大數據智慧分析模型建置機制之運作說明

機器學習基石

北祥股份有限公司
PERSHING SYSTEMS CORPORATION

第一部分 概論

- 機器學習、深度學習與傳統統計方法差異
- 機器學習、深度學習與傳統統計方法於各領域應用

傳統統計方法

- 如何得知母體情報？--點估計
- 如何衡量估計品質？--不偏性、有效性、一致性
- 估計品質的極限？--UMVUE(uniformly minimum-variance unbiased estimator)
- 頻率學派　vs　貝氏學派

機器學習與傳統統計的異同

- 大師觀點(Ian Goodfellow)

機器學習為應用統計的一種形式。比較於傳統統計來說，更多的使用電腦做出複雜的估計函數，較少著重在證明這些函數的信賴區間

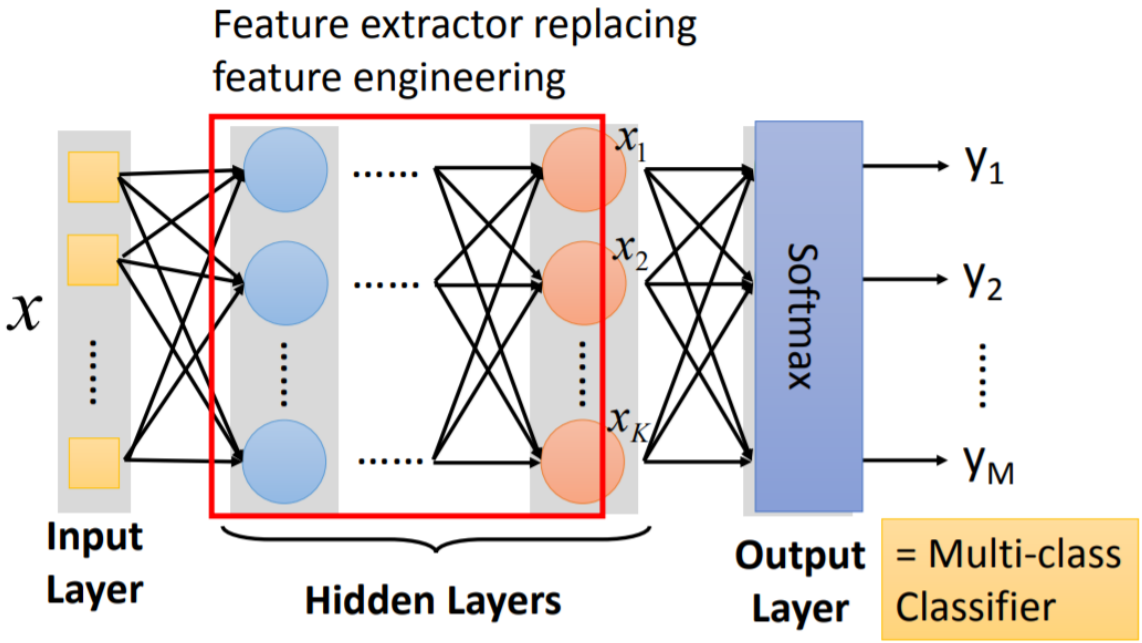
範例：

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

旅遊景點人數　　小吃店家數　　當日雨量

機器學習與深度學習的異同

- 深度學習為機器學習的分支
- 深度學習更重視特徵自動化萃取



(圖片取自李宏毅老師上課教材)

傳統統計模型應用

- 因素分析--分析顧客屬性
- 存活分析--預估標的存活率
- 未觀察效果模型--複雜狀況排除
- 工具變數模型--互為因果處理

機器學習應用

- 分類議題--該如何推薦產品給顧客？
- 分群議題--購買某產品的顧客都長什麼樣子？
- 關聯性探討--新產品開發可以朝什麼方向發展？

深度學習應用

- AlphaGo--圖像辨識、增強學習
- 對話機器人--語音辨識、問答模型
- 機器作詩、作畫--生成式對抗網路

第二部分 傳統統計常用模型--迴歸

- 迴歸概念
- 連續性數值的預測--簡單迴歸 / 複迴歸
- 解決分類問題--羅吉斯迴歸 / 離散選擇模型
- 迴歸模型簡單應用--中介效果模型 / 交互作用模型
- 進階迴歸模型--如何解決因果問題？
- 實際案例分享

迴歸概念

- 以參數方式評估變數影響力
- 顯著與否的意義
- 依預測標的為連續型或間斷型而有不同迴歸模型

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

旅遊景點人數　　小吃店家數　　當日雨量

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

顧客購買的洗髮精品牌　　產品顏色　　顧客年齡

迴歸概念

- 迴歸模型準確與否--內生性問題

缺少重要變數--同時影響Y及X

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

身體健康指數　　做早操　　早起

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon$$

身體健康指數　　做早操

互為因果 / 因果倒置

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon$$

商品價格　　購買數量

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon$$

購買數量　　商品價格

簡單迴歸 / 複迴歸技巧

- 將變數改為ln？
- 加入二次項？

$$\ln Y = \beta_0 + \beta_1 X_1 + \varepsilon$$

畫作拍賣價格　　畫作尺寸

$$\ln Y = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2 + \varepsilon$$

畫作拍賣價格　　畫作尺寸　　畫作尺寸平方項

羅吉斯迴歸

- 處理二元分類問題
- logistic function--軟性分類

(範例來源：wiki)

Hours	0.50	0.75	1.00	1.25	1.50	1.75	1.75	2.00	2.25	2.50	2.75	3.00	3.25	3.50	4.00	4.25	4.50	4.75	5.00	5.50
Pass	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1

	Coefficient	Std.Error	P-value
Intercept	−4.0777	1.7610	0.0206
Hours	1.5046	0.6287	0.0167

Hours of study	Passing exam		
	Log-odds	Odds	Probability
1	−2.57	0.076 ≈ 1:13.1	0.07
2	−1.07	0.34 ≈ 1:2.91	0.26
3	0.44	1.55	0.61
4	1.94	6.96	0.87
5	3.45	31.4	0.97

機率 = $\frac{1}{1+b^{-(\beta_0+\beta_1 x_1+\beta_2 x_2)}}$

多重羅吉斯迴歸

- 處理多元分類問題
- 多個羅吉斯迴歸重疊

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

顧客購買的洗髮精品牌　　產品顏色　　顧客年齡

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

顧客是否購買洗髮精品牌A　　產品顏色　　顧客年齡

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

顧客是否購買洗髮精品牌B　　產品顏色　　顧客年齡

中介效果模型

- 直接影響　or　間接影響

顯著

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon$$

身體健康指數　　做早操

顯著

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon$$

身體健康指數　　早起

顯著

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

身體健康指數　　做早操　　早起

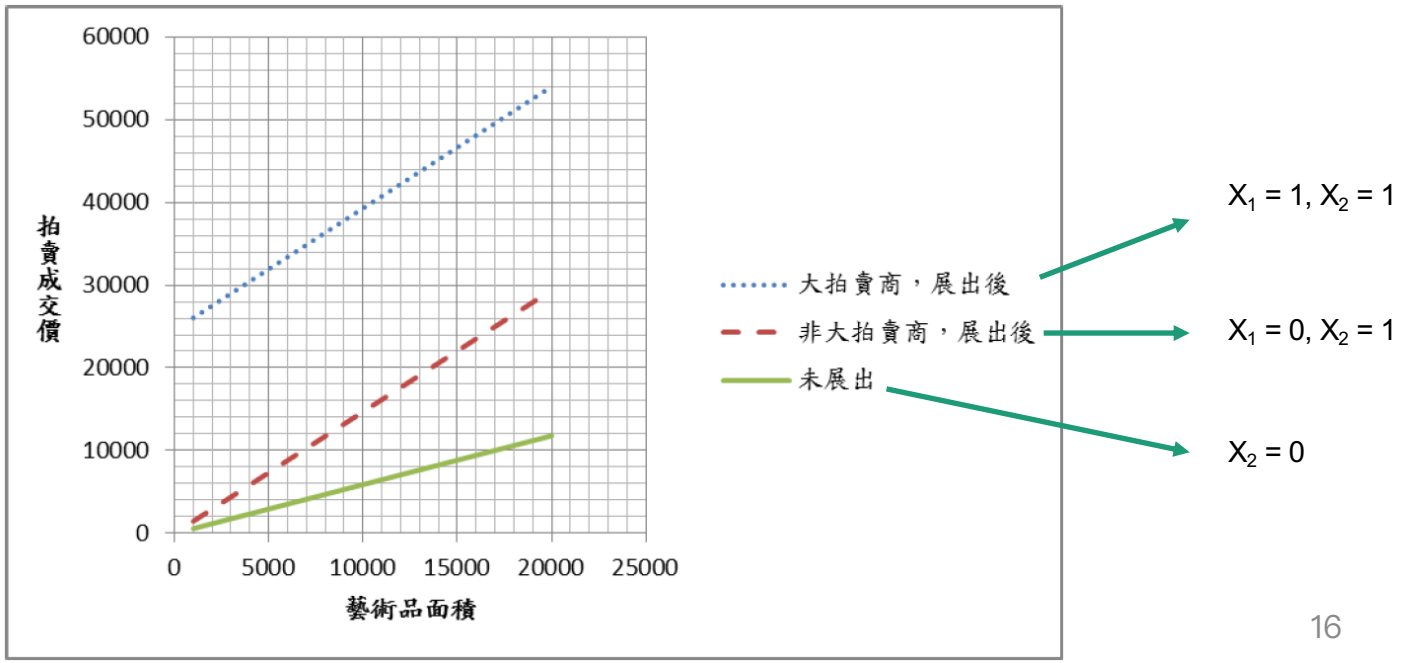
交互作用模型

- 了解變數之間的交互影響

顯著　　顯著　　顯著

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_1 X_3 + \beta_5 X_2 X_3 + \varepsilon$$

畫作價格　　大拍賣商　　展覽　　面積



進階模型--工具變數迴歸

- 解決內生性問題(互為因果、遺漏變數)
- 2SLS

電影評價分數　　電影觀看人數

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_K x_K + u$$

$$x_K = \delta_0 + \delta_1 x_1 + \delta_2 x_2 + \cdots + \delta_{K-1} x_{K-1} + \theta_1 z_1 + r_K$$

電影觀看人數　　當日是否下雨

實際案例分享

- 廣告效益評估怎麼做？

Step1: 確認廣告形式，廣告影響如何測量(Y)

Step2: 訂出影響可能因素(X)

Step3: 趨勢圖觀察--區分系統性影響

Step4: 廣告影響時間長短測試

第三部分　機器學習基礎

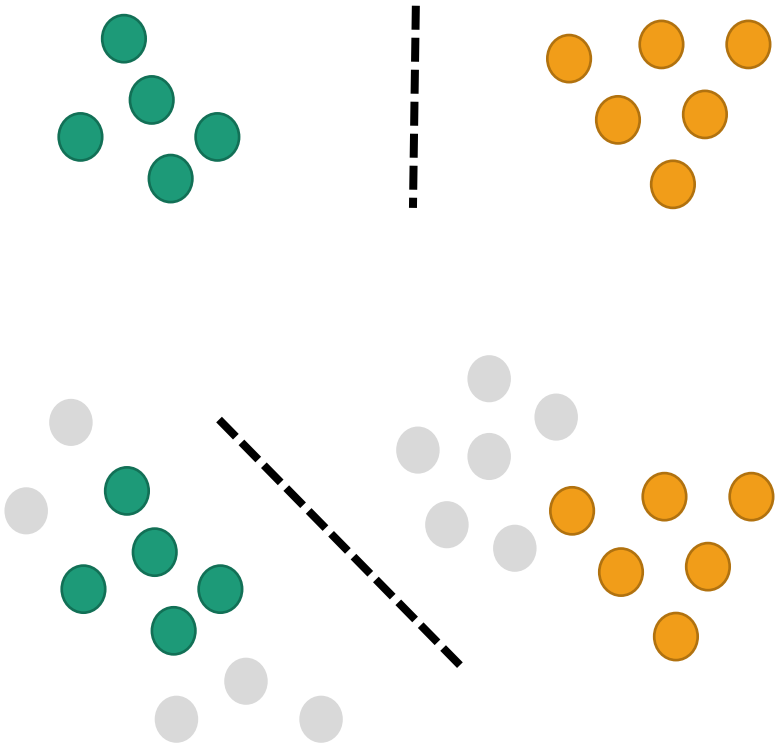
- 機器學習基本流程
- 監督式 / 半監督式 / 非監督式學習的差異
- 何謂過度擬合？如何讓模型避免過度擬合?
- 超參數 / 驗證集的作用
- 資料不平均可能造成的問題及處理方法
- 實際案例分享--如何從頭開始建構模型

機器學習基本流程

- 盤點欲探討的問題與現有資源
- 資料整合 / 資料定位
- 資料探索--找尋切入點
- 提出模型架構
- 驗證模型

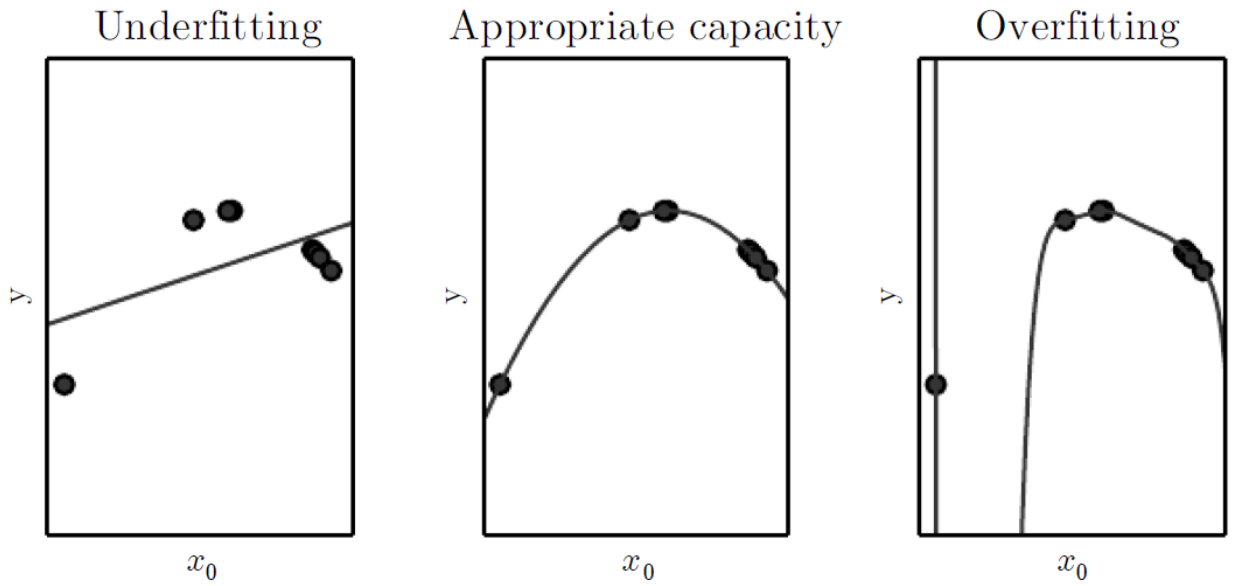
監督式 / 半監督式 / 非監督式

- 監督者的詞意
- 監督式 / 非監督式邊界模糊，常綜合使用



過度擬合 / 擬合不足

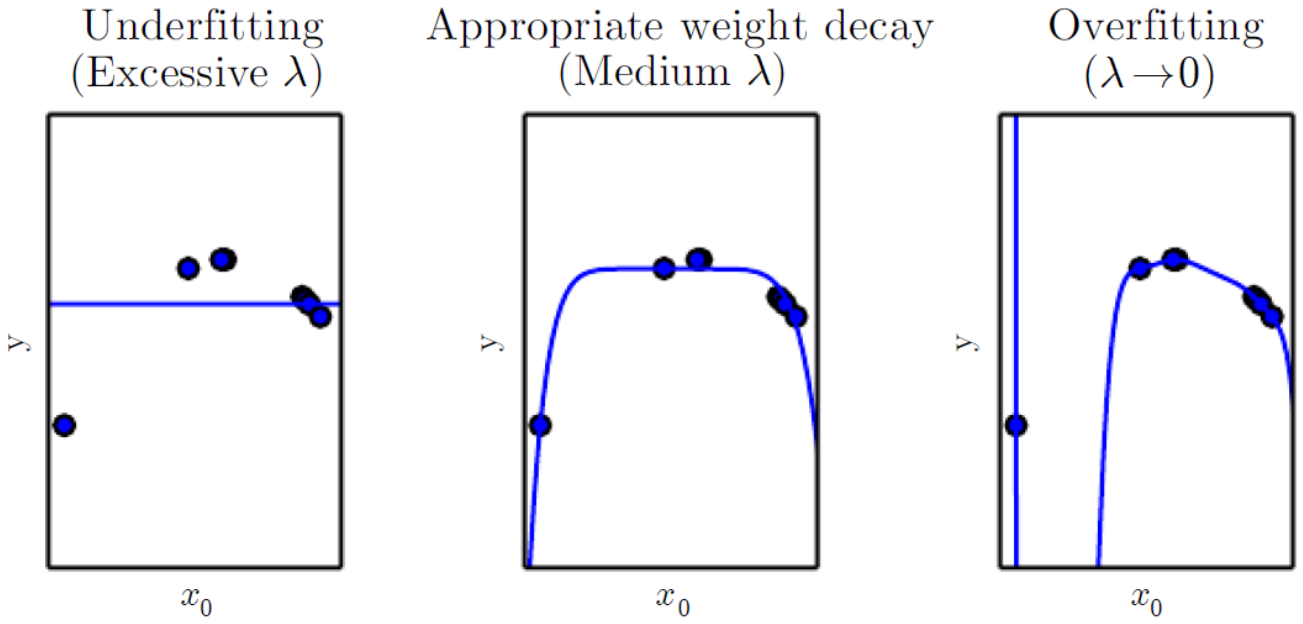
- 什麼是「capacity」
- 「training error」及「generalization error」



(圖片來源： Ian Goodfellow的Deep Learning書籍)

正規化(Regularization)

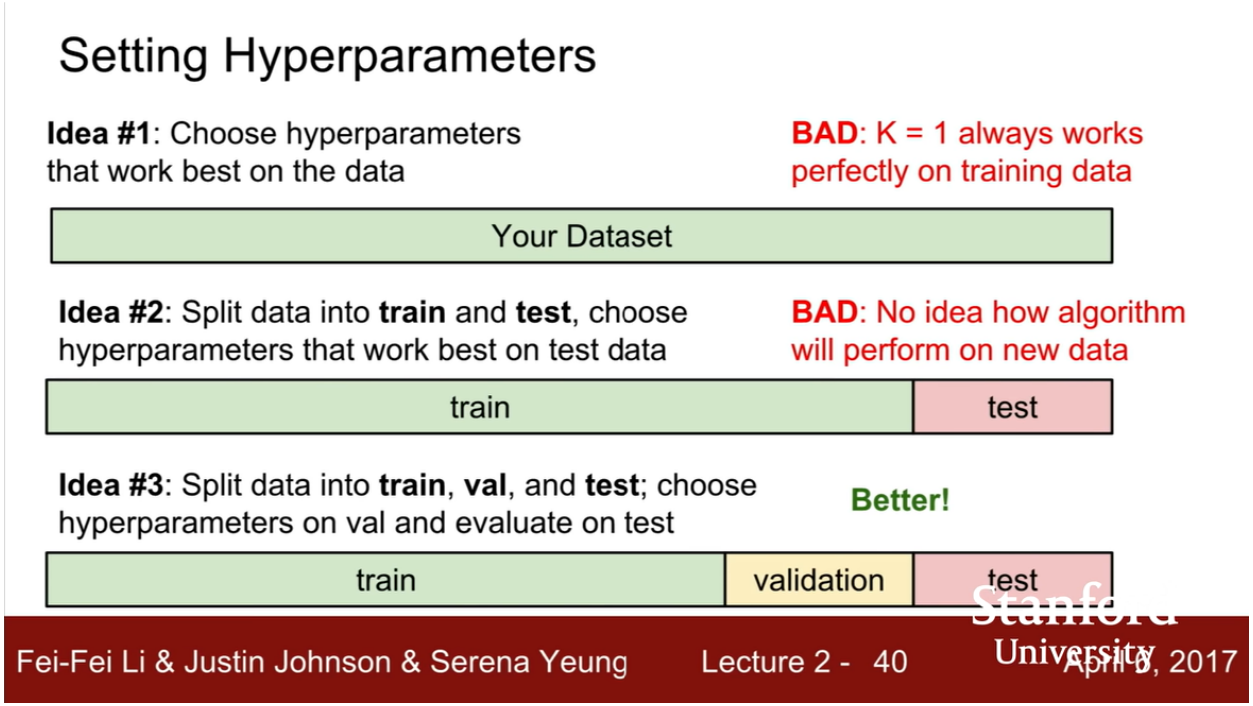
- 減少「generalization error」
- 舉例：迴歸的weight decay



(圖片來源： Ian Goodfellow的Deep Learning書籍)

超參數 / 驗證集

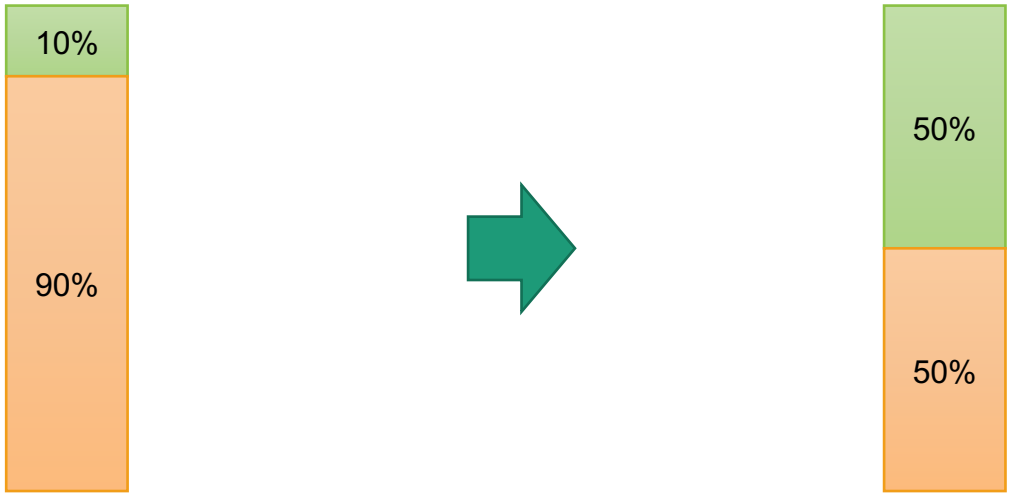
- 超參數為控制學習演算法的設定
- 「validation set」為參數調整的參照



(圖片來源： Stanford的cs231n課程內容)

資料不平均

- 類別比例不均（如：90%為正常，10%為異常）
- 讓少數資料特徵顯現（抽樣、複製）



實際案例分享--如何從頭開始建構模型

- 如何推薦顧客手機？

Step1：盤點欲探討的問題與現有資源	→	問題點切分：新舊客 現有資料：手機銷售紀錄、問卷調查、觀察記錄
Step2：資料整合 / 資料定位	→	手機銷售紀錄：舊客模型 問卷調查、觀察記錄：新客模型
Step3：資料探索--找尋切入點	→	成交率與推薦手機
Step4：提出模型架構	→	兩階段式模型：先預測是否成交，再決定推哪隻
Step5：驗證模型	→	資料驗證；小規模驗證

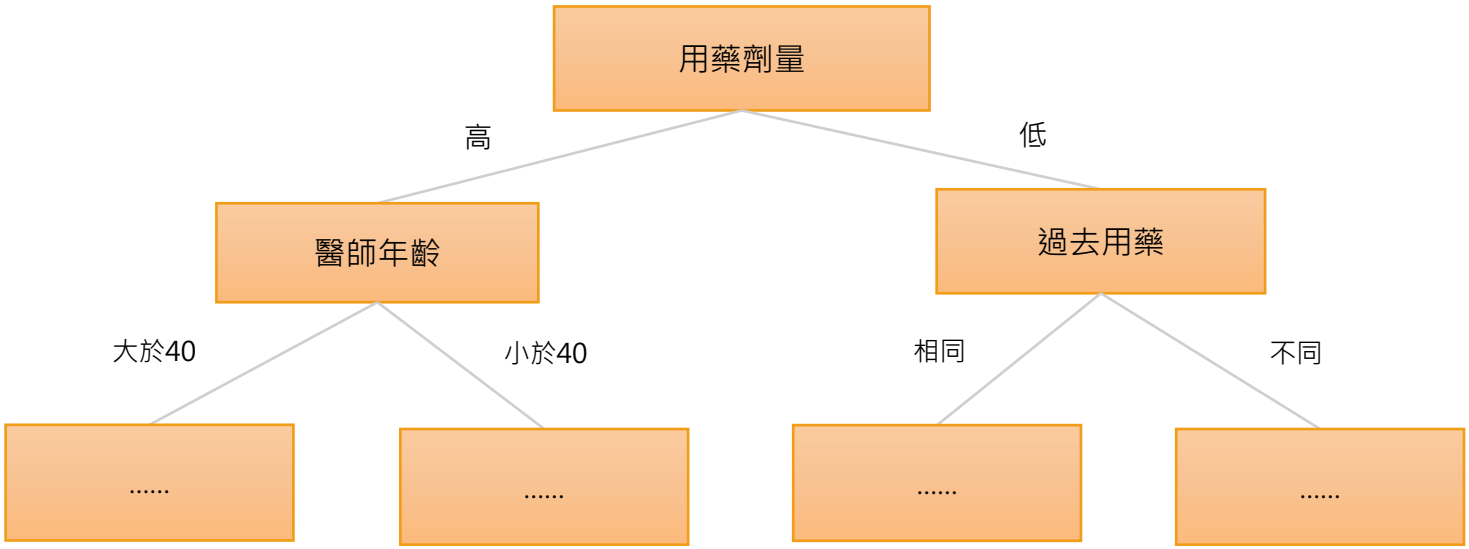
第四部分　機器學習常用模型

- 解釋性高的模型--決策樹
- 基於決策樹的進階模型--隨機森林、GBDT
- 二元分類的霸主--SVM
- Graph Model--貝氏網路
- 常用的分群方法—K-means
- 基於密度的分群方法--DBSCAN
- 主流推薦系統算法--矩陣分解
- 專案模型實務經驗分享

決策樹

- 易於讓「人」理解
- ID3、C4.5、C5.0、CART

分類樹　　迴歸樹



基於決策樹的算法--隨機森林、GBDT

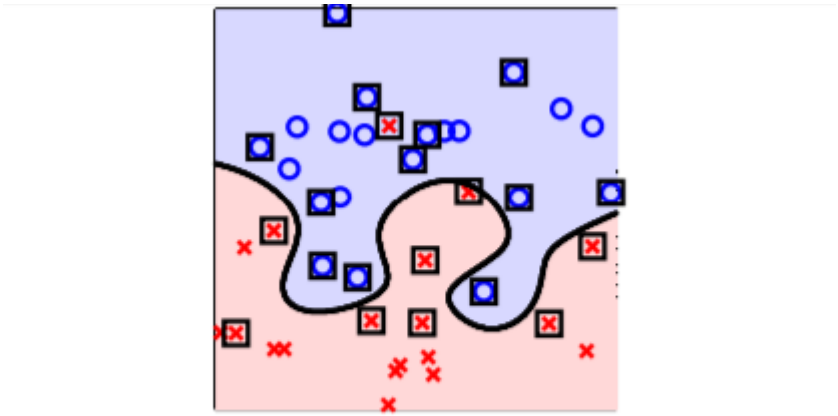
- 隨機森林--投票
- GBDT--利用殘差

隨機森林

GBDT

SVM

- 模型由Support Vector決定劃分界線
- Kernel的意義--考量Regularization



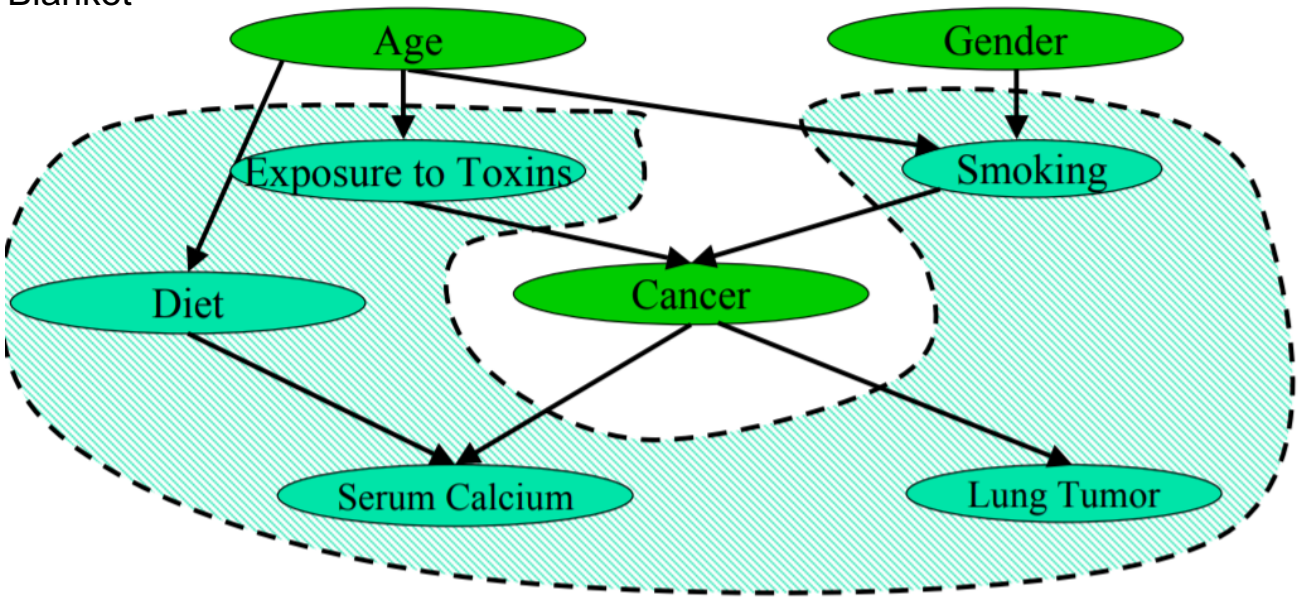
soft-margin Gaussian SVM

(圖片來源：林軒田老師上課PPT)

貝氏網路

- 有向圖模型
- 減低運算複雜度--條件獨立的重要

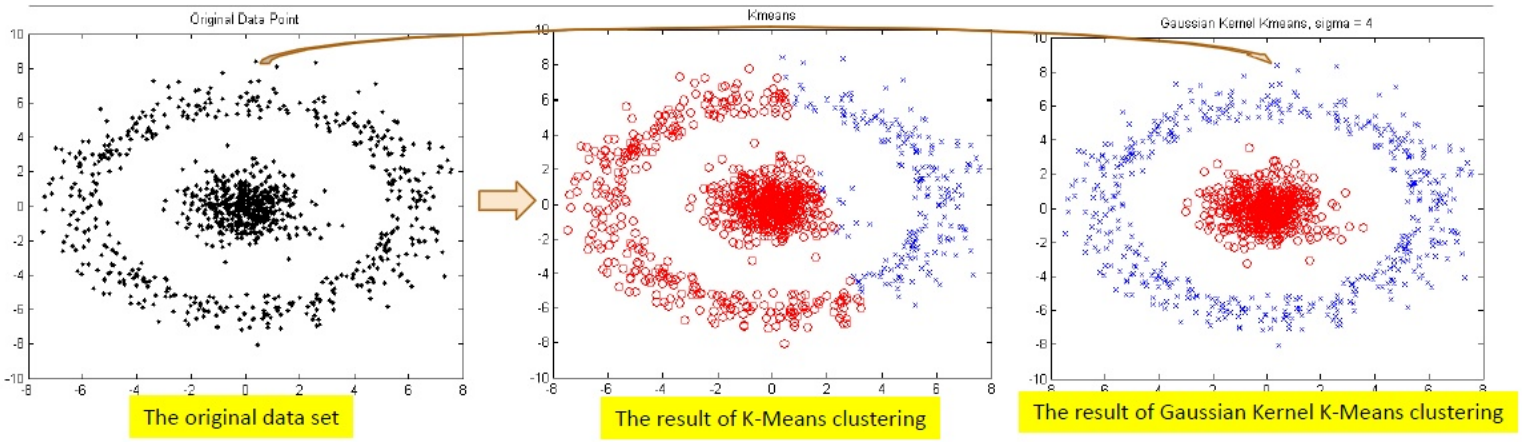
Markov Blanket



[Breese & Koller, 97]

K-means

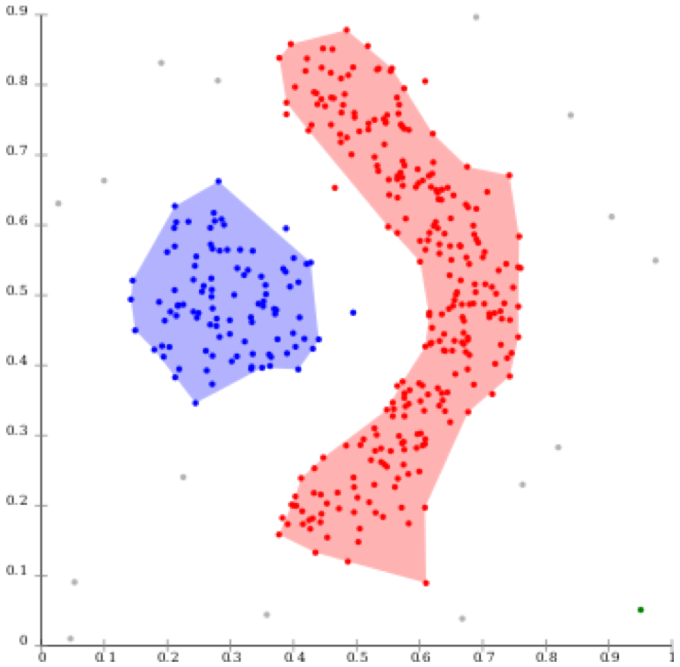
- Partition-based methods
- 易與其他方法結合



(圖片來源：知乎)

DBSCAN

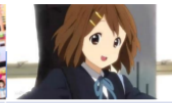
- Density-based methods
- 解決不規則形狀問題



(圖片來源：wiki)

矩陣分解

- 推薦系統常用
- 觀察大量行為 / 評分做出預測

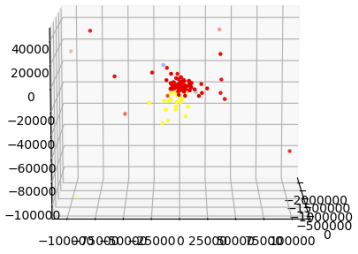
				
A	5	3	?	1
B	4	3	?	1
C	1	1	?	5
D	1	?	4	4
E	?	1	5	4

				
A	5	3	-0.4	1
B	4	3	-0.3	1
C	1	1	2.2	5
D	1	0.6	4	4
E	0.1	1	5	4

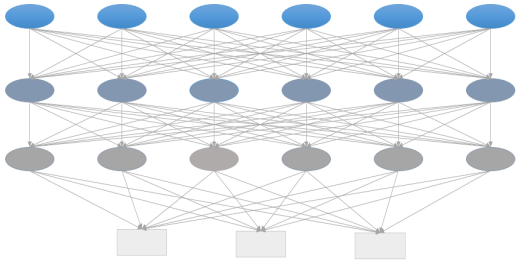
(圖片來源：李宏毅老師上課PPT)

專案實務模型分享-初期規劃

- 以有無開藥 / 有無診斷嘗試找出審查異常群集
- 建構模型：LSI、貝氏網路

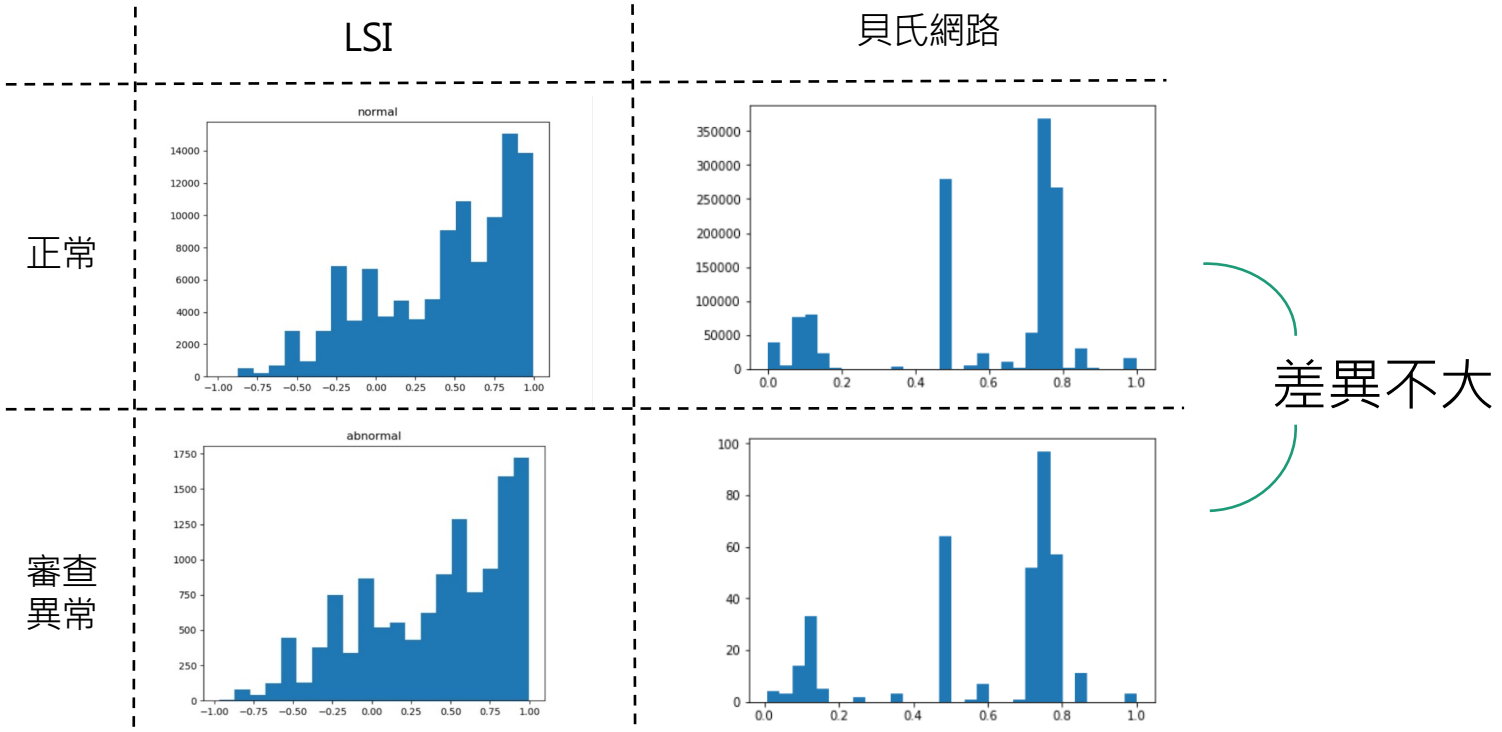


LSI：探討診斷與用藥間的關係



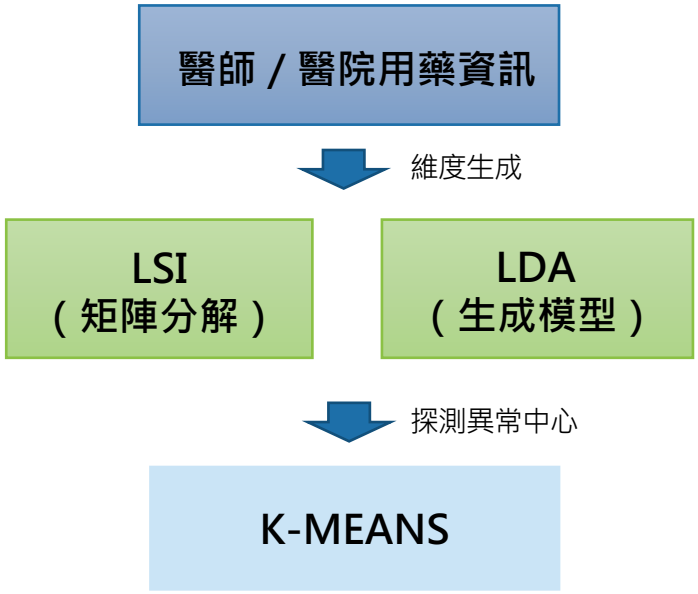
貝氏網路：模擬醫生開藥機率

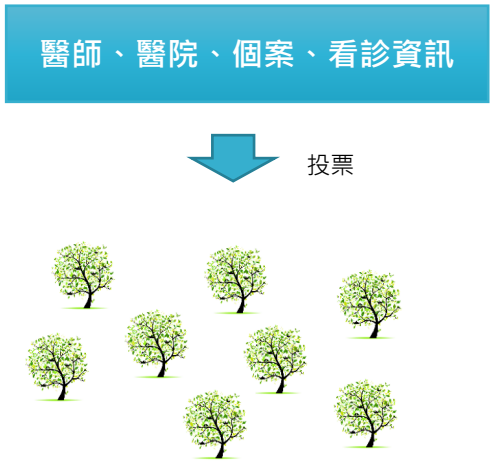
專案實務模型分享-初期遭遇難題



專案實務模型分享-模型中期調整

- 以診斷 / 用藥資訊無法將審查異常族群分離，改以「用藥型態」著手
- 建構模型：LSI、LDA、RF

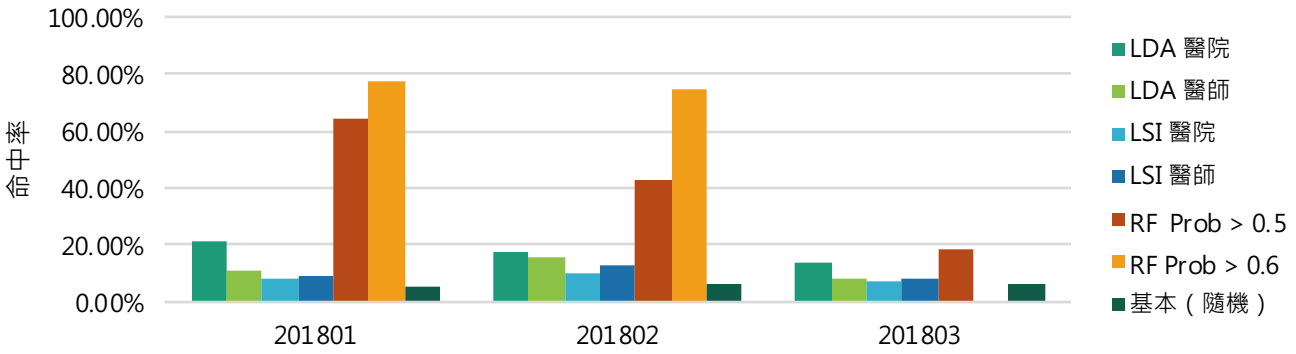




模型結果與期待落差

模型於三個月的測試中取得一定成果，但與需求單位期待尚有落差，主要落差為：

- **模型解釋性不足**
- **醫審歷史結果可信度不高**

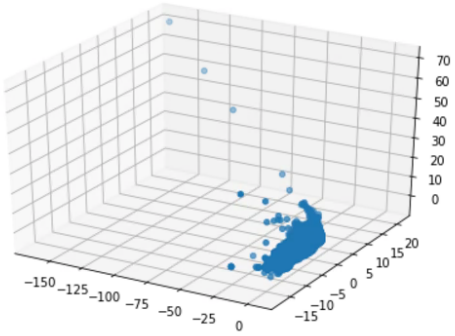


	LDA (醫院)		LDA (醫師)		LSI (醫院)		LSI (醫師)		RF Prob > 0.5		RF Prob > 0.6		基本比例
	猜測筆數	命中率	猜測筆數	命中率	猜測筆數	命中率	猜測筆數	命中率	猜測筆數	命中率	猜測筆數	命中率	
2018/01	122	21.3%	116	11.2%	1,435	8.2%	1,792	9.0%	28	64.2%	13	77.0%	5.0%
2018/02	136	17.6%	121	15.7%	1,262	10.4%	1,695	13.0%	14	42.9%	4	75.0%	5.9%
2018/03	111	13.5%	82	8.5%	1,385	7.1%	1,757	8.5%	11	18.2%	0	-	6.1%

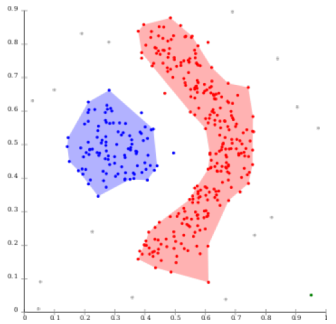
最終模型建置

- 依據選出的變數做資料分群，找出離群值
- 建構模型：MDS+DBSCAN

原始變數



MDS：維度縮減



DBSCAN：找離群值

最終模型結果

費用年月	送審案件數	核減案件數	核減比例	隨機抽審比例
201801	501	161	32.14%	5.0%
201802	516	162	31.40%	5.9%
201803	568	187	32.92%	6.1%
Total	1585	510	32.18%	

201801-03

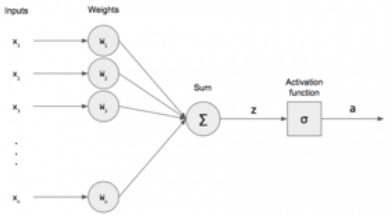
rf_prob	送審案件數	核減案件數	核減比例
>0.5	26	16	61.54%
>0.4	154	81	52.60%
>0.3	858	322	37.53%
>0.0	1585	510	32.18%

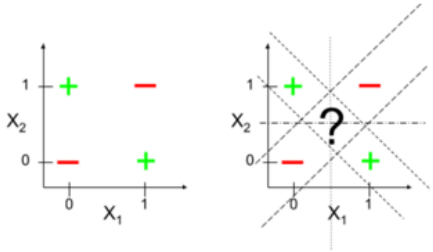
第五部分　深度學習基礎

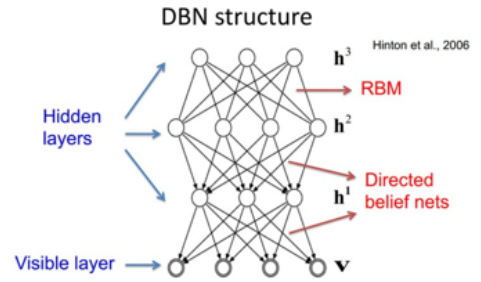
- 深度學習歷史演進
- 模型基本架構
- 圖像辨識 / 分層 / 物體偵測--CNN
- 順序性模型--RNN

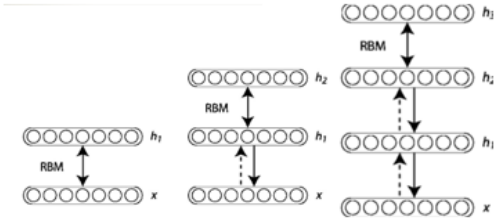
深度學習歷史演進

- 1958: Perceptron (linear model)
- 1969: Perceptron has limitation
- 1980s: Multi-layer perceptron
  - Do not have significant difference from DNN today
- 1986: Backpropagation
  - Usually more than 3 hidden layers is not helpful
- 1989: 1 hidden layer is “good enough”, why deep?
- 2006: RBM initialization (breakthrough)
- 2009: GPU
- 2011: Start to be popular in speech recognition
- 2012: win ILSVRC image competition





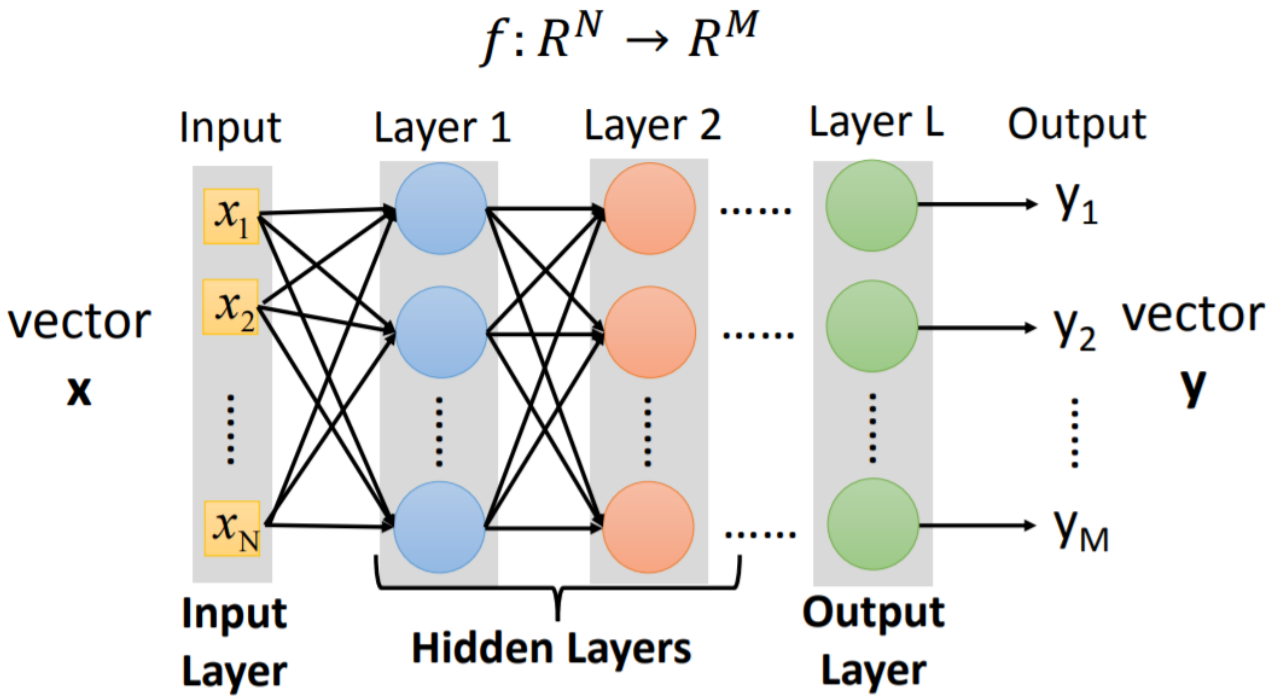




(文字內容來源：李宏毅老師上課PPT)

模型基本架構

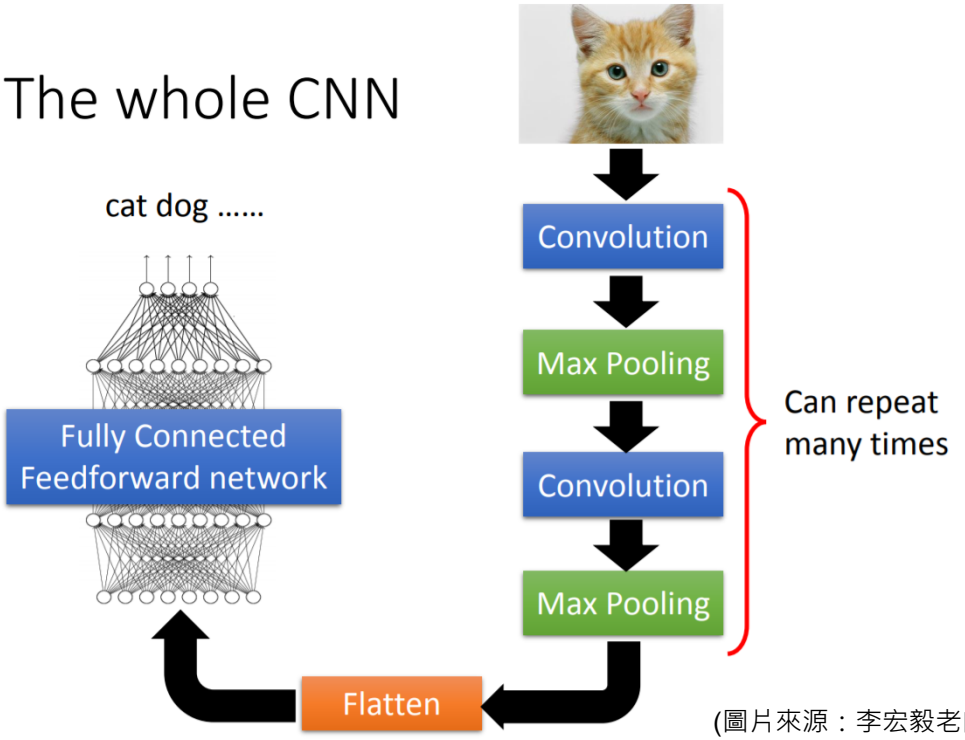
- 前面幾層做特徵萃取，最後一層為分類器
- 若激發函數為sigmoid，等同於串接很多羅吉斯回歸



(圖片來源：李宏毅老師上課PPT)

CNN

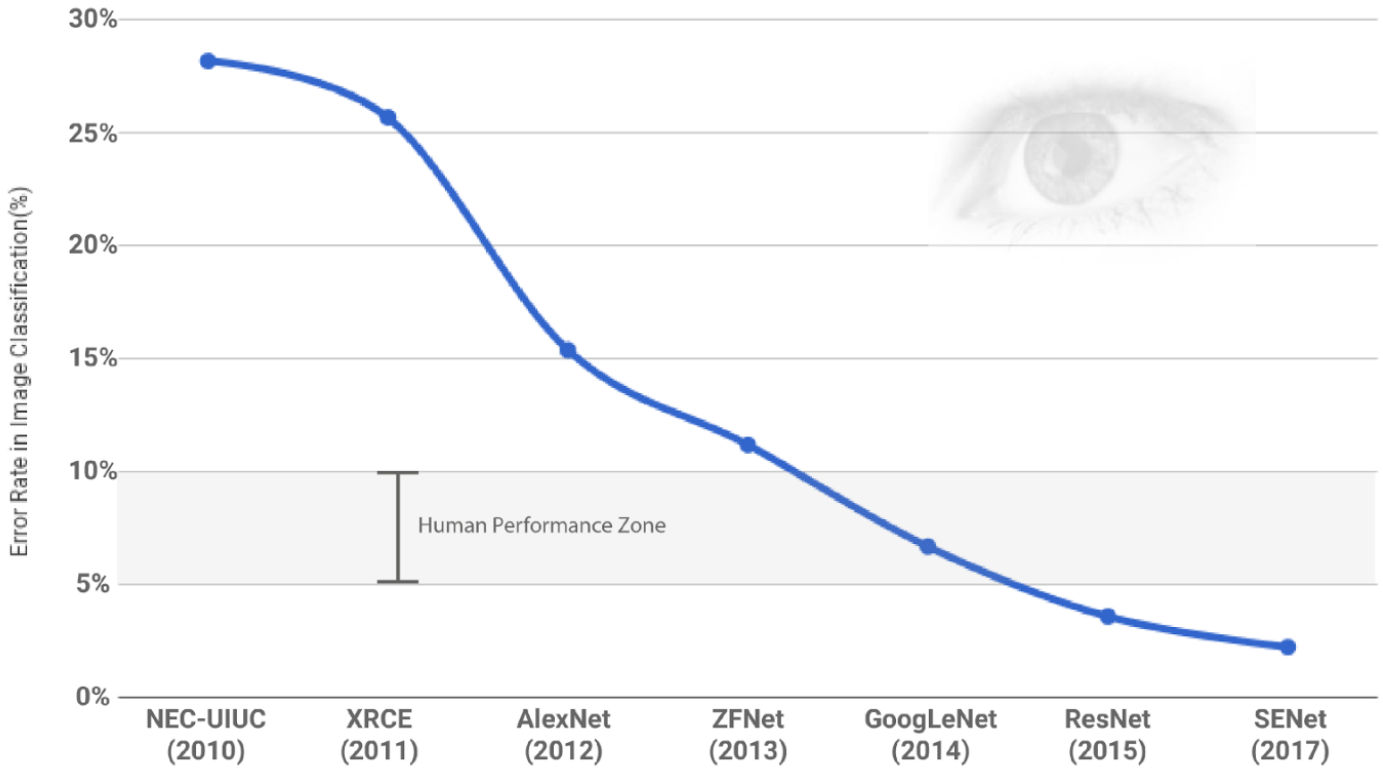
- 卷積層與池化層的作用
- 最基礎的應用--圖像辨識



(圖片來源：李宏毅老師上課PPT)

CNN

- ILSVRC

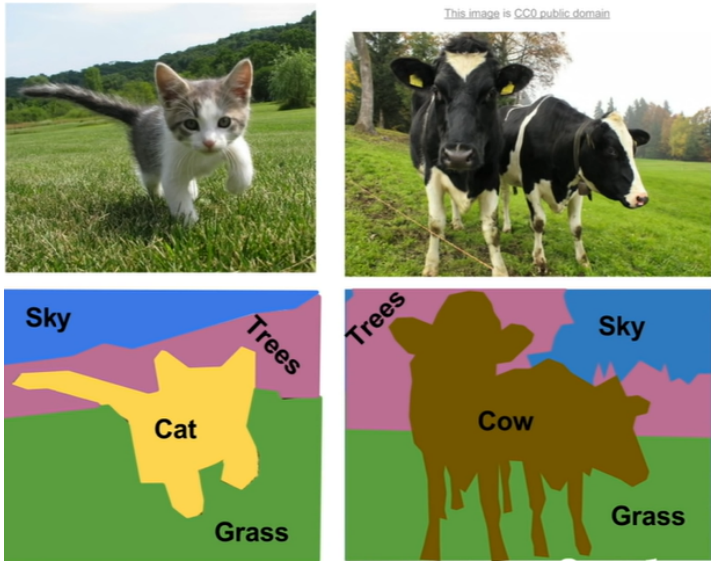


(圖片來源：wordpress上的網誌內容)

CNN應用於Segmentation

- Deep Learning前：使用RF當分類器
- 現今兩大主流：

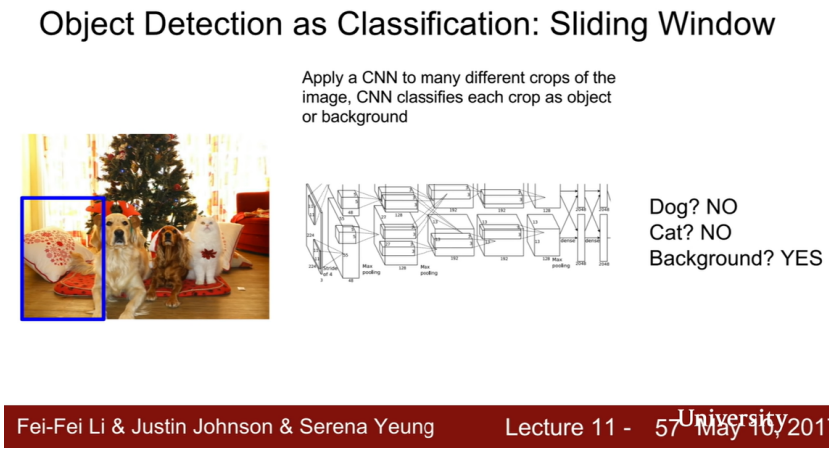
Dilated Convolutions、encoder-decoder architecture



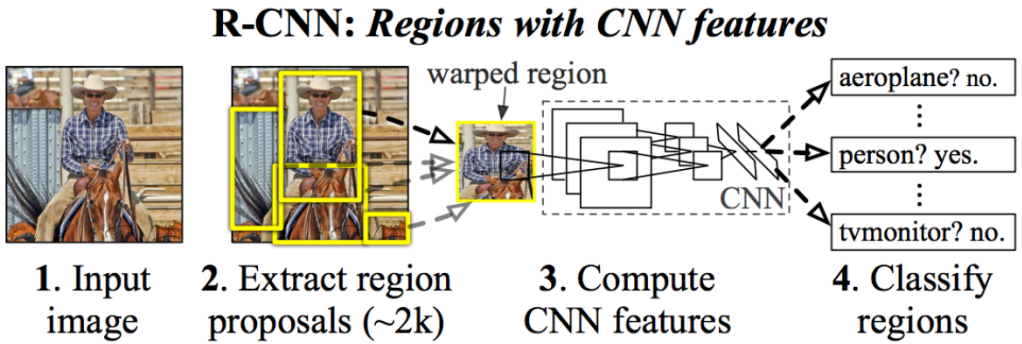
(圖片來源： Stanford的cs231n課程內容)

CNN應用於物體偵測

- Sliding Window → Selective Search



(圖片來源： Stanford的cs231n課程內容)

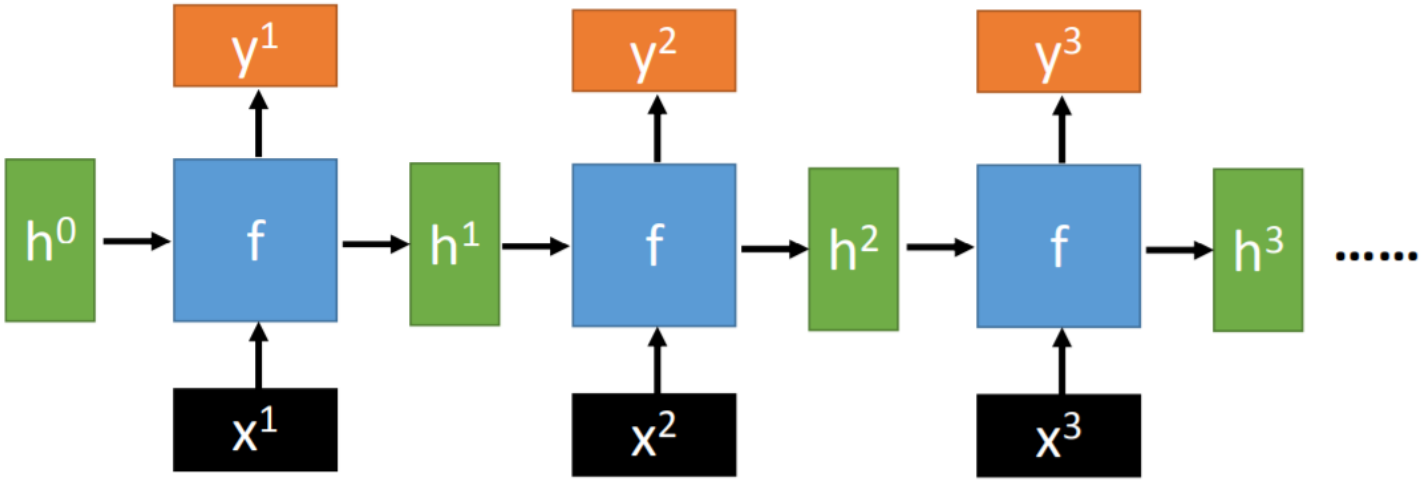


RNN

- 順序型資料

- Given function f: $h', y = f(h, x)$

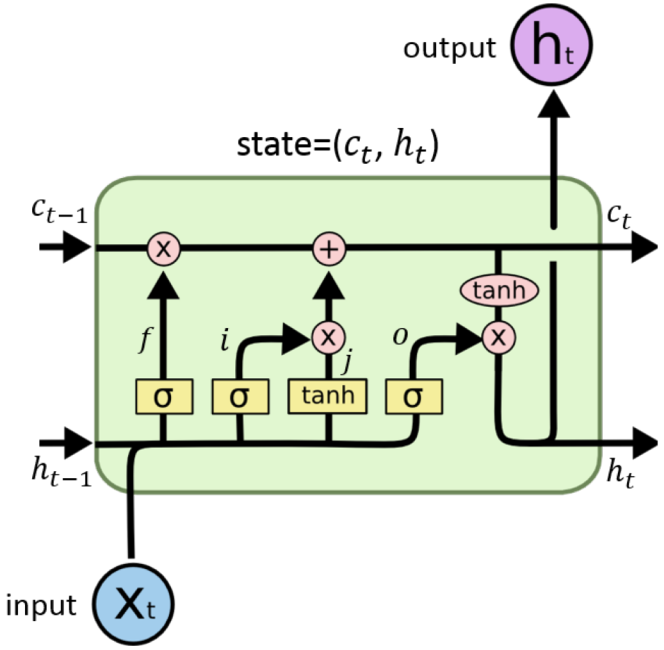
h and h' are vectors with the same dimension



(圖片來源：李宏毅老師上課PPT)

RNN的變型

- LSTM--解決梯度消失問題
- GRU--參數減少



$$gate\begin{cases} i = \sigma(W_i[h_{t-1}, x_t] + b_i) \\ f = \sigma(W_f[h_{t-1}, x_t] + b_f) \\ o = \sigma(W_o[h_{t-1}, x_t] + b_o) \end{cases}$$

$$j = tanh(W_j[h_{t-1}, x_t] + b_j)$$

$$update\begin{cases} c_t = f \odot c_{t-1} + i \odot j \\ h_t = o \odot \tanh(c_t) \end{cases}$$

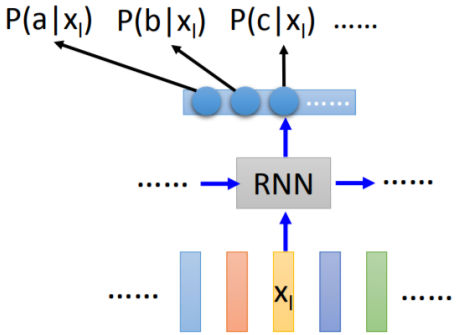
(圖片來源：李宏毅老師上課PPT)

RNN應用

- 常綜合其他模型一起使用

• Speech Recognition: CNN/LSTM/DNN + HMM

$$P(x,y) = P(y_1|start)\prod_{l=1}^{L-1} P(y_{l+1}|y_l)\, P(end|y_L)\prod_{l=1}^{L} P(x_l|y_l)$$



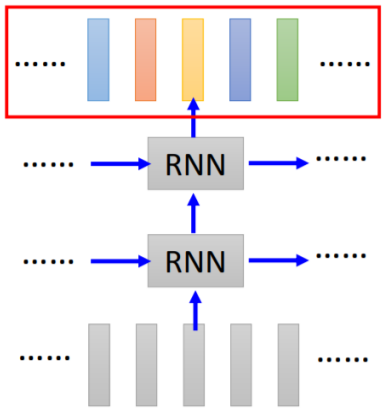
$$P(x_l|y_l) = \frac{P(x_l, y_l)}{P(y_l)}$$

RNN

$$= \frac{P(y_l|x_l)\cancel{P(x_l)}}{P(y_l)}$$

Count

• Semantic Tagging: Bi-directional LSTM + CRF/Structured SVM



$$w \cdot \phi(x,y)$$

Testing:

$$\tilde{y} = \arg\max_{y \in \mathbb{Y}} w \cdot \phi(x,y)$$

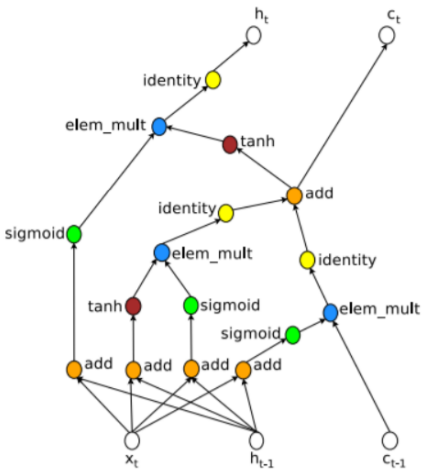
(圖片來源：李宏毅老師上課PPT)

RNN的架構應該長怎樣？

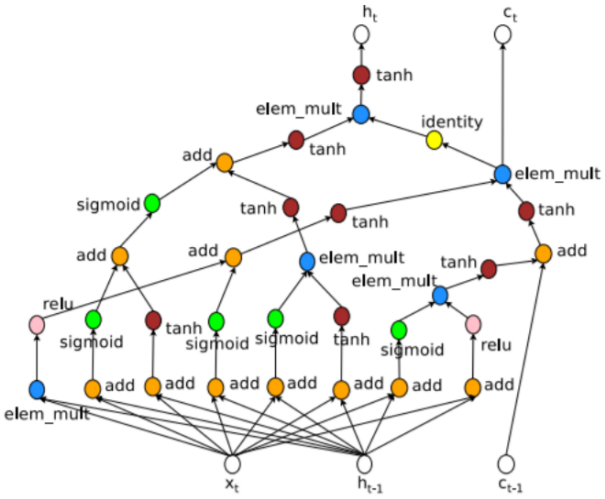
- Google的NAS架構學出來的結果

Neural Architecture Search with Reinforcement Learning

LSTM



From Reinforcement Learning



Computation Issue?

- Original version: 450 GPUs for 3-4 days (32,400-43,200 GPU hours).
- New version: Nvidia GTX 1080Ti GPU takes less than 16 hours.
- Main idea: forcing all child models to share weights to instead of training from scratch.

Hieu Pham, Melody Y. Guan, Barret Zoph, Quoc V. Le, Jeff Dean, "Efficient Neural Architecture Search via Parameter Sharing", arXiv, 2018

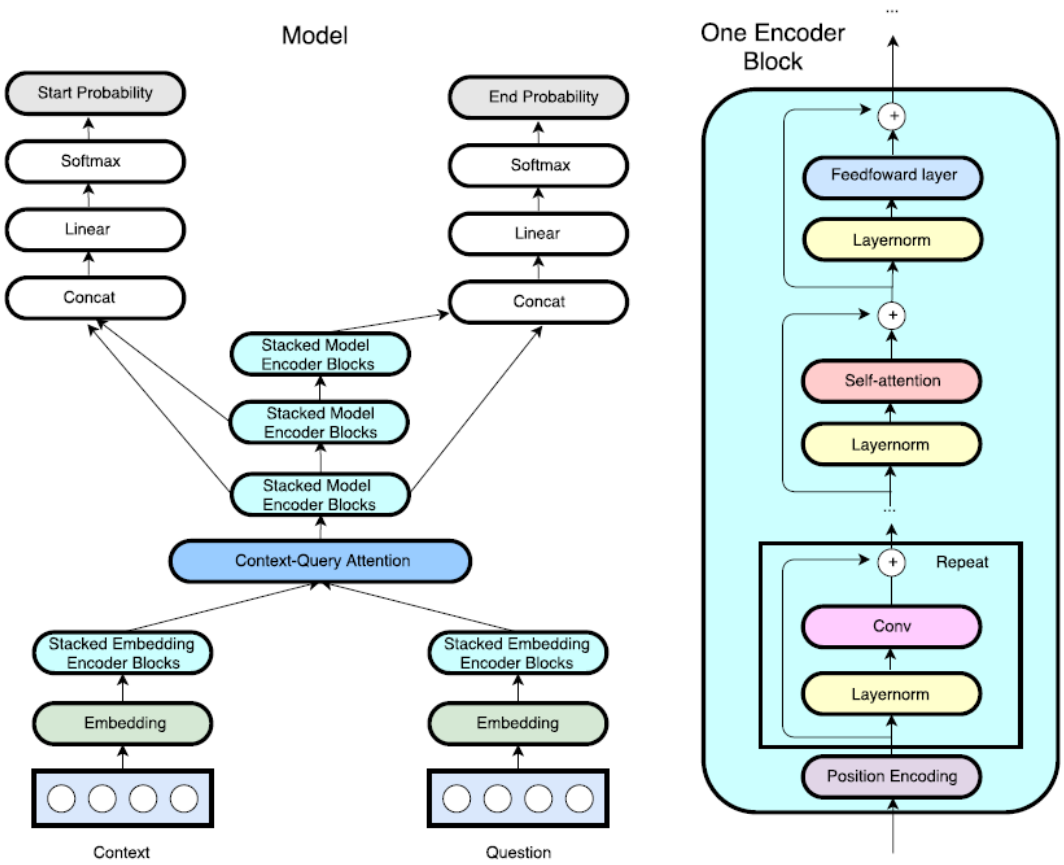
(圖片來源：李宏毅老師上課PPT)

第六部分　深度學習進階議題

- 機器如何看文章回答問題--QANet
- AlphaGo與專家訓練--增強學習 / 逆反式增強學習
- 機器創作--生成式對抗網路(GAN)

QANet

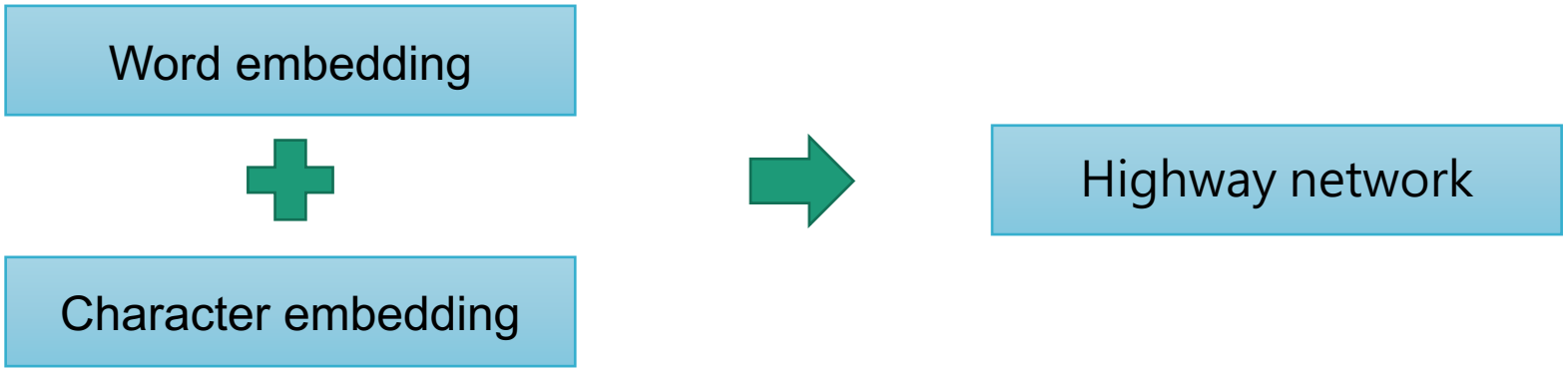
- Google於2018年初推出的架構，超過人類表現



(圖片來源：QANet論文)

QANet--Input Embedding Layer

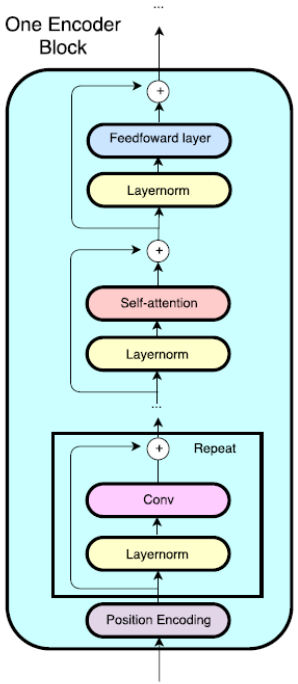
- Word embedding (字詞關聯)
- Character embedding (錯別字問題)
- Highway network (調節資訊量)



(圖片來源：QANet論文)

QANet--Embedding Encoder Layer

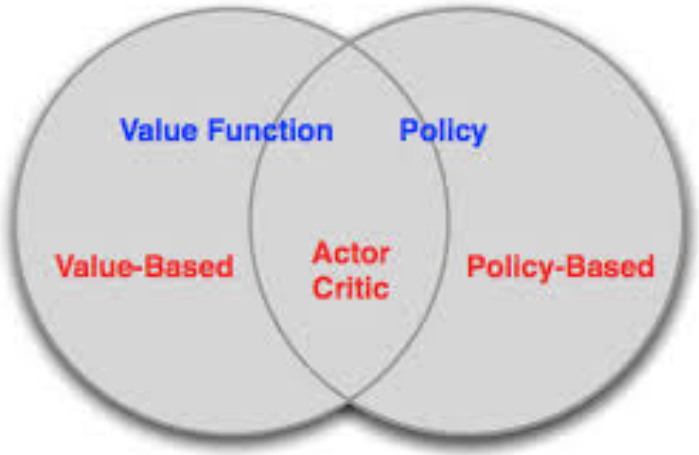
- Depthwise separable convolutions (局部資訊)
- Self-attention (全局資訊)
- Feed-forward-layer (加速)



(圖片來源：QANet論文)

增強學習

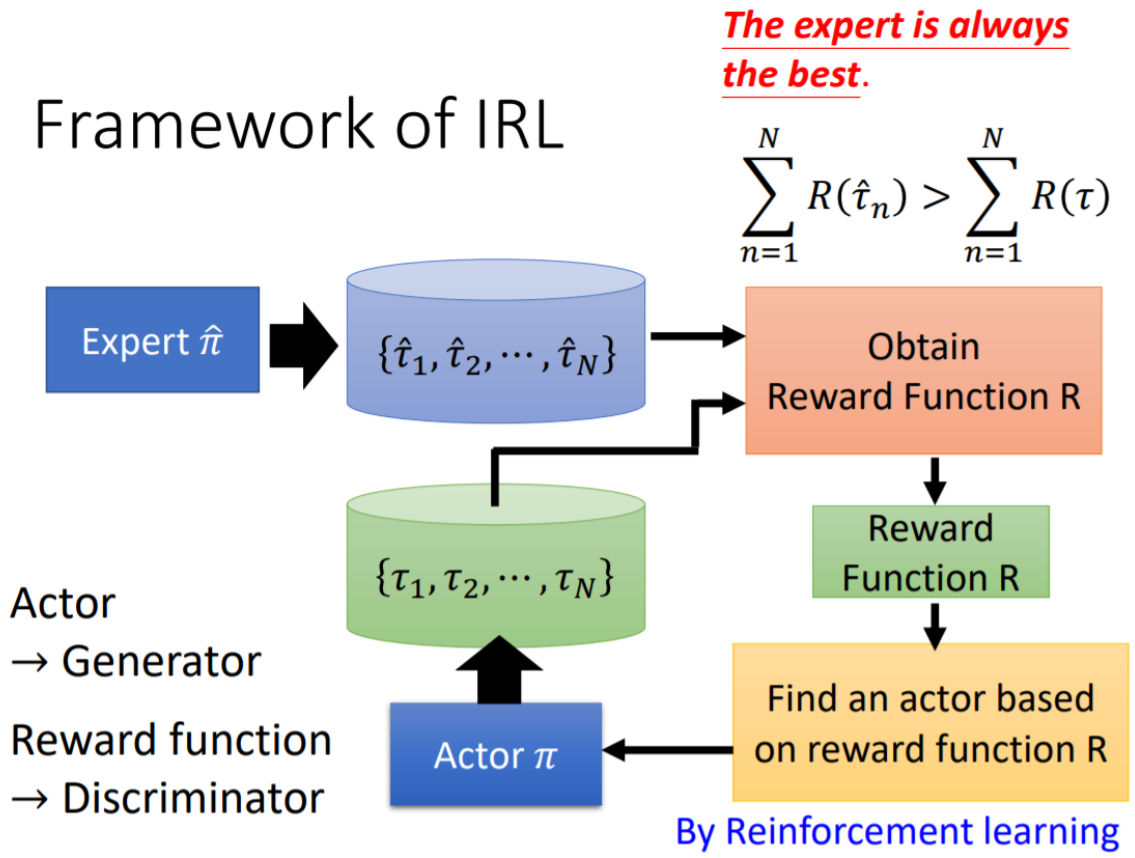
- Q-Learning、PPO → Actor-critic → A3C



(圖片來源：QANet論文)

逆反式增強學習

- 學習專家知識

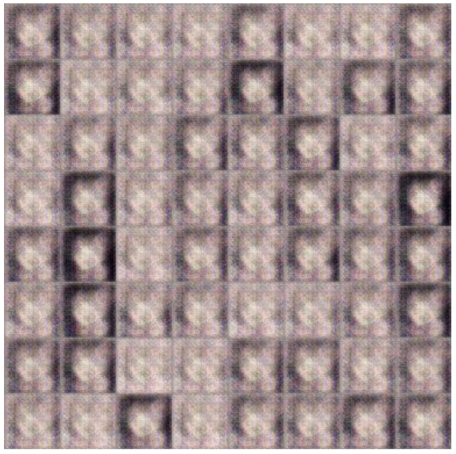


(圖片來源：李宏毅老師上課PPT)

GAN

- 機器生成模擬資料

100次



1000次

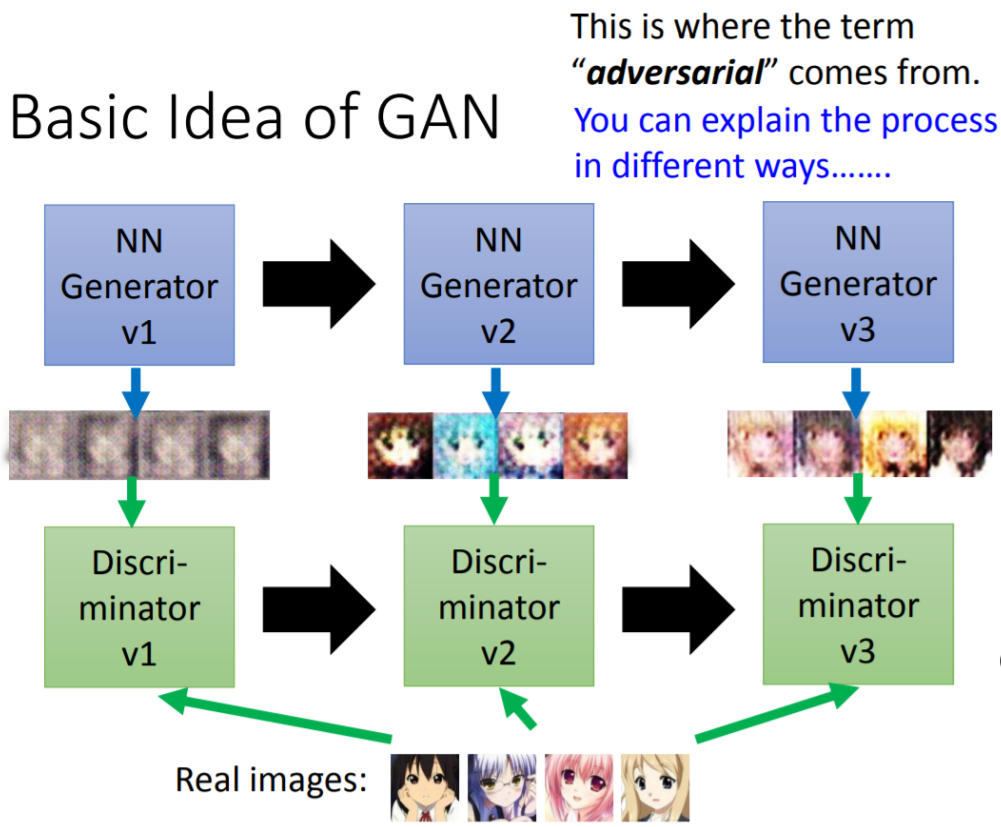
5000次

50000次

(圖片來源：李宏毅老師上課PPT)

GAN

- 警察小偷的對抗



(圖片來源：李宏毅老師上課PPT)

問題與討論