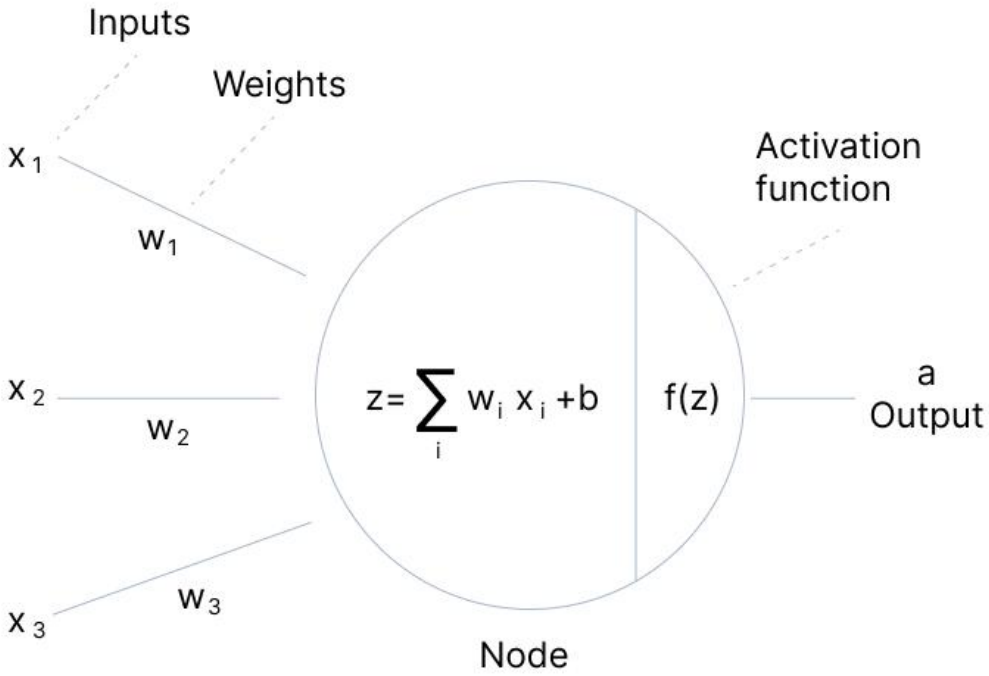
Deep Learning x Computer Vision

A Simple Neural Network



Source: https://victorzhou.com/blog/intro-to-neural-networks/

For each node



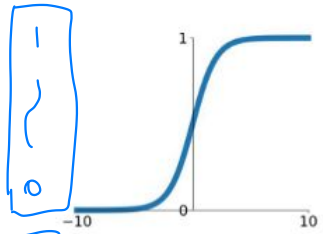
Source: https://www.v7labs.com/blog/neural-networks-activation-functions

Activation function

Introduce non-linearity

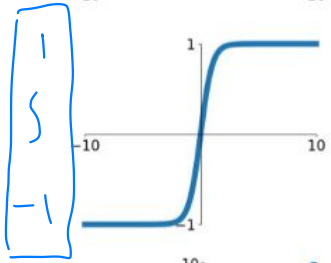
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



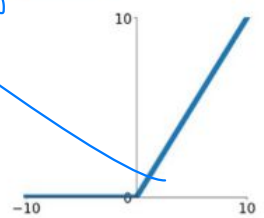
tanh

$$\tanh(x)$$



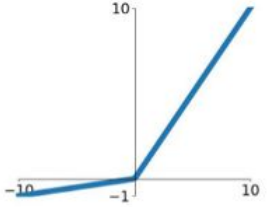
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

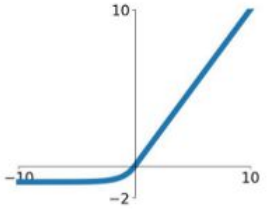


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$

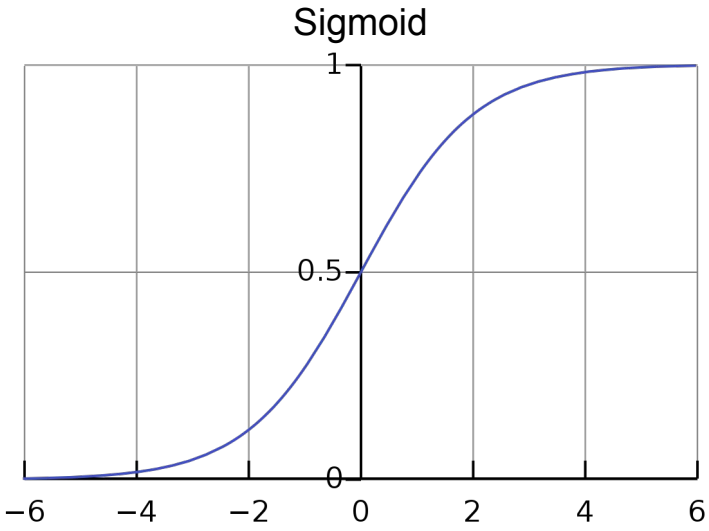


Source: https://medium.com/@shrutijadon/survey-on-activation-functions-for-deep-learning-9689331ba092

Output層的輸出

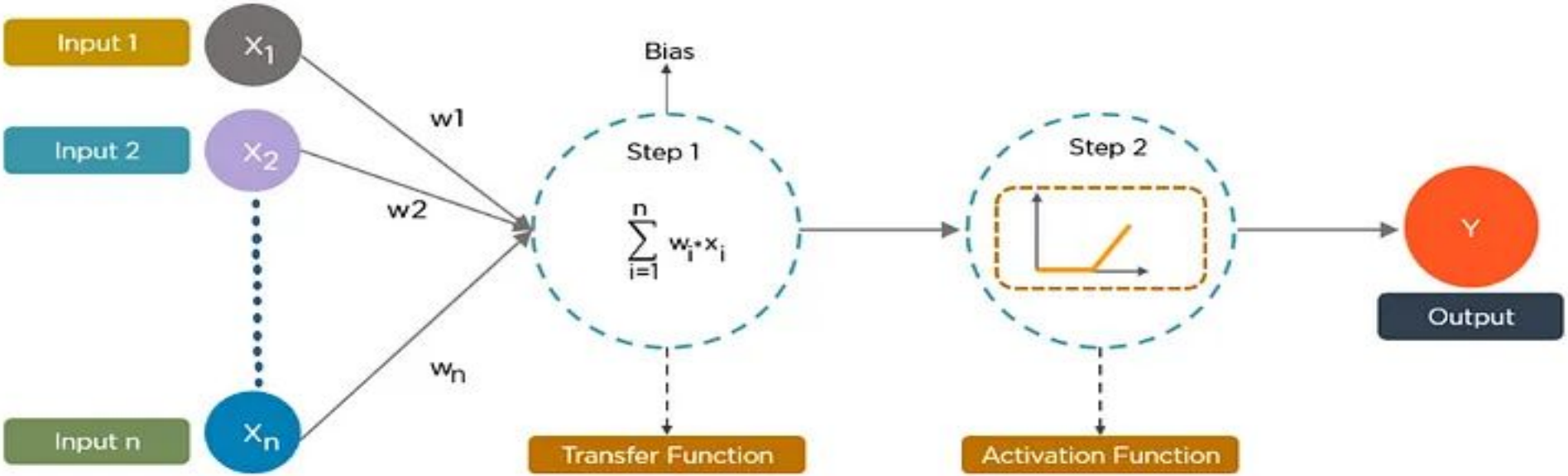
For classification task

- 二元分類 -> Sigmoid
- 多類別分類 -> Softmax



Softmax

Forward Propagation



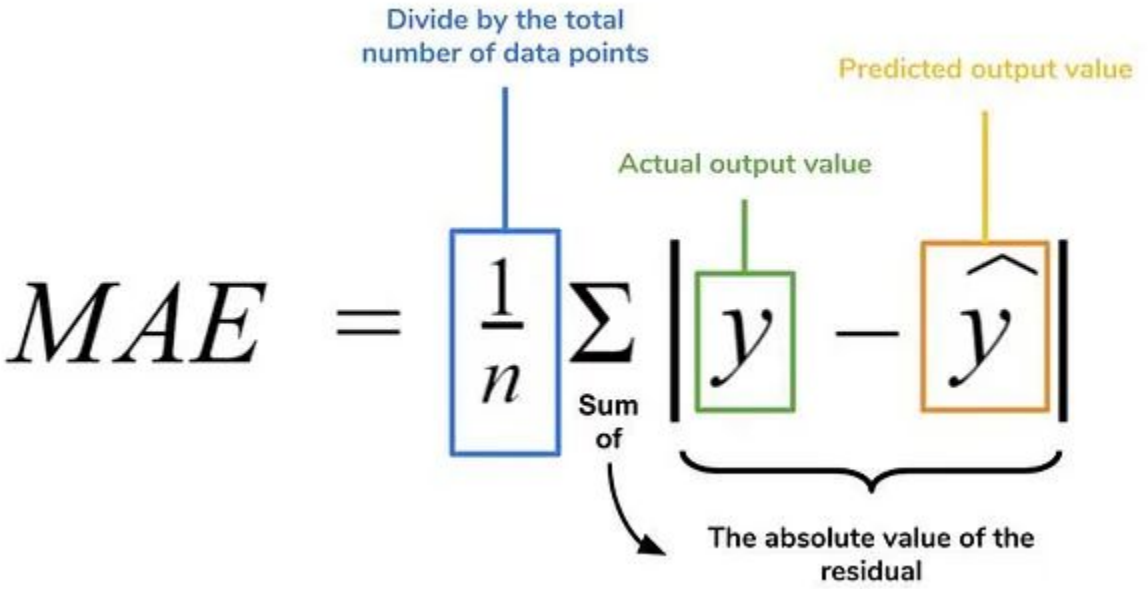
Source:https://kilong31442.medium.com/top-10-%E6%82%A8%E6%87%89%E8%A9%B2%E8%A6%81%E5%AD%B8%E6%9C%83%E7%9A%84%E6%B7%B1%E5%BA%A6%E5%AD%B8%E7%BF%92%E6%BC%94%E7%AE%97%E6%B3%95-fundamental-review-series-d8c69897e010

訓練損失 (training loss)

衡量神經網路預測結果與真實值之間的差距，並根據結果來調整神經網路的權重

- 分類任務：交叉熵 (cross-entropy)
- 迴歸任務：Mean squared error (MSE)

MSE



Source:https://cinnamonaitaiwan.medium.com/cnn%E6%A8%A1%E5%9E%8B-%E6%90%8D%E5%A4%B1%E5%87%BD%E6%95%B8-loss-function-647e13956c50

Cross-Entropy

就是兩個機率分佈的差異

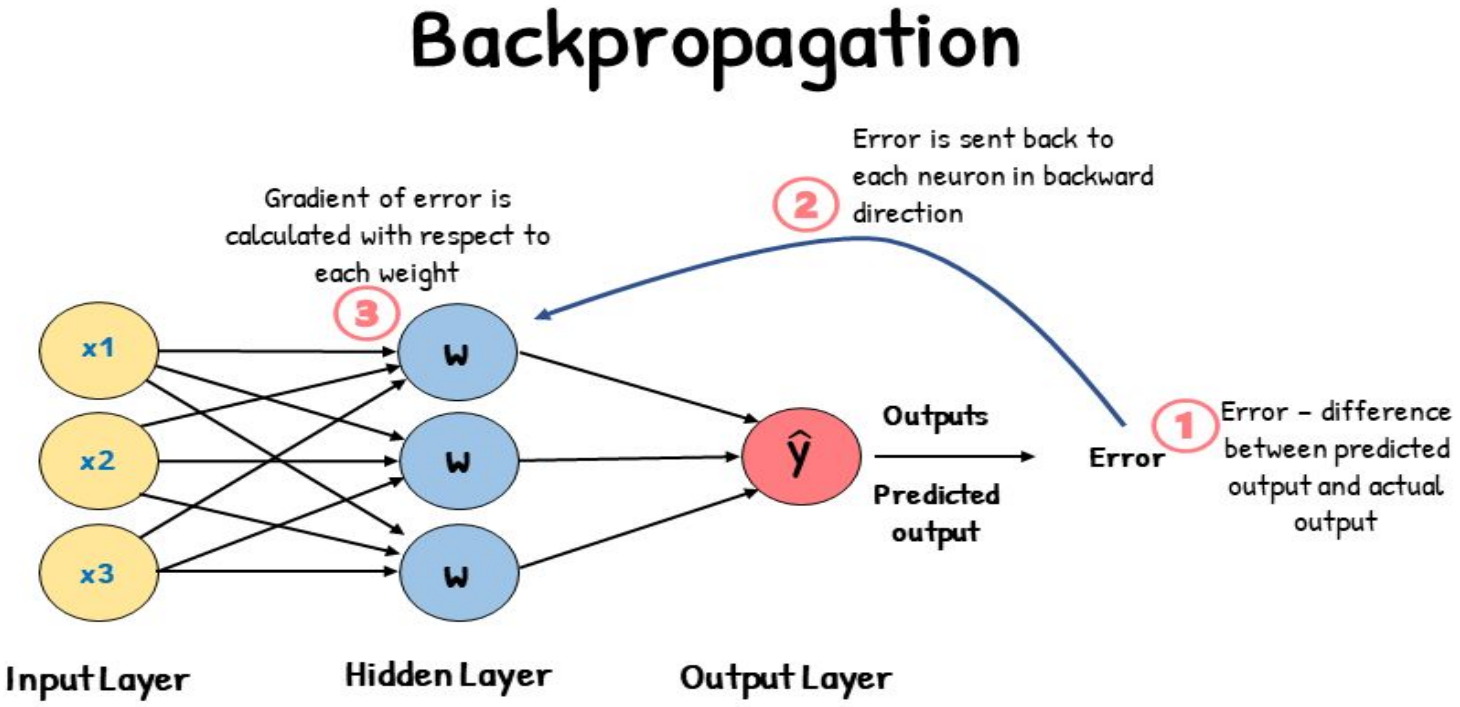
$$H(p,q) = -\sum_{x} p(x) \log q(x).$$

真實機率分布　　預測機率分布

Source:https://cinnamonaitaiwan.medium.com/cnn%E6%A8%A1%E5%9E%8B-%E6%90%8D%E5%A4%B1%E5%87%BD%E6%95%B8-loss-function-647e13956c50

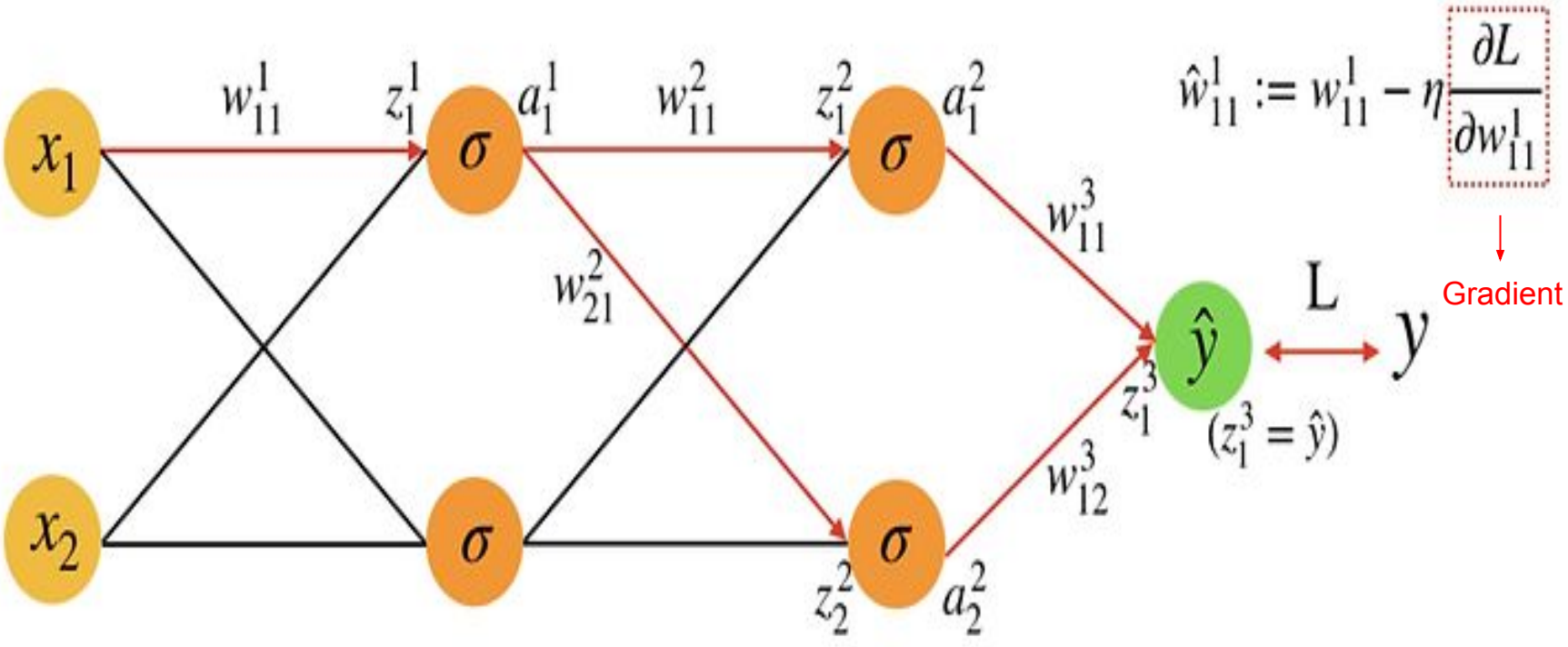
反向傳遞

Backward Propagation



Source:https://www.analyticsvidhya.com/blog/2023/01/gradient-descent-vs-backpropagation-whats-the-difference/

Backward Propagation



Source:https://www.met-ai.net/post/%E5%8F%8D%E5%90%91%E5%82%B3%E6%92%AD%E7%AE%97%E6%B3%95-back-propagation

Backward Propagation

- The gradient of w_{11}^1 :

$$\frac{\partial L}{\partial w_{11}^1} = \frac{\partial L}{\partial z_1^3}[\sum_{i=1}^{2} \frac{\partial z_1^3}{\partial a_i^2}\frac{\partial a_i^2}{\partial z_i^2}\frac{\partial z_i^2}{\partial a_1^1}]\frac{\partial a_1^1}{\partial z_1^1}\frac{\partial z_1^1}{\partial w_{11}^1} = \frac{1}{n}(\hat{y}-y)[\sum_{i}^{2} w_{1i}^3\sigma'(z_i^2)w_{i1}^2]\sigma'(z_1^1)x_1$$

1. 2. 3.

1. $\frac{\partial L}{\partial z_1^3} = \frac{\partial}{\partial \hat{y}}\frac{1}{2n}(\hat{y}-y)^2 = \frac{1}{n}(\hat{y}-y),\ (z_1^3 = \hat{y})$

2. $\frac{\partial z_1^3}{\partial a_1^2} = \frac{\partial}{\partial a_1^2}(a_1^2 w_{11}^3 + a_2^2 w_{12}^3) = w_{11}^3,\ (z_1^3 = a_1^2 w_{11}^3 + a_2^2 w_{12}^3)$

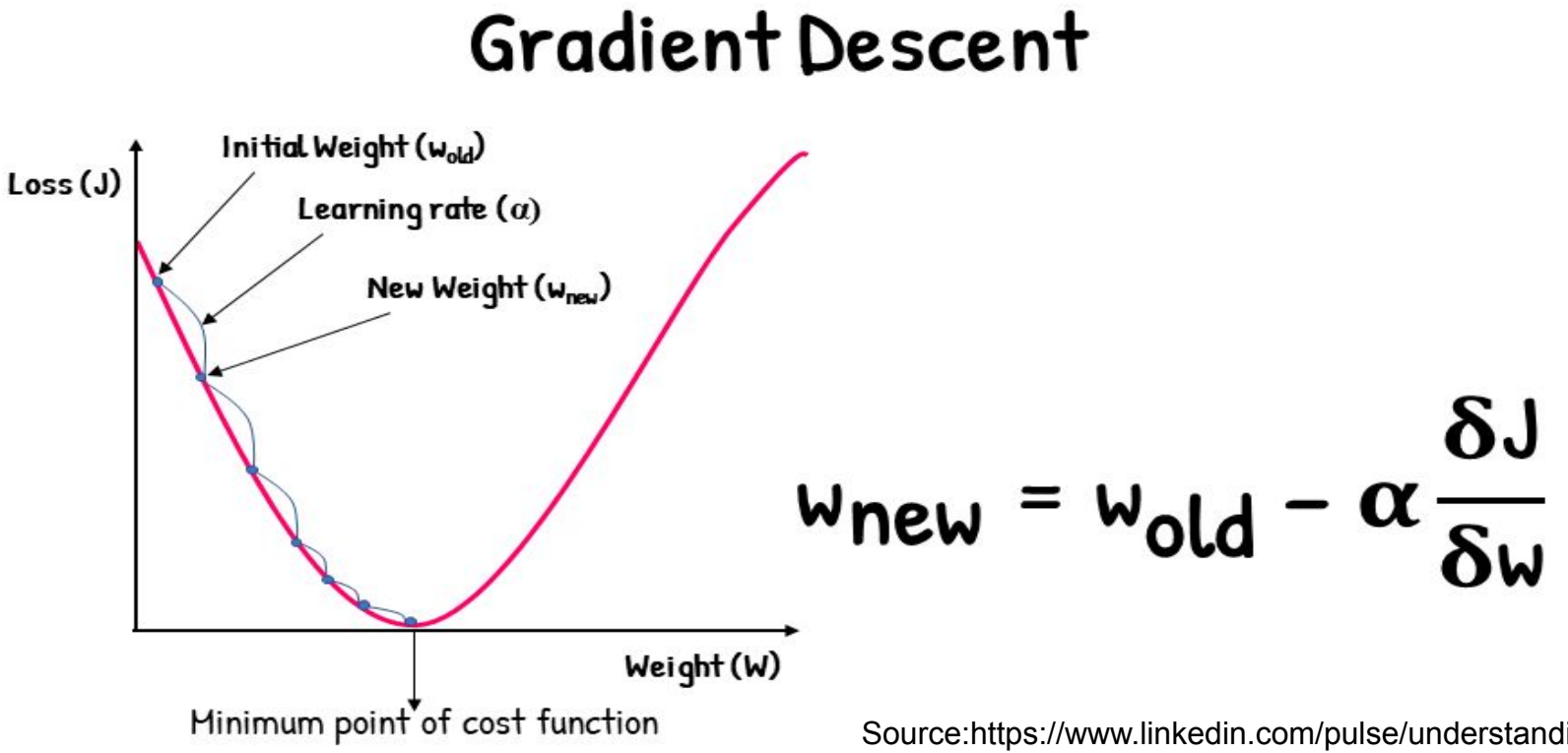
3. $\frac{\partial a_1^2}{\partial z_1^2} = \sigma'(z_1^2) = \sigma(z_1^2)(1-\sigma(z_1^2)),\ (a_1^2 = \sigma(z_1^2))$

Source:https://www.met-ai.net/post/%E5%8F%8D%E5%90%91%E5%82%B3%E6%92%AD%E7%AE%97%E6%B3%95-back-propagation

Gradient Descent

- Gradient Descent是一種優化算法，用來根據Loss的梯度更新神經網路的權重，目的是讓Loss最小化

Gradient Descent



Source:https://www.linkedin.com/pulse/understanding-gradient-descent-algorithm-its-role-linear-mhango-kjbvf/

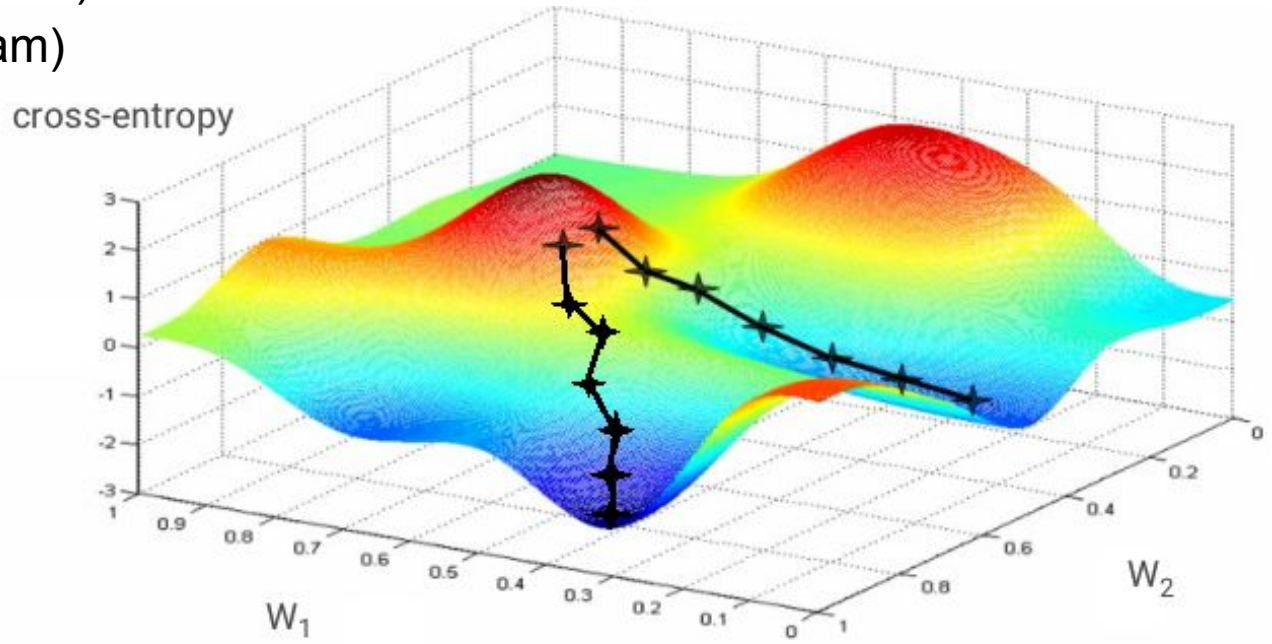
Learning rate

- Learning rate決定了每次梯度下降時，模型的權重應該更新多少
- 太大的Learning rate可能會導致模型更新過快，可能會跳過最佳解
- 太小的Learning rate則可能會導致模型更新過慢，無法到達最佳解
- 是一個需要調整的Hyperparameter

優化器 (Optimizer)

根據訓練損失調整模型參數

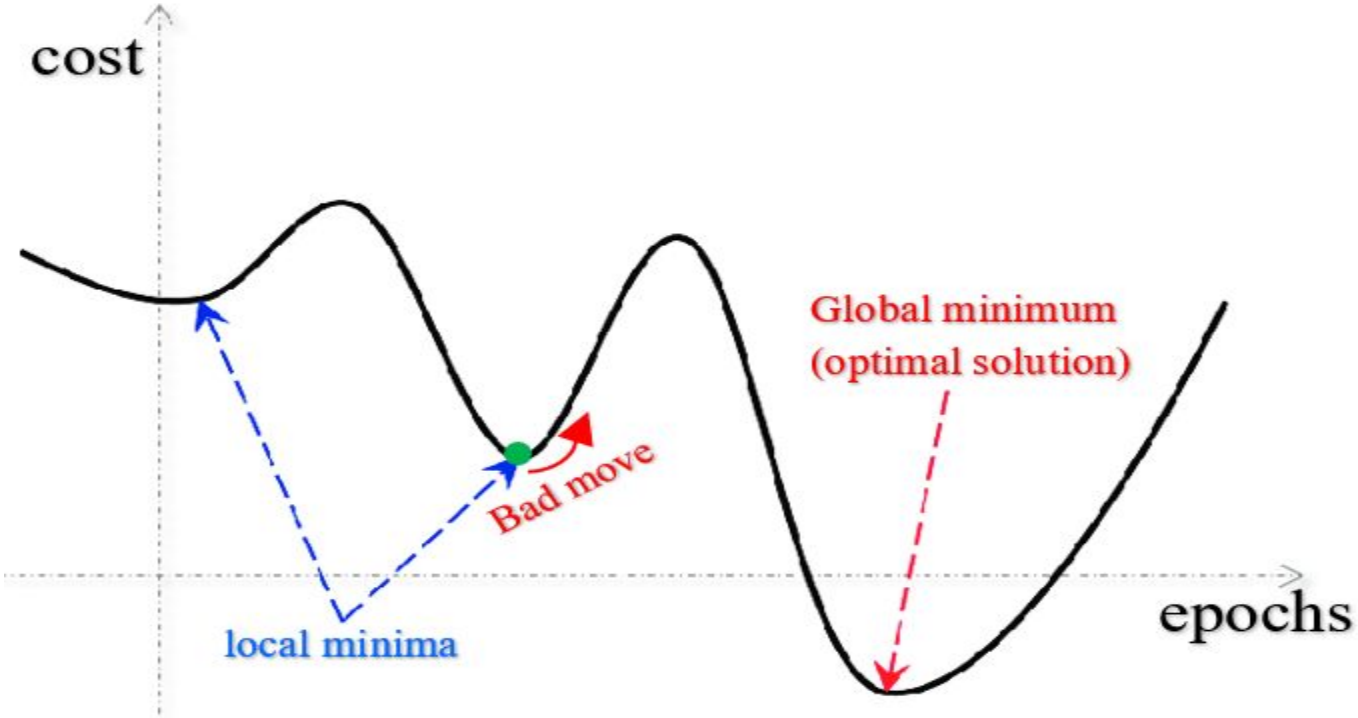
- 隨機梯度下降 (SGD)
- 自適應梯度 (Adam)



隨機梯度下降(SGD)

- 每次更新權重時，選擇一筆資料或一個batch的資料來計算梯度
- 這樣可以加快訓練速度，但可能會導致更新過程中的波動和收斂不穩定
- 但這個不穩定性可能會幫助模型脫離local minima

Local Minima

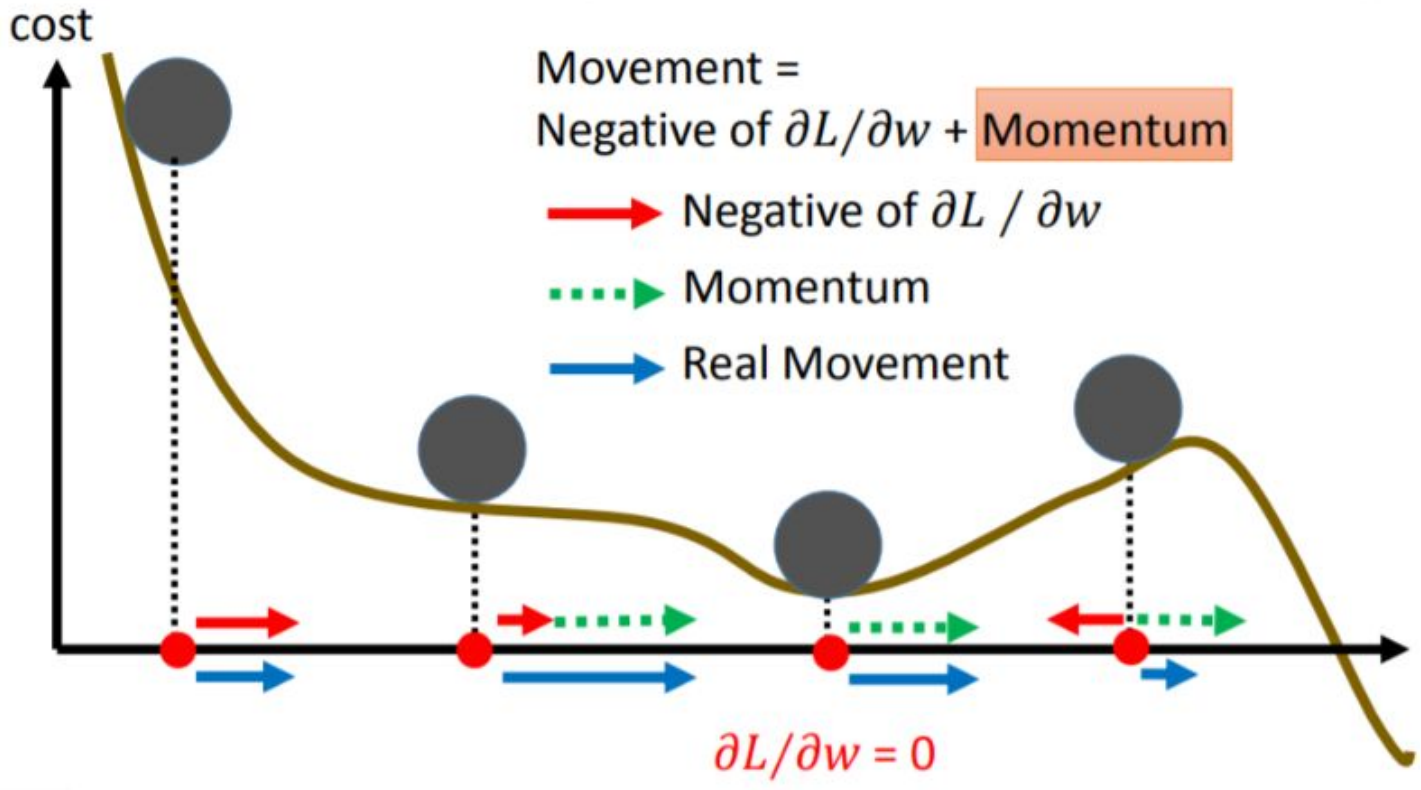


Source:https://www.researchgate.net/figure/local-minima-vs-global-minimum_fig2_341902041

自適應梯度(ADAM)

- 結合了動量（Momentum）和 RMSProp 的優點
- 考慮動量來調整梯度更新方向的同時也自適應地調整Learning rate

Momentum



Source:https://www.youtube.com/@HungyiLeeNTU

RMSProps **是在動態調整Lr**

- RMSProp 根據每個參數的最近幾次更新的平方梯度均值來調整Learning rate
- 如果參數的梯度變化較大，會減少Learning rate，防止步伐過大跳過最佳解
- 如果梯度變化較小，則會相對增大學習率，加速收斂

ADAM

$$m \leftarrow \beta_1 m - (1 - \beta_1)\nabla_\theta J(\theta)$$

$$s \leftarrow \beta_2 s - (1 - \beta_2)\nabla_\theta J(\theta)$$

$$\hat{m} \leftarrow \frac{m}{1-\beta_1^T} ; \hat{s} \leftarrow \frac{s}{1-\beta_2^T}$$

$$\theta_{nextstep} \leftarrow \theta + \frac{\eta \hat{m}}{\sqrt{\hat{s}+\epsilon}}$$

η: learning rate

s: exponential average square of gradients

m: momentum vector

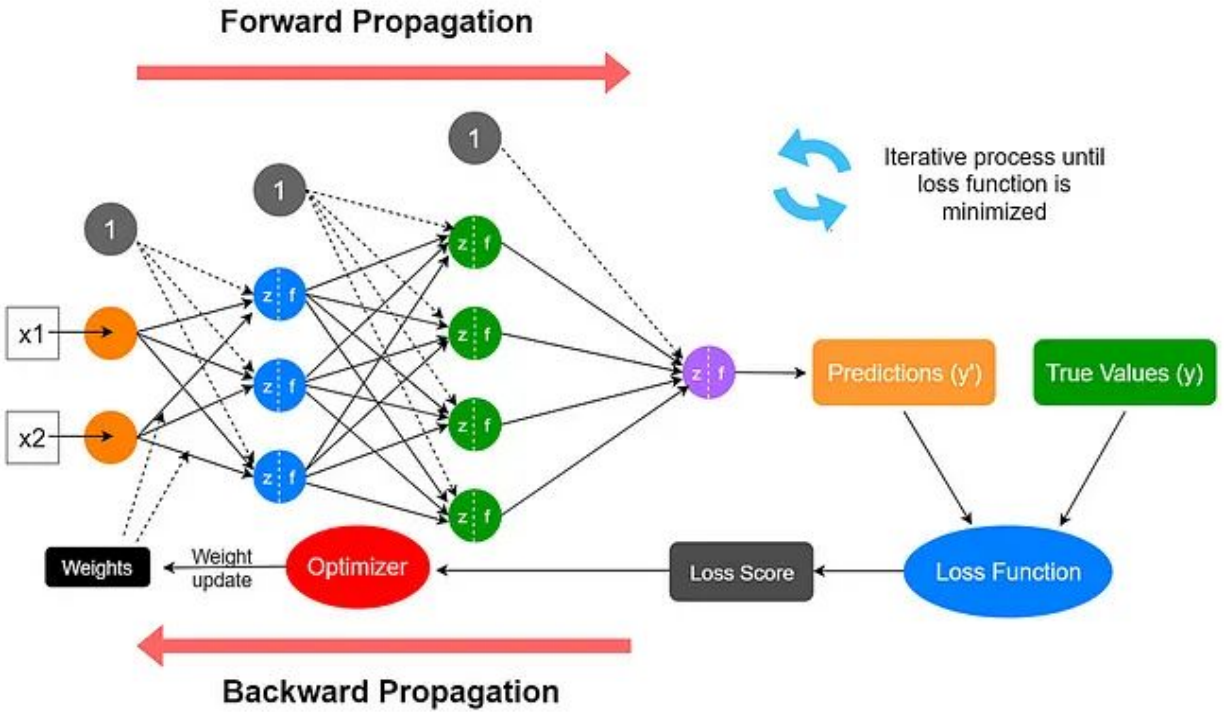
β1: momentum decay, typically set at 0.9

β2: scaling decay, typically set at 0.999

Ε: smoothing term

Source:https://medium.com/@nishantnikhil/adam-optimizer-notes-ddac4fd7218

Whole picture



標籤編碼

- Label encoding (sparse representation)
- One hot encoding

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

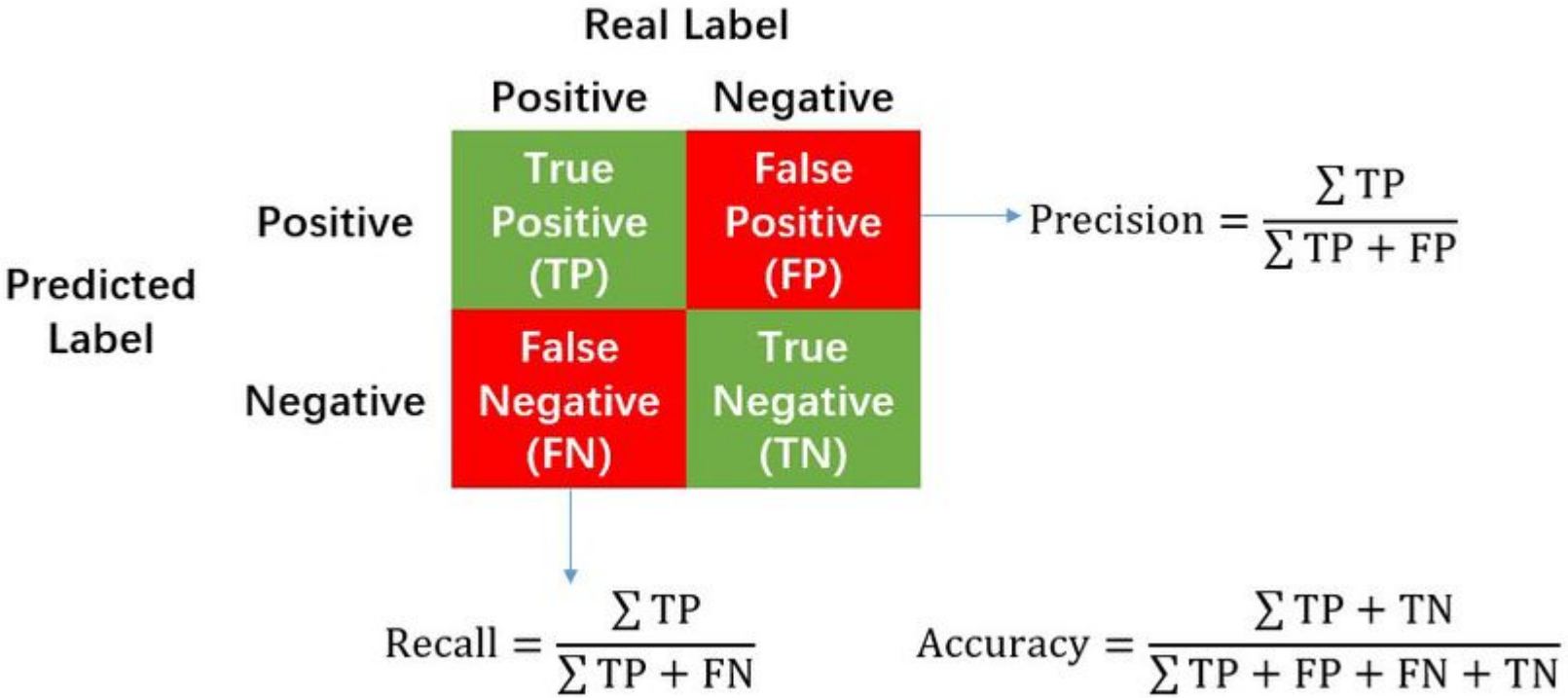
One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Source: https://medium.com/@michaeldelsole/what-is-one-hot-encoding-and-how-to-do-it-f0ae272f1179

Metrics

通常使用 Accuracy



Metrics

- Precision (精確率)
  - 模型預測為positive的結果中有多少是True positive
- Recall (召回率)
  - 評估的是模型能夠正確識別出多少實際為positive的結果
- Accuracy (準確率):
  - 反映了模型在所有測試數據中有多少預測的準確率

需要設定的超參數

- 每一層
  - Number of nodes
  - Activation function
- Optimizer
- Learning rate
- Training loss
- Metrics

Implementation

Download notebook at:
https://github.com/albert831229/nchu-computer-vision/tree/main/113/day