國泰大數據競賽 2020

許劭廷 - 交大統研所
高季伶 - 交大統研所
蔡濬安 - 交大統研所
傅琦佳 - 交大統研所

1. 關於國泰大數據競賽

類似Kaggle競賽

- 給定資料集以及問題(Supervised)
- 以客觀指標作為排名首要依據

問題介紹

- 人口學資料
- 過往保險資料、金融資料
- 目標：預測哪些人會保「重大疾病險」，為二元問題

評分標準

- 針對二元問題所產生的指標，假設模型能夠算出各筆樣本為1的機率(抑或是信心)。
- AUC：ROC線下面積
- ROC(receiver operating characteristic curve)：

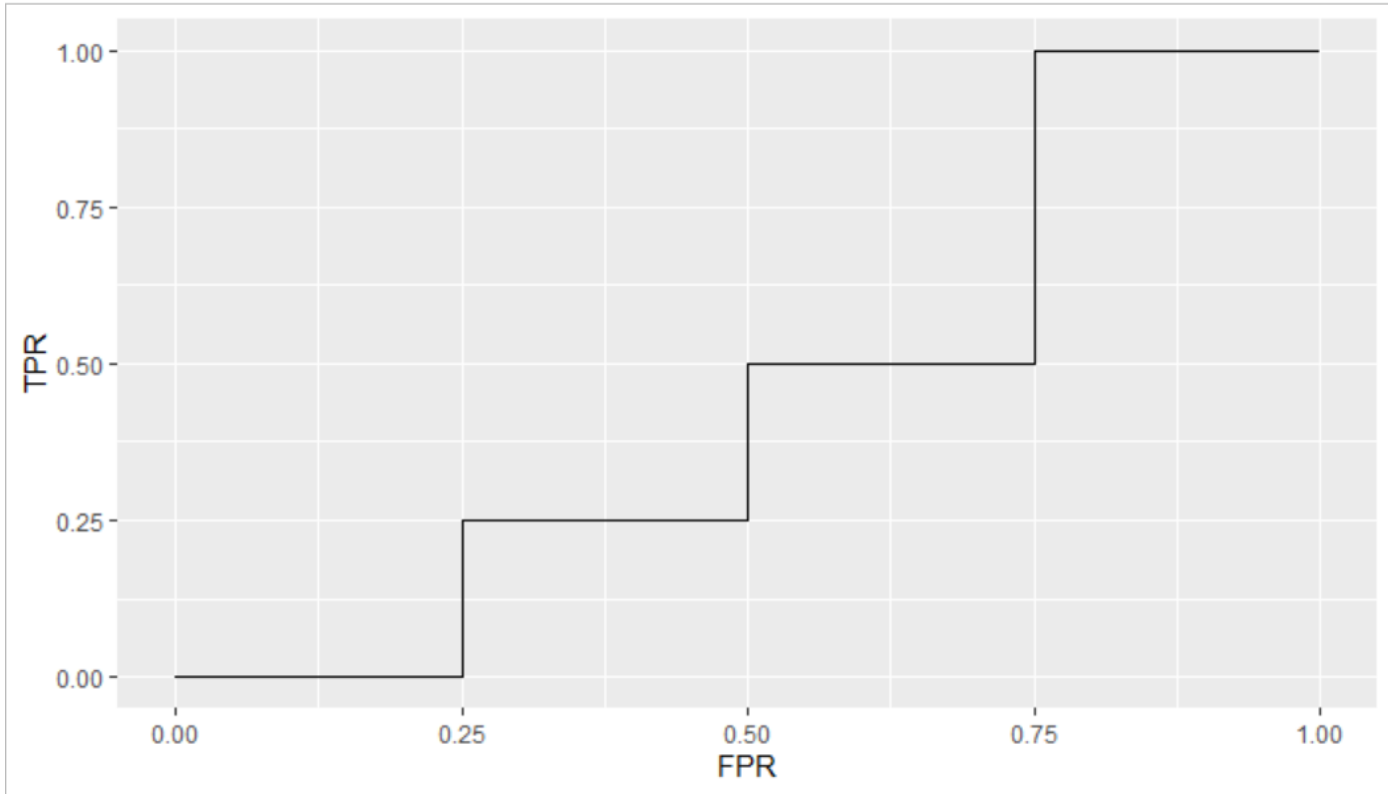
將同一模型每個閾值(Threshold)的(FPR, TPR) 座標都畫出來

- FPR = FP/(FP+TN) = FP/#(實際是N)
- TPR = TP/(TP+FN) = TP/#(實際是P)

		真實值		總數
		p	*n*	
預測輸出	*p′*	真陽性 (TP)	偽陽性 (FP)	P′
	n′	偽陰性 (FN)	真陰性 (TN)	N′
總數		P	N	

ROC

Threshold = 0.7	p	n
p’	25	25
n’	75	75
Threshold = 0.5	**p**	**n**
p’	50	50
n’	50	50
Threshold = 0.3	**p**	**n**
p’	75	100
n’	25	0



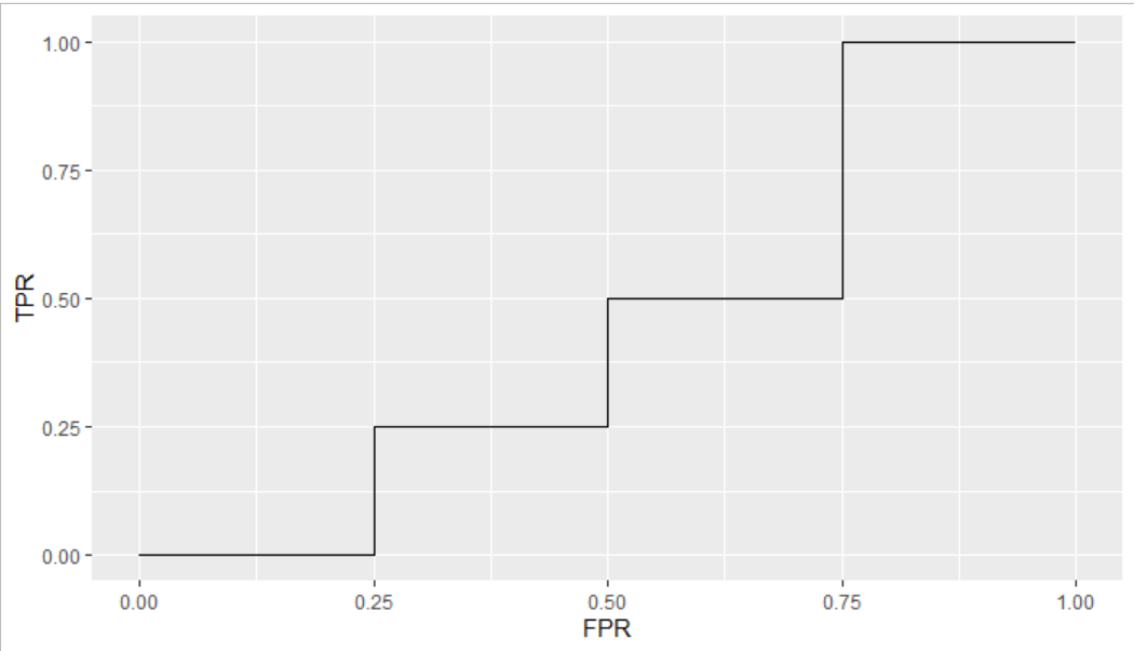
- 隨著Threshold愈小，愈多筆資料被預測成True，FPR以及TPR也隨之變大

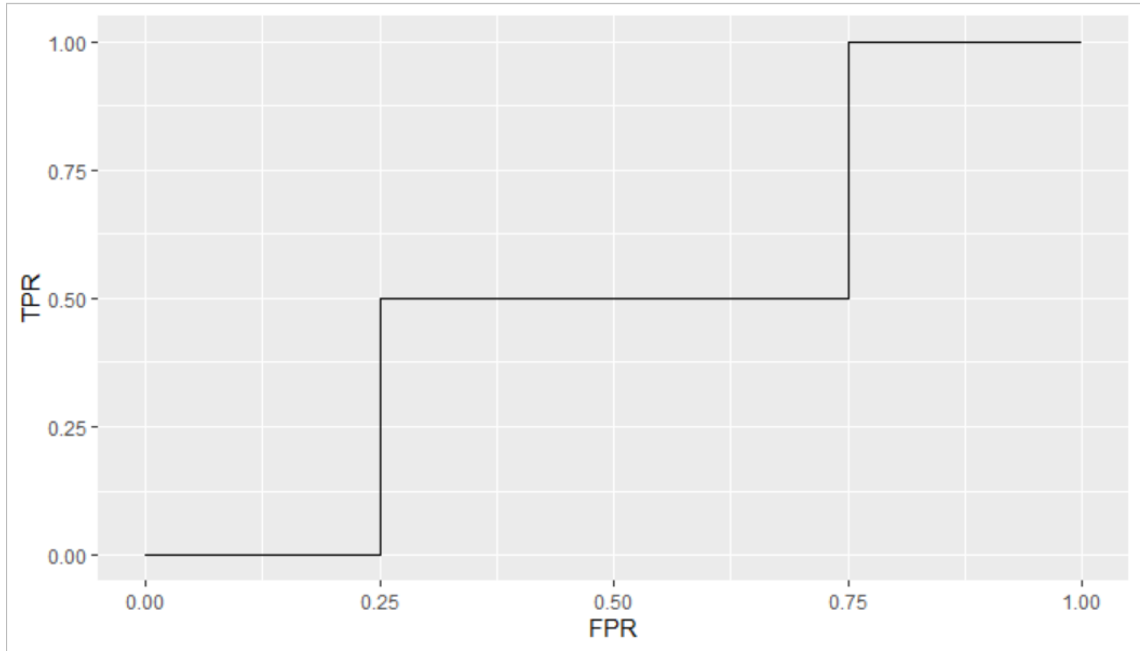
ROC的特性

- Threshold愈小，FPR以及TPR跟著愈高 -> ROC為一條非遞減線

不同模型下：

1. ROC愈早衝高，代表在相對小的FPR下，我們就能有高的TPR
2. ROC愈早衝高，則AUC愈大





AUC

- 結論：AUC愈大 -> 模型預測準確率愈好

- 使用AUC目的：

AUC是以0, 1資料被正確預測的**比例**做為評判標準

-> 不因imbalnce data而對哪邊資料有所偏頗

Public & Private

- 每天有兩次上繳答案的機會

並計算Public AUC成績(部分測試資料集)

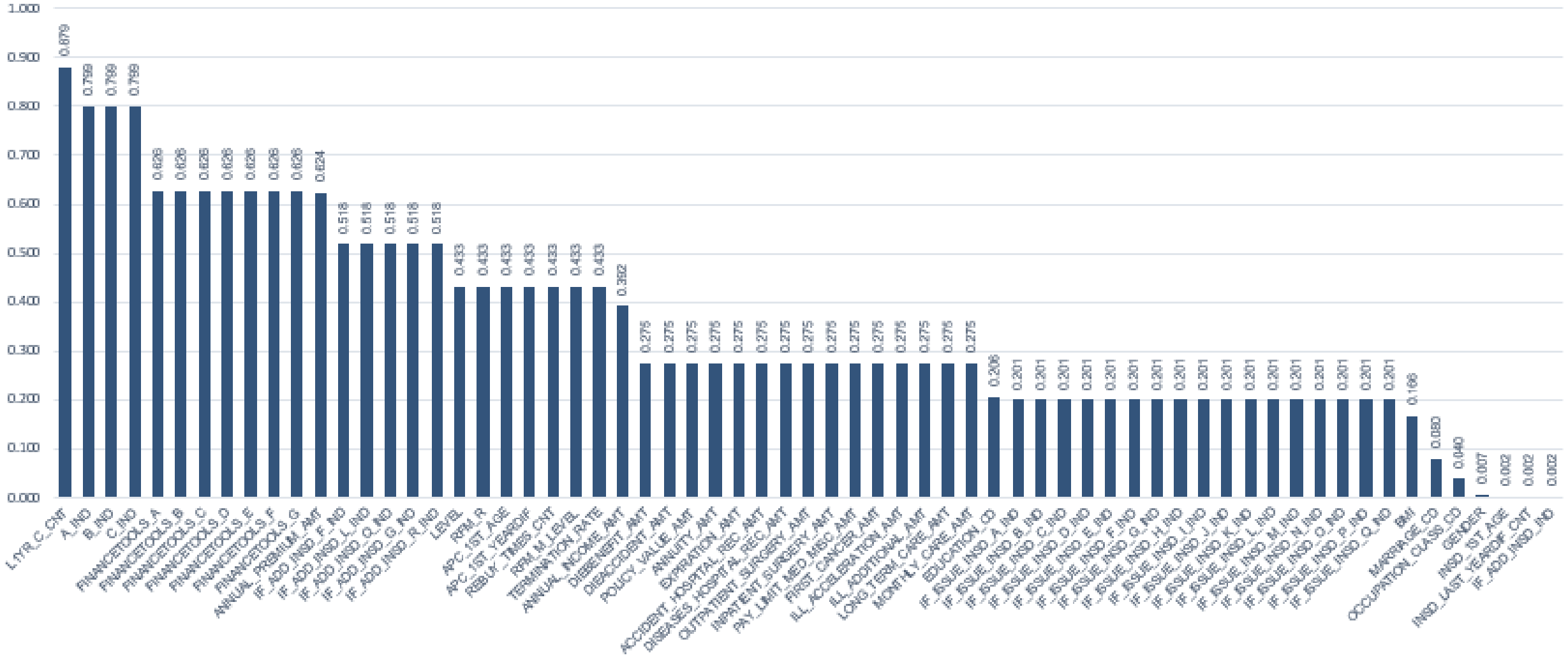
- 最後結算以Private AUC成績為主(所有測試資料集)

2. 資料介紹

資料維度

- Train：100,000筆
- Test：150,000筆
- Feature數量：131
- 缺失值的feature數量：65
- 保「重大疾病險」的比例：2%有保、98%沒保

缺失值比例


1.000
0.900
0.800
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0.000
0.879
0.799
0.799
0.799
0.626
0.626
0.626
0.626
0.626
0.626
0.626
0.624
0.518
0.518
0.518
0.518
0.518
0.433
0.433
0.433
0.433
0.433
0.433
0.433
0.392
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.275
0.206
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.201
0.166
0.080
0.040
0.007
0.002
0.002
0.002
L_IYR_C_CNT
A_IND
B_IND
C_IND
FINANCETOOLS_A
FINANCETOOLS_B
FINANCETOOLS_C
FINANCETOOLS_D
FINANCETOOLS_E
FINANCETOOLS_F
FINANCETOOLS_G
ANNUAL_PREMIUM_AMT
IF_ADD_INSD_F_IND
IF_ADD_INSD_L_IND
IF_ADD_INSD_Q_IND
IF_ADD_INSD_G_IND
IF_ADD_INSD_R_IND
LEVEL
RFM_R
APC_1ST_AGE
RSBUY_1ST_YEAROF
RFM_M_LEVEL
TERMINATION_RATE
ANNUAL_INCOME_AMT
DISEBENEFIT_AMT
DISEACCIDENT_AMT
POLICY_VALUE_AMT
ANNUITY_AMT
EXPIRATION_AMT
ACCIDENT_HOSPITAL_REC_AMT
DISEASES_HOSPITAL_REC_AMT
OUTPATIENT_SURGERY_AMT
INPATIENT_SURGERY_AMT
PAY_LIMIT_MED_MISC_AMT
FIRST_CANCER_AMT
ILL_ACCELERATION_AMT
ILL_ADDITIONAL_AMT
LONG_TERM_CARE_AMT
MONTHLY_CARE_AMT
EDUCATION_CD
IF_ISSUE_INSD_A_IND
IF_ISSUE_INSD_B_IND
IF_ISSUE_INSD_C_IND
IF_ISSUE_INSD_D_IND
IF_ISSUE_INSD_E_IND
IF_ISSUE_INSD_F_IND
IF_ISSUE_INSD_G_IND
IF_ISSUE_INSD_H_IND
IF_ISSUE_INSD_I_IND
IF_ISSUE_INSD_J_IND
IF_ISSUE_INSD_K_IND
IF_ISSUE_INSD_L_IND
IF_ISSUE_INSD_M_IND
IF_ISSUE_INSD_N_IND
IF_ISSUE_INSD_O_IND
IF_ISSUE_INSD_P_IND
IF_ISSUE_INSD_Q_IND
BMI
MARRIAGE_CD
OCCUPATION_CLASS_CD
GENDER
INSD_1ST_AGE
INSD_LAST_YEAROF_CNT
IF_ADD_INSD_IND

3. 資料清洗

(1) 資料清理 (Data Cleaning)

① 紀錄缺失值資訊

② 以決策樹方式補缺失值

- 建模：缺失值變量 ~ 其他沒有缺失值的變量
- 以模型預測有缺失的筆項

(2)資料整合（Data Integration）

- 由於資料乾淨，並不需要多個dataset做合併或是同個變量下不同尺度間的統一。
- 使用程式檢查，並無重複之ID，無須根據時序重新整合資料。

(3)資料轉換（Data Transformation）

- 尺度變換：樹結構不需要做轉換
- 刪減變數：選擇不刪除變數

刪除變數的原因

- 有影響力相似(colinear)的變量
- 有因果關係的變量
- 舉「薪資~是否為理工學院 + 性別」為例：

人數	Male	Female
是工學院	80人	20人
不是工學院	20人	80人

薪水	Male	Female
是工學院	$10	$10
不是工學院	$5	$5

Male	Female
10*0.8+5*0.2 = 9	10*0.2+5*0.8 = 6

刪除變數可能錯失的

- 交互作用

人數	Male	Female
是工學院	80人	20人
不是工學院	20人	80人

薪水	Male	Female
是工學院	$10	$5
不是工學院	$5	$10

Male	Female
10*0.8+5*0.2 = 9	5*0.2+10*0.8 = 9

是工學院	不是工學院
10*0.8+5*0.2 = 9	5*0.2+10*0.8 = 9

4. 建立模型

(1) 非平衡資料問題

- SMOTE

1. 找出與陽性個體 $\mathbf{x}_i$ 的最近的 k 個陽性鄰點 (k-nearest neighbors)
2. 在 k 個鄰點中隨機選擇一個，稱作 $\mathbf{x}_j$，我們會利用該鄰點用來生成新樣本
3. 計算 $\mathbf{x}_i$ 與 $\mathbf{x}_j$ 的差異 $\Delta = \mathbf{x}_j - \mathbf{x}_i$
4. 產生一個 0 – 1 之間的隨機亂數 η
5. 生成新的樣本點 $\mathbf{x}_i^{(new)} = \mathbf{x}_i + \eta\Delta$

- 使用不一樣的loss function: AUC、log-loss

(2) XGBoost

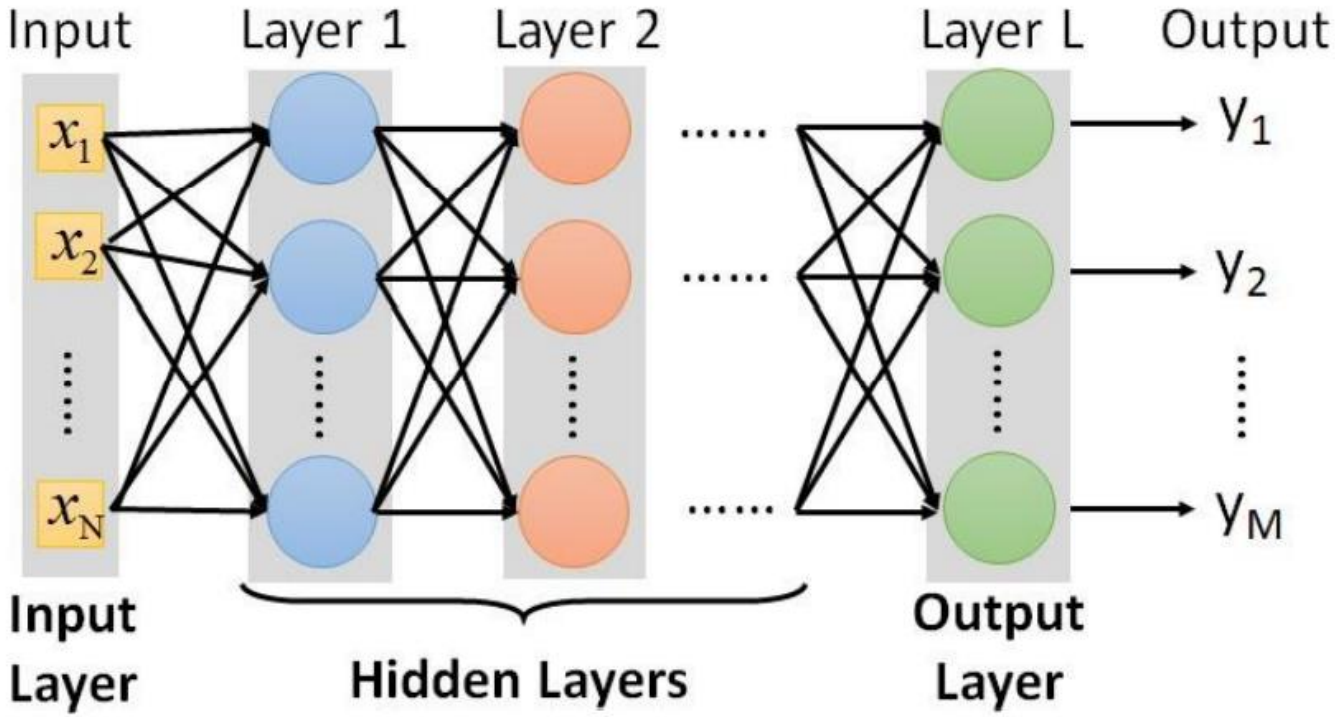
- 為Boosting算法
- 每次迭代為預測前一棵樹的loss

-> 確保迭代過程中，下次會比上次好

- 加入許多找樹的優化，使得繁瑣計算得以高效實現

(3) ANN類神經網路

- Universal approximation theorem：類神經網路架構可以逼近任意函數



最陡下降法、Chain rule

- 想辦法找到最低的loss function函數值發生點

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \gamma \nabla f\left(\boldsymbol{x}^{(t)}\right)$$

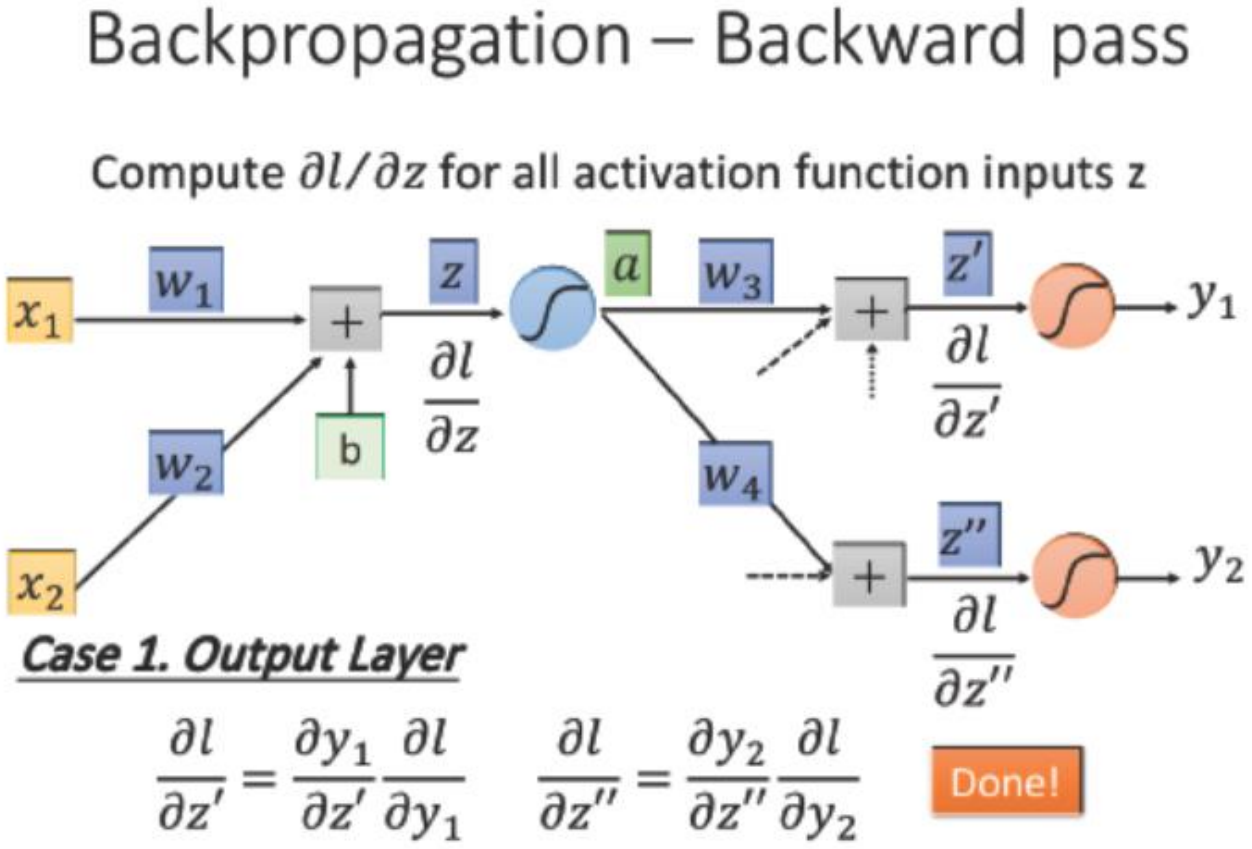
- 微分怎麼算?

根據不同的層數

不一樣的activate functi

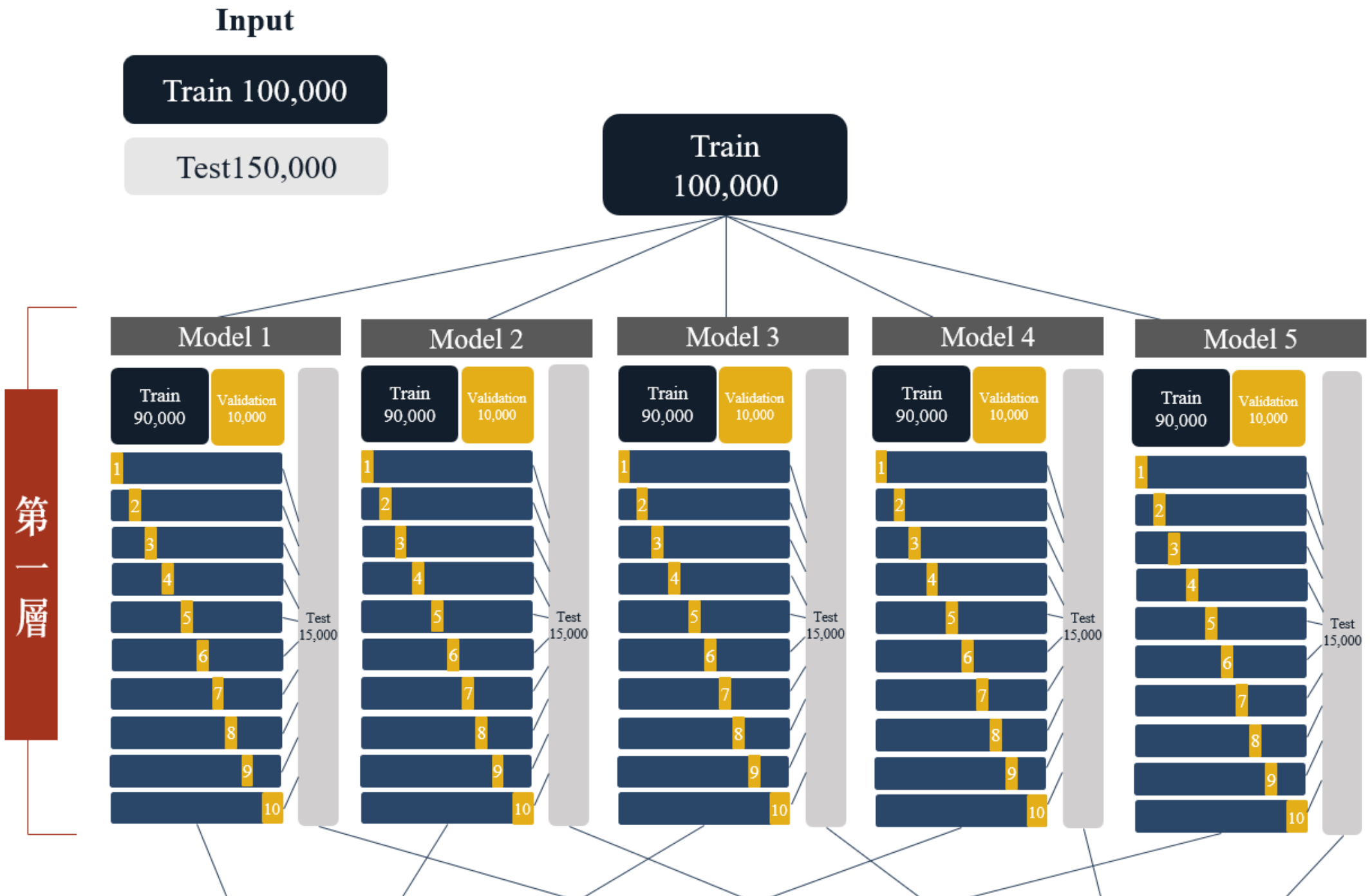
似乎每層參數找不到

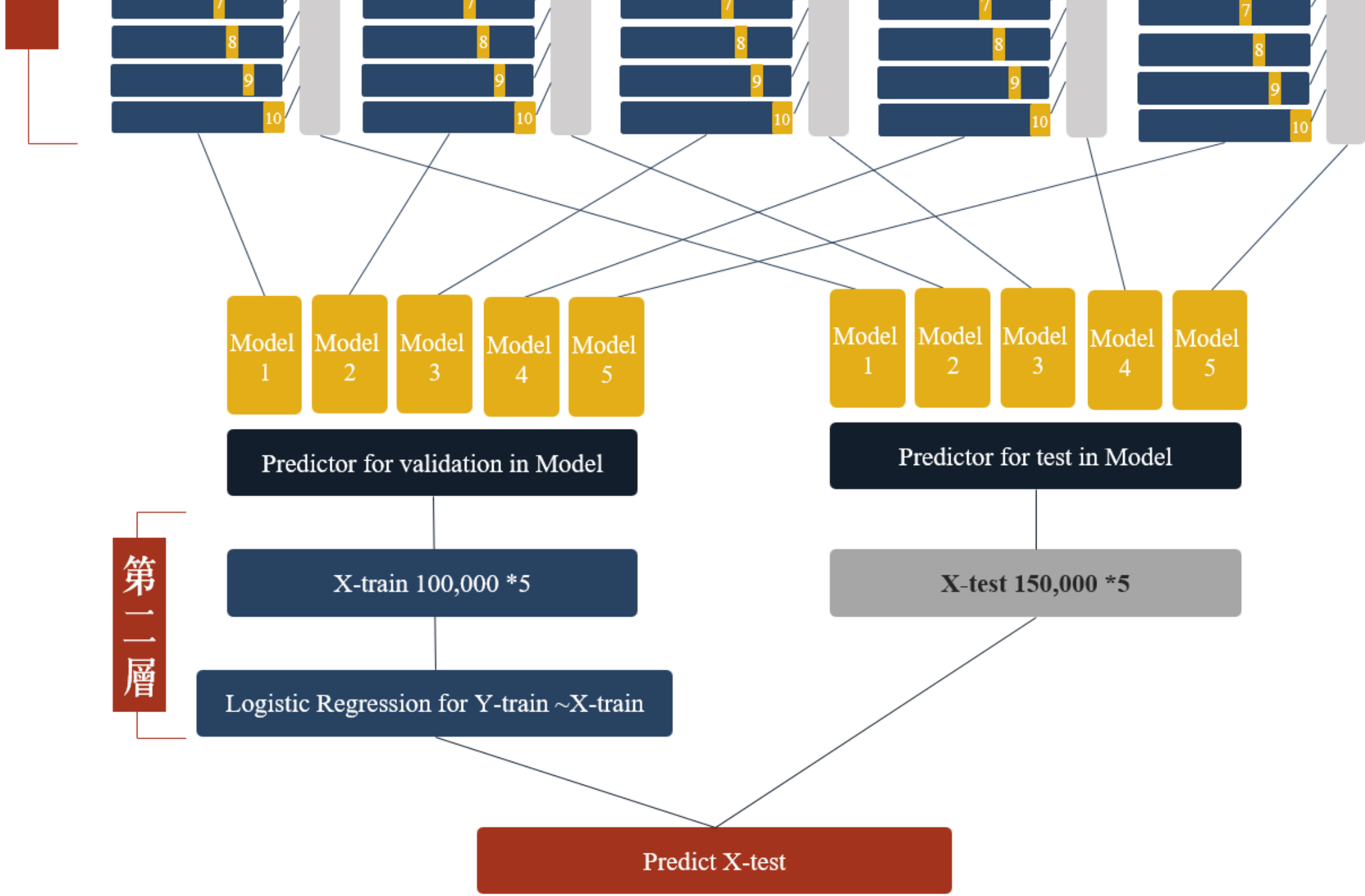
固定公式



(4) Stacking結合模型

- 類似Bagging概念
- 以類似投票方式調整各筆資料的預測值


Input
Train 100,000
Test150,000
Train
100,000
第一層
Model 1
Model 2
Model 3
Model 4
Model 5
Train
90,000
Validation
10,000
Test
15,000
1
2
3
4
5
6
7
8
9
10


7
8
9
10
Model 1
Model 2
Model 3
Model 4
Model 5
Model 1
Model 2
Model 3
Model 4
Model 5
Predictor for validation in Model
Predictor for test in Model
第二層
X-train 100,000 *5
X-test 150,000 *5
Logistic Regression for Y-train ~X-train
Predict X-test

5. 結論

AUC和排名

- Public AUC到達0.850789，為9/244名
- Private AUC到達0.846202，為21/244名