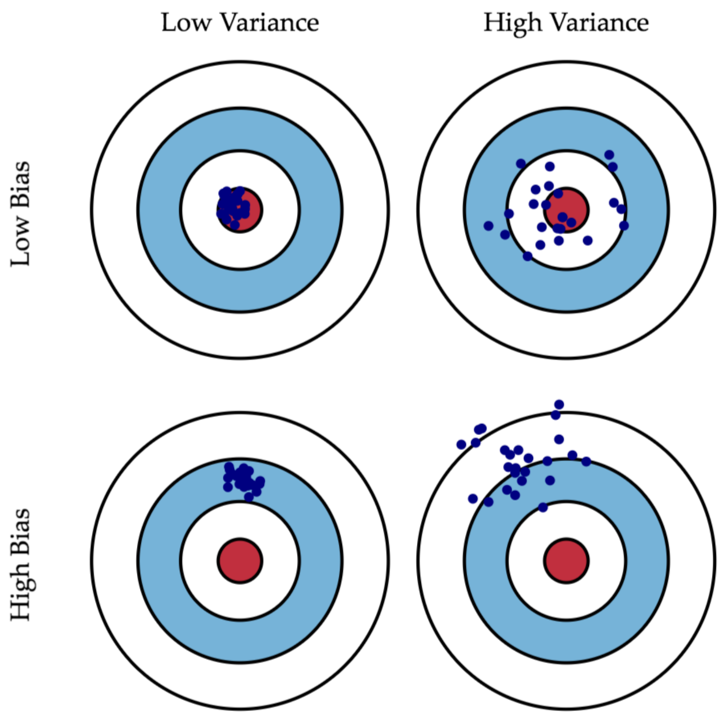
Ensemble Learning

講者：Isaac

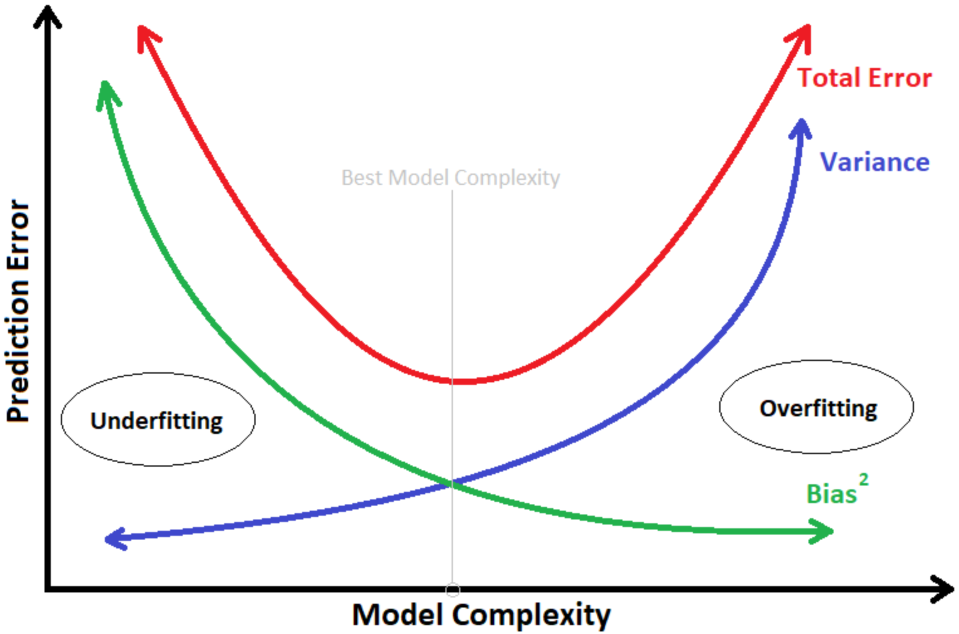
Outline

- What’s ensemble learning
- Bagging method
- Boosting method
  - Adaboost, XGBoost

Bias–variance tradeoff



Bias–variance tradeoff



What's ensemble learning

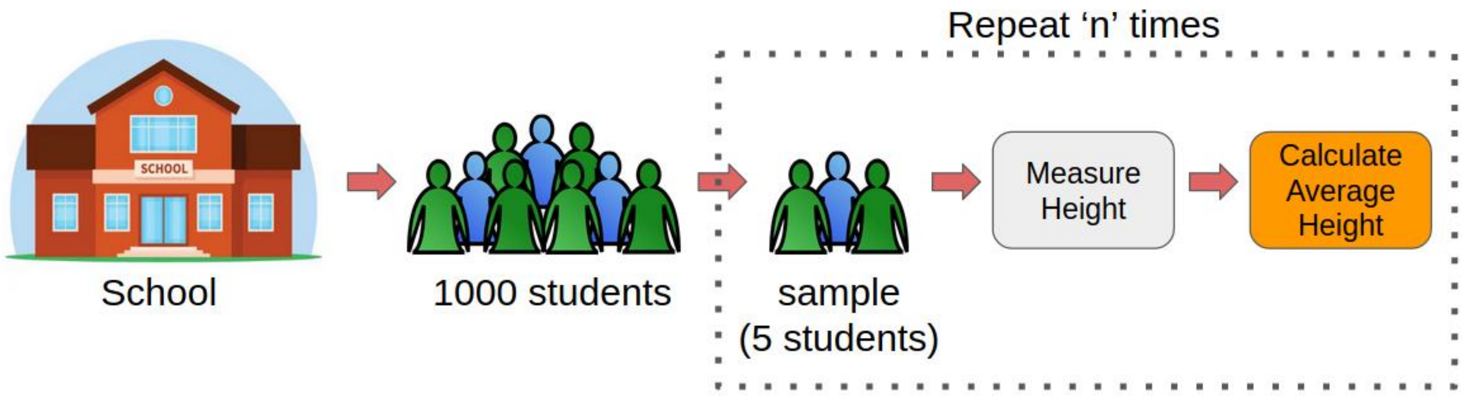
- Ensemble models in machine learning combine the decisions from multiple models to improve the overall performance
- Two common ensemble learning method
  - Bagging
  - Boosting

Bagging

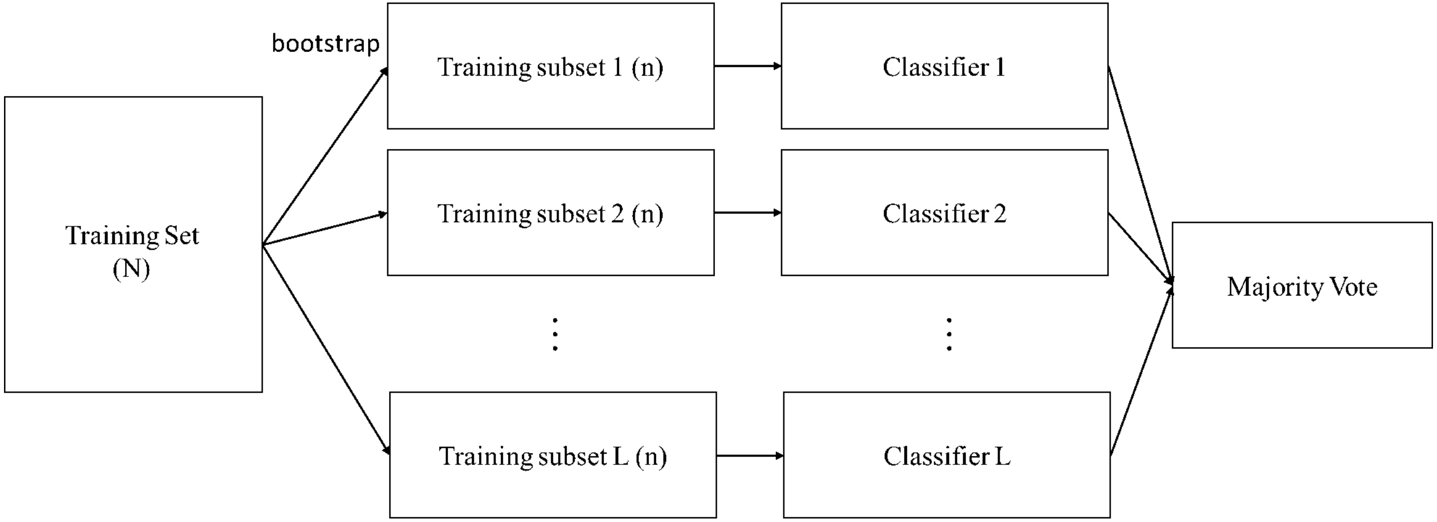
- Bagging = bootstrapping + aggregating
- Bootstrapping is a method to help decrease the variance of the classifier and reduce overfitting
  - resampling data from the training set
- Bagging common algorithm
  - Random forest

What's bootstrapping

- In statistics, bootstrap sampling is a method that involves drawing of sample data repeatedly with replacement from a data source to estimate a population parameter.



Bagging Big Picture



Out of Bag sample

Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Sunny	Hot	High	Weak	Yes
Windy	Cold	Low	Weak	Yes

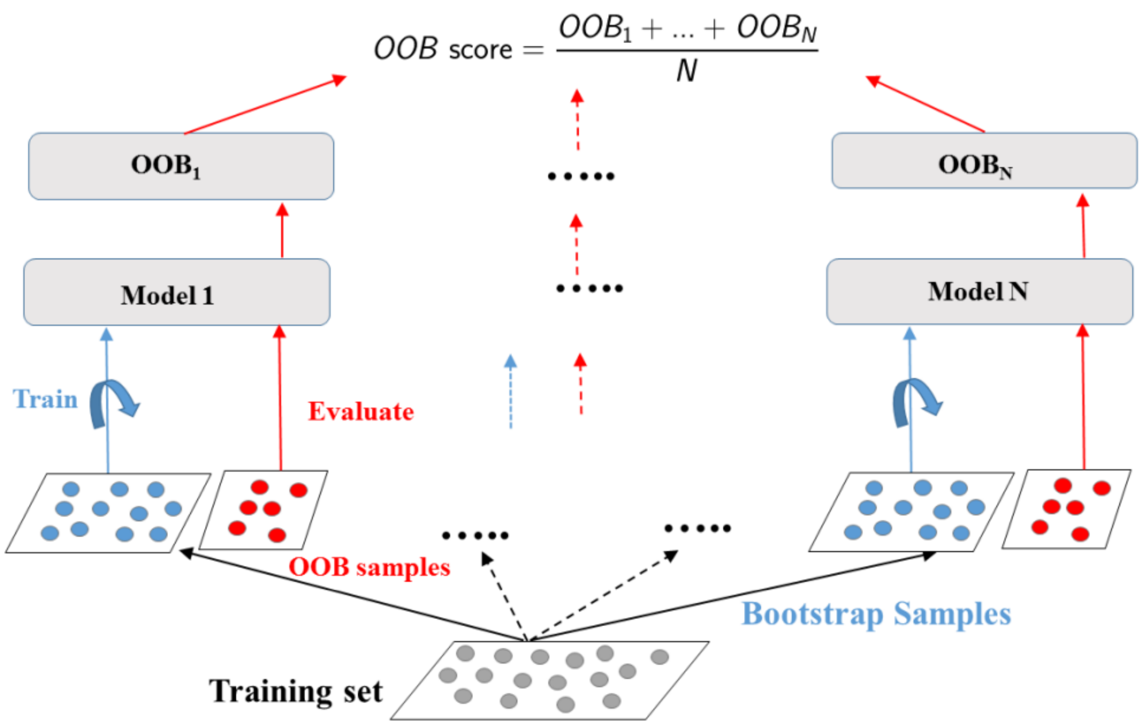
Bootstrap sample

Out of Bag sample

Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Sunny	Hot	High	Weak	Yes
Windy	Cold	Low	Weak	Yes

Out of Bag sample

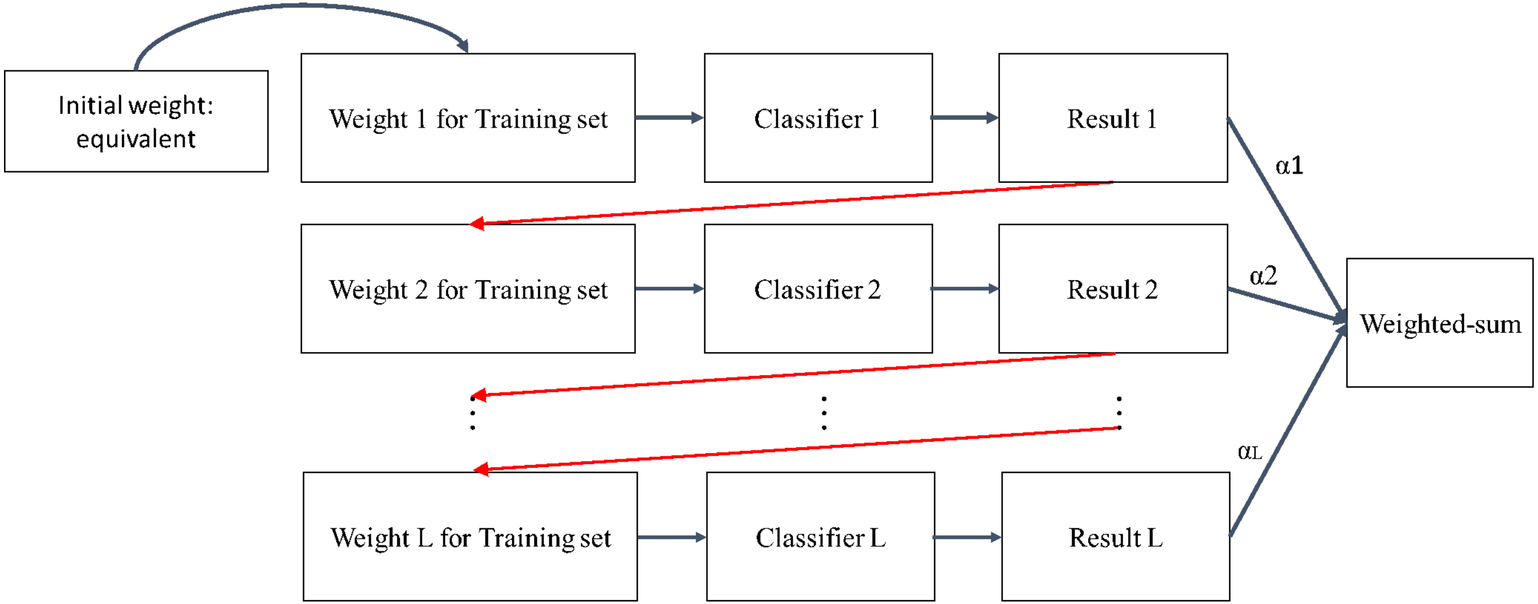
Out of Bag sample



Boosting

- Create a sequence of weak model and each of them try to reduce bias
  - AdaBoost, Gradient Boosting

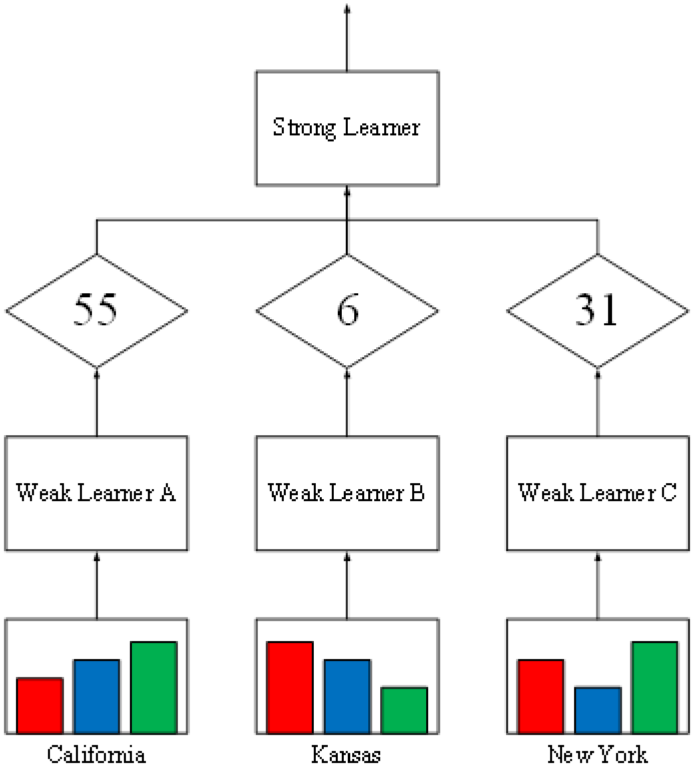
Boosting Big Picture



Adaboost

- AdaBoost(Adaptive Boosting) is a machine learning algorithm
  - can be used in conjunction with many other types of learning algorithms to improve performance
  - output of the other learning algorithms is combined into a weighted sum that represents the final output of the boosted classifier

Adaboost



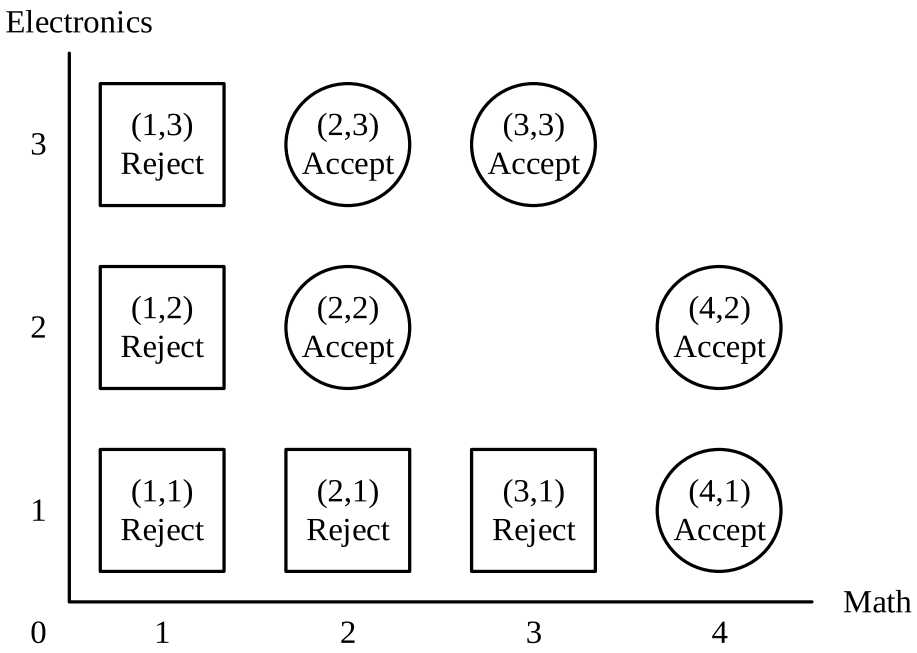
Strong Learner
55
6
31
Weak Learner A
Weak Learner B
Weak Learner C
California
Kansas
New York

Example

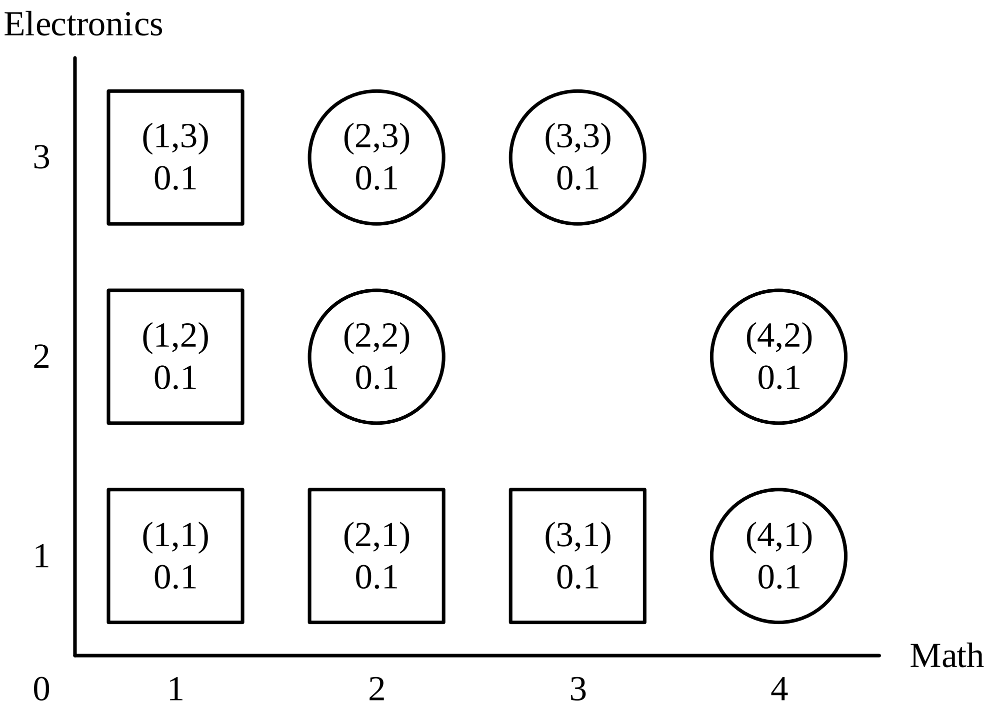
教授	建議	缺點
A	數學差的學生再怎麼教也沒有用!應該“大幅提高”數學的錄取門檻。	過於偏激!可能會導致招生不足。
B	數學的內容在所有的學科中都會用到，只要“稍微提高”數學的錄取門檻即可。	可能會招收不到電子學極好但是數學較差的學生。
C	半導體產業龐大，電子學是基礎，所以只要“稍微提高”電子學的錄取門檻即可。	可能會招收不到數學極好但是電子學較差的學生。

Example

ID	數學 (Math)	電子 (Electronics)	類別
1	2	2	Accept
2	2	3	Accept
3	3	3	Accept
4	4	1	Accept
5	4	2	Accept
6	1	1	Reject
7	1	2	Reject
8	1	3	Reject
9	2	1	Reject
10	3	1	Reject

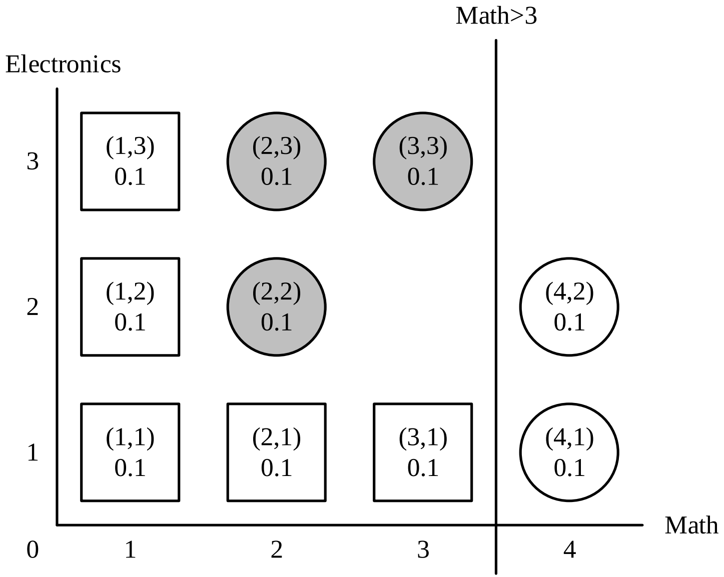


Step 1 - Initialization



Step 2 – iterative A

- 假設A教授的錄取標準是數學成績必須大於三級分才可以錄取，也就是：Math > 3，如下圖所示：



Step 2 – iterative A

- calculate error rate
  - ε_m=0.1+0.1+0.1=0.3
- calculate model weight
  - $\alpha_m = \frac{1}{2}\ln\left(\frac{1\text{-}0.3}{0.3}\right) = 0.424$

$$\varepsilon_m = \sum_{i=1}^{N} w_{m,i} \mid h_m(x_i) \neq y_i$$

$$\alpha_m = \frac{1}{2}\ln\left(\frac{1\text{-}\varepsilon_m}{\varepsilon_m}\right)$$

Step 2 – iterative A

- reweight data

$$w_{m+1,i} = \frac{1}{Z_{m,i}} w_{m,i} \cdot e^{I_m \cdot \alpha_m},$$

$$Z_{m,i} = \sum_{i=1}^{N} w_{m,i}$$

$$I_m = \begin{cases} 1 & , \quad if\,(h_m(x_i) \neq y_i) \\ -1 & , \quad if\,(h_m(x_i) = y_i) \end{cases}$$

Step 2 – iterative A

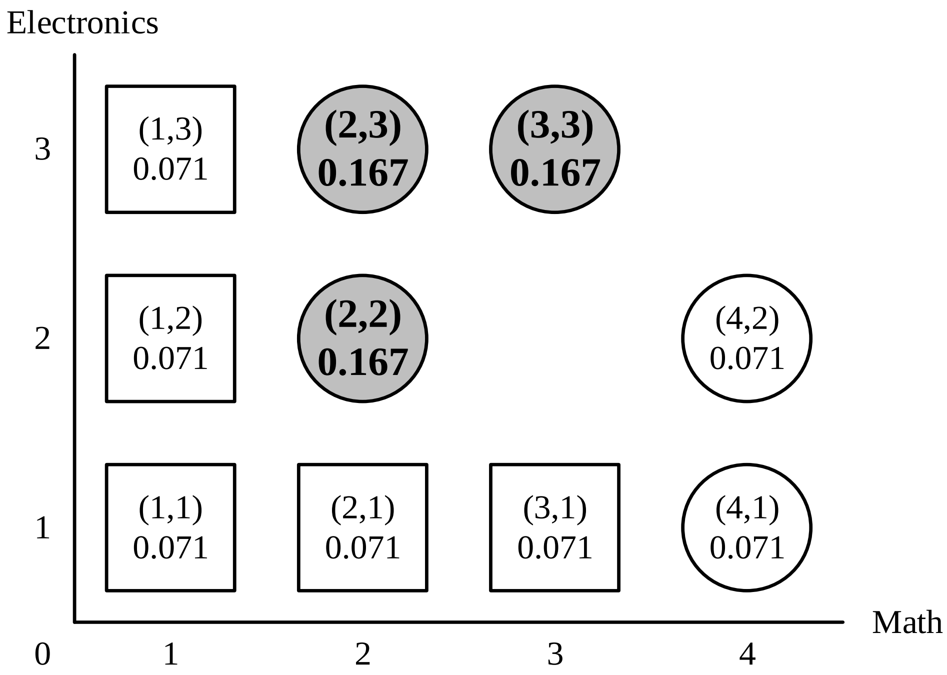
- **reweight data**

Data	$w_{m,i}$ (現在關注度)	$\mathrm{I}_m \cdot \alpha_m$ (參考度)	$w_{m,i} \cdot e^{\mathrm{I}_m \cdot \alpha_m}$	$Z_{m,i}$ (總關注度)	$w_{m+1,i}$ (未來關注度)
(2, 3)	0.1	0.424	0.153	0.914	**0.167**
(3, 3)	0.1	0.424	0.153	0.914	**0.167**
(2, 2)	0.1	0.424	0.153	0.914	**0.167**
(4, 2)	0.1	-0.424	0.065	0.914	**0.071**
(4, 1)	0.1	-0.424	0.065	0.914	**0.071**
(1, 3)	0.1	-0.424	0.065	0.914	**0.071**
(1, 2)	0.1	-0.424	0.065	0.914	**0.071**
(1, 1)	0.1	-0.424	0.065	0.914	**0.071**
(2, 1)	**0.1**	**-0.424**	**0.065**	**0.914**	**0.071**
(3, 1)	**0.1**	**-0.424**	**0.065**	**0.914**	**0.071**

$$Z_m = 0.153+0.153+0.153+0.065+0.065+0.065+0.065+0.065+0.065+0.065 = 0.914$$

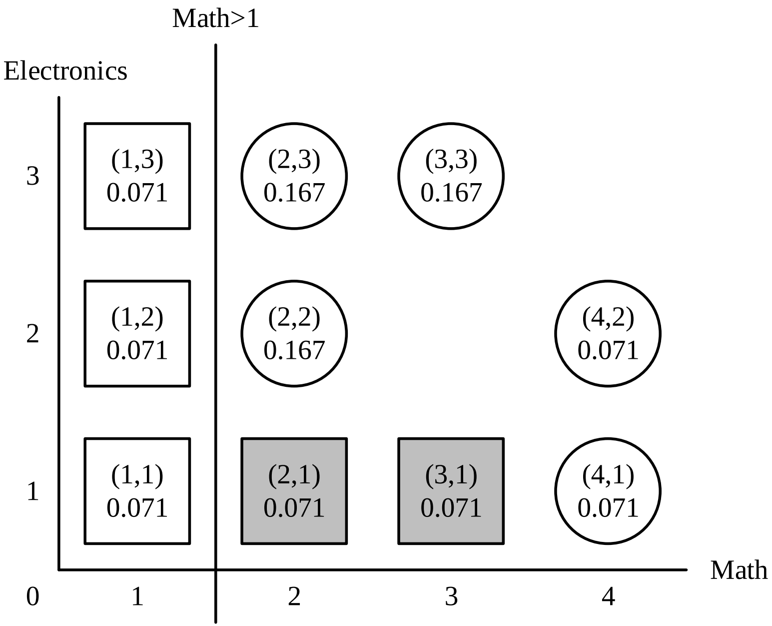
Step 2 – iterative A

- reweight data



Step 2 – iterative B

- 假設B教授的錄取標準是：數學成績必須大於1級分才可以錄取，也就是：Math > 1，如下圖所示：



Step 2 – iterative B

- calculate error rate
  - ε_m=0.071+0.071=0.142
- calculate model weight
  - $\alpha_m = \frac{1}{2}\ln\left(\frac{1-0.142}{0.142}\right) = 0.896$

$$\varepsilon_m = \sum_{i=1}^{N} w_{m,i} \mid h_m(x_i) \neq y_i$$

$$\alpha_m = \frac{1}{2}\ln\left(\frac{1-\varepsilon_m}{\varepsilon_m}\right)$$

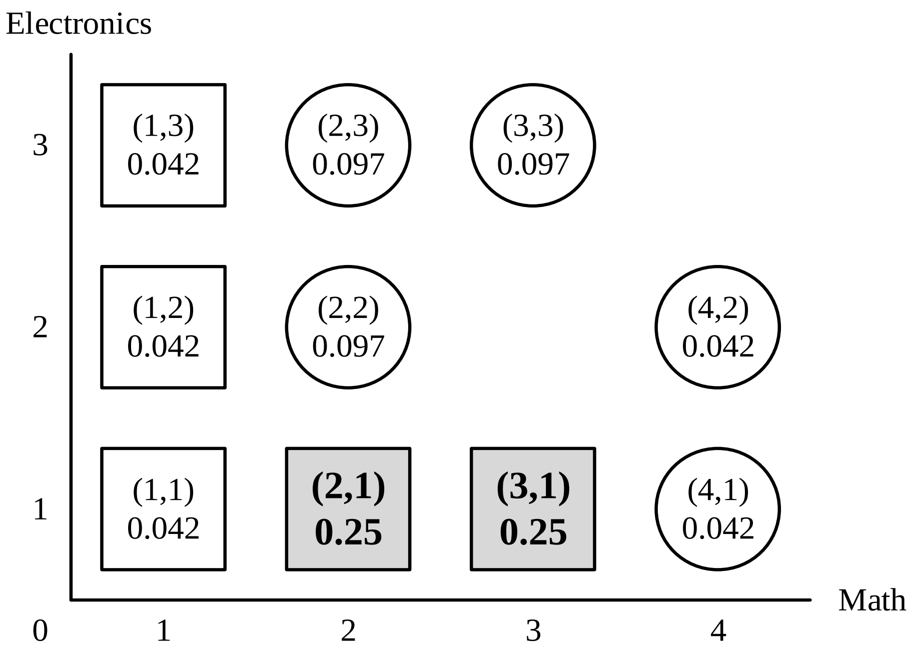
Step 2 – iterative B

- reweight data

Data	$w_{m,i}$ (現在關注度)	$\mathrm{I}_m \cdot \alpha_m$ (參考度)	$w_{m,i} \cdot e^{\mathrm{I}_m \cdot \alpha_m}$	$Z_{m,i}$ (總關注度)	$w_{m+1,i}$ (未來關注度)
(2, 3)	0.167	-0.896	0.068	0.7	**0.097**
(3, 3)	0.167	-0.896	0.068	0.7	**0.097**
(2, 2)	0.167	-0.896	0.068	0.7	**0.097**
(4, 2)	0.071	-0.896	0.029	0.7	**0.042**
(4, 1)	0.071	-0.896	0.029	0.7	**0.042**
(1, 3)	0.071	-0.896	0.029	0.7	**0.042**
(1, 2)	0.071	-0.896	0.029	0.7	**0.042**
(1, 1)	0.071	-0.896	0.029	0.7	**0.042**
(2, 1)	**0.071**	**0.896**	**0.175**	**0.7**	**0.25**
(3, 1)	**0.071**	**0.896**	**0.175**	**0.7**	**0.25**

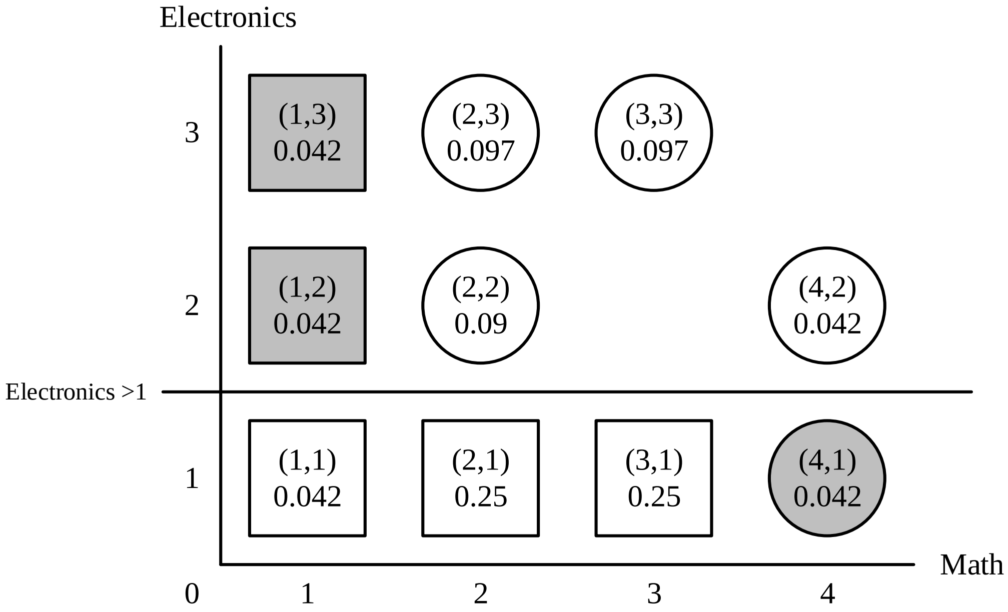
Step 2 – iterative B

- reweight data



Step 2 – iterative C

- 假設C教授的錄取標準是電子學成績必須大於1級分才可以錄取，也就是：Electronics > 1，如下圖所示：



Step 2 – iterative C

- calculate error rate
  - ε_m= 0.042+0.042+0.042=0.126
- calculate model weight
  - $$\alpha_m = \frac{1}{2}\ln\left(\frac{1-0.126}{0.126}\right) = 0.973$$

$$\varepsilon_m = \sum_{i=1}^{N} w_{m,i} \mid h_m(x_i) \neq y_i$$

$$\alpha_m = \frac{1}{2}\ln\left(\frac{1-\varepsilon_m}{\varepsilon_m}\right)$$

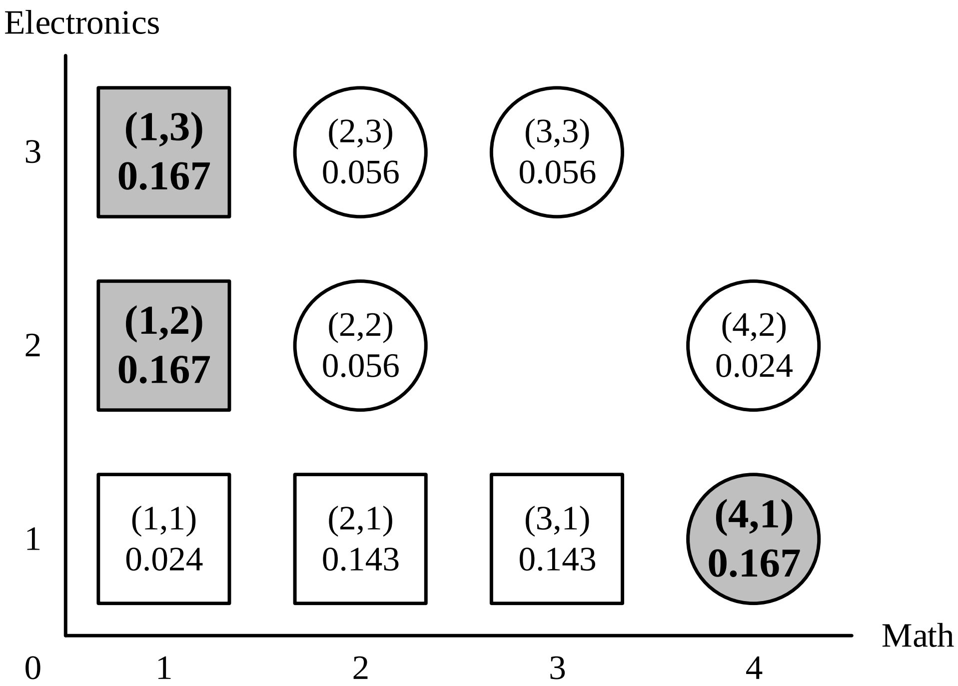
Step 2 – iterative C

- reweight data

Data	$w_{m,i}$ **(現在關注度)**	$\mathrm{I}_m \cdot \alpha_m$ **(參考度)**	$w_{m,i} \cdot e^{\mathrm{I}_m \cdot \alpha_m}$	$Z_{m,i}$ **(總關注度)**	$w_{m+1,i}$ **(未來關注度)**
(2, 3)	0.097	-0.973	0.037	0.661	**0.056**
(3, 3)	0.097	-0.973	0.037	0.661	**0.056**
(2, 2)	0.097	-0.973	0.037	0.661	**0.056**
(4, 2)	0.042	-0.973	0.016	0.661	**0.024**
(4, 1)	**0.042**	**0.973**	**0.11**	**0.661**	**0.167**
(1, 3)	**0.042**	**0.973**	**0.11**	**0.661**	**0.167**
(1, 2)	**0.042**	**0.973**	**0.11**	**0.661**	**0.167**
(1, 1)	0.042	-0.973	0.016	0.661	**0.024**
(2, 1)	0.25	-0.973	0.094	0.661	**0.143**
(3, 1)	0.25	-0.973	0.094	0.661	**0.143**

Step 2 – iterative C

- reweight data

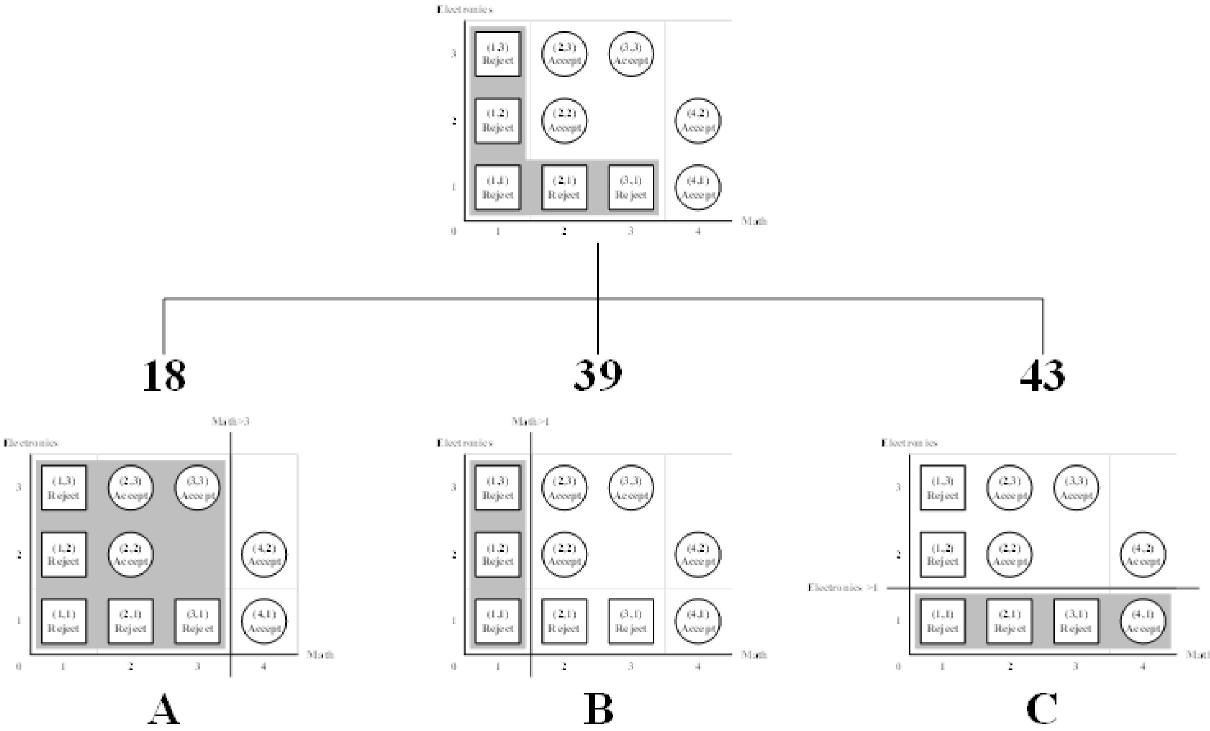


Step 3 – Combine models

- Combine all models

弱分類器	參考度 α	正規化	票數
A	0.424	$\frac{0.424}{0.424+0.896+0.973}=0.18$	$0.18\times100=18$
B	0.896	$\frac{0.896}{0.424+0.896+0.973}=0.39$	$0.39\times100=39$
C	0.973	$\frac{0.973}{0.424+0.896+0.973}=0.43$	$0.43\times100=43$

Step 3 – Combine models



Step 4 – Predict data

	Math	Electronics	類別
Jack	3	2	?

弱分類器	規則	預測類別	票數
A	Math > 3	Reject	18
B	Math > 1	Accept	39
C	Electronics > 1	Accept	43

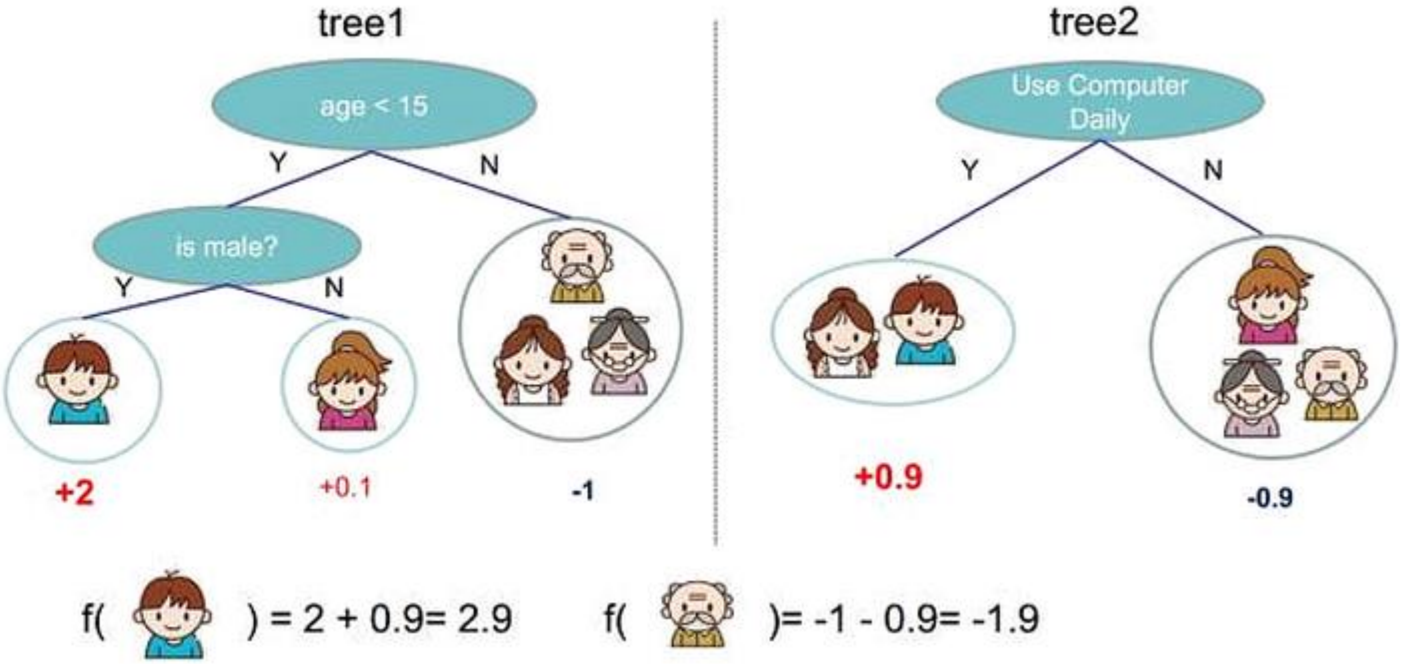
Step 4 – Predict data

預測類別	總得票數
Accept	39+43=82
Reject	18

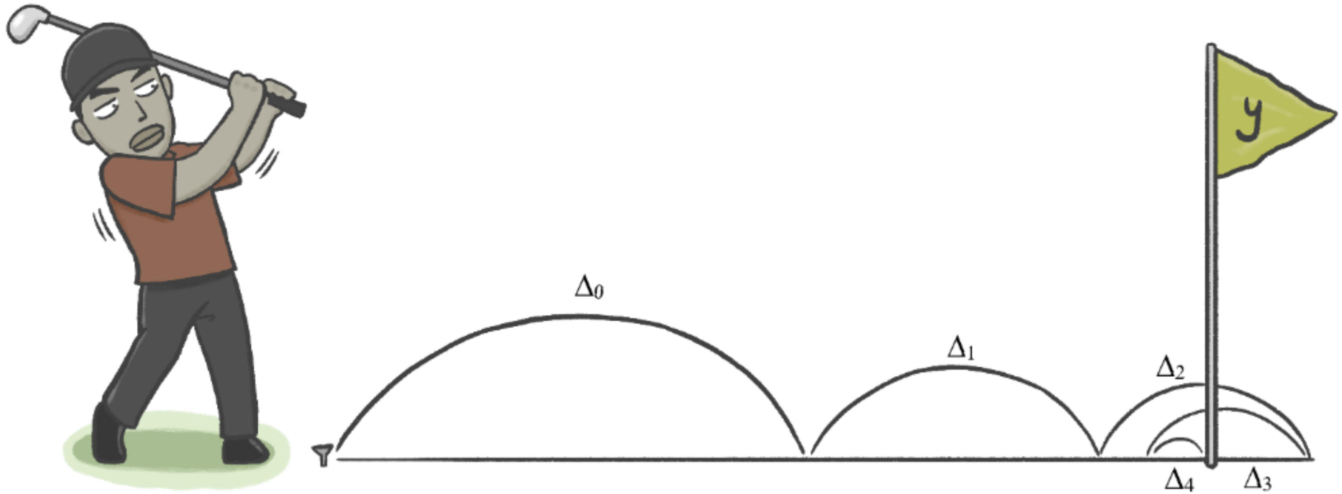
XGboost

- eXtreme Gradient Boosting(XGboost) is one of the famous machine learning algorithms
  - it is based on gradient boosting framework
  - push the extreme of the computation limits of machines to provide a scalable, portable and accurate library

Xgboost Illustration



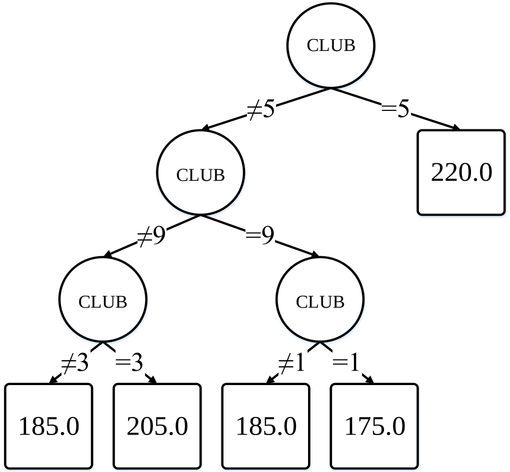
Xgboost Illustration



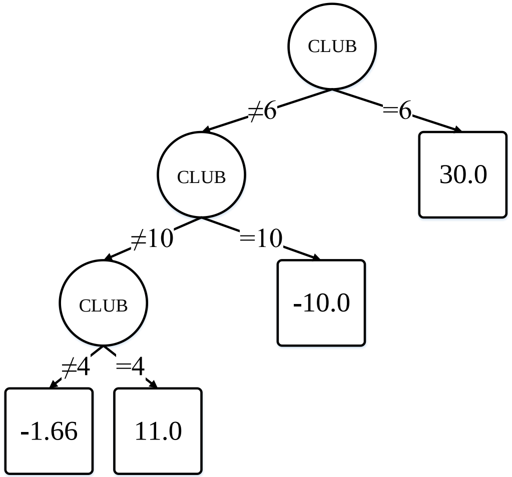
Xgboost Illustration

球桿編號	擊球距離
1	180.0
2	190.0
3	200.0
4	210.0
5	220.0
6	215.0
7	205.0
8	195.0
9	185.0
10	175.0

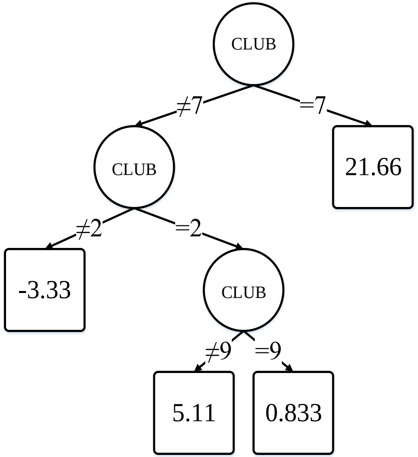
Xgboost Illustration



Tree 1

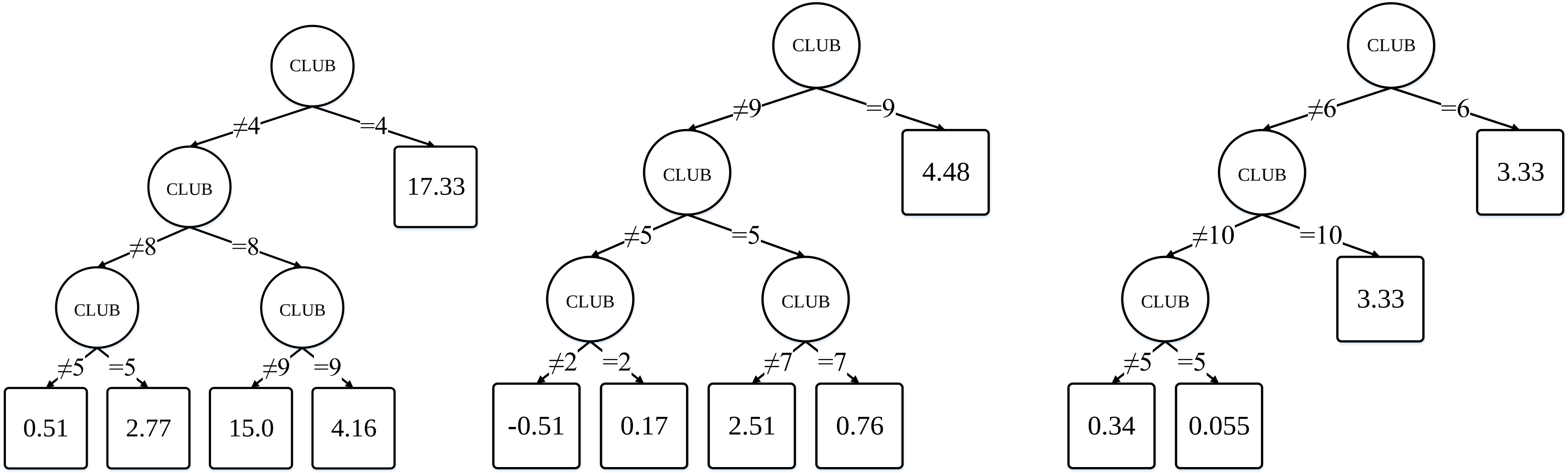


Tree 2



Tree 3

Xgboost Illustration



Tree 4 Tree 5 Tree 6

Xgboost Illustration

CLUB	Tree 1	Tree 2	Tree 3	Tree 4	Tree 5	Tree 6	預測值 (DIST)	實際值 (DIST)
1	185.0	-1.67	-3.33	0.52	-0.52	0.35	180.35	180.0

$$185.0+(-1.67)+(-3.33)+0.52+(-0.52)+0.35=180.35$$

- Model:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathcal{F}$$

Space of functions containing all Regression trees

- Model parameters:
  - structure of each tree and the score in the leaf

- Objective:

$$Obj = \underbrace{\sum_{i=1}^{n} l(y_i, \hat{y}_i)}_{\text{Training loss}} + \underbrace{\sum_{k=1}^{K} \Omega(f_k)}_{\text{Complexity of the Trees}}$$

- Solution:

$$
\begin{aligned}
\hat{y}_i^{(0)} &= 0 \\
\hat{y}_i^{(1)} &= f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\
\hat{y}_i^{(2)} &= f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\
&\dots \\
\hat{y}_i^{(t)} &= \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i)
\end{aligned}
$$

New function

Model at training round t

Keep functions added in previous round

- Solution:

$$
\begin{aligned}
Obj^{(t)} &= \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^{t} \Omega(f_i) \\
&= \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) + constant
\end{aligned}
$$

Goal: find f_t to minimize this

if we use square loss

$$
\begin{aligned}
Obj^{(t)} &= \sum_{i=1}^{n} \left(y_i - (\hat{y}_i^{(t-1)} + f_t(x_i))\right)^2 + \Omega(f_t) + const \\
&= \sum_{i=1}^{n} \left[2(\hat{y}_i^{(t-1)} - y_i) f_t(x_i) + f_t(x_i)^2\right] + \Omega(f_t) + const
\end{aligned}
$$

This is usually called residual from previous round

How to build tree ?

- Use the following index instead of gini index or entropy

建立分支前(***noSplit***)：
$$Obj_{noSplit} = -\frac{1}{2}\left[\frac{(G_L^2 + G_R^2)}{H_L + H_R + \lambda}\right] + \gamma T_{noSplit}$$

建立分支後(***split***)：
$$Obj_{split} = -\frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda}\right] + \gamma T_{split}$$

T：樹葉節點的數量
w：葉權值
λ, γ：係數
G_L, G_R：代價函數的一階導數
H_L, H_R：代價函數的二階導數

How to build tree ?

$$
\begin{aligned}
Gain \quad &= Obj_{noSplit} - Obj_{split} \\
&= \left\{ -\frac{1}{2}\left[\frac{(G_L^2 + G_R^2)}{H_L + H_R + \lambda} \right] + \gamma T_{noSplit} \right\} - \left\{ -\frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} \right] + \gamma T_{split} \right\} \\
&= -\frac{1}{2}\left[\frac{(G_L^2 + G_R^2)}{H_L + H_R + \lambda} \right] + \frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} \right] + \gamma T_{noSplit} - \gamma T_{split} \\
&= \frac{1}{2}\left[-\frac{(G_L^2 + G_R^2)}{H_L + H_R + \lambda} + \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} \right] + \gamma T_{noSplit} - \gamma T_{split} \\
&= \frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \left(T_{split} - T_{noSplit} \right)
\end{aligned}
$$

Advantages of ensemble methods

- More accurate prediction results
- Stable and more robust model
  - multiple models is always less noisy than the individual models
- Ensemble models can be used to capture the linear as well as the non-linear relationships in the data

Disadvantages of ensemble methods

- Reduction in model interpret-ability
- Computation and design time is high
  - not good for real time applications

Bagging V.S. Boosting