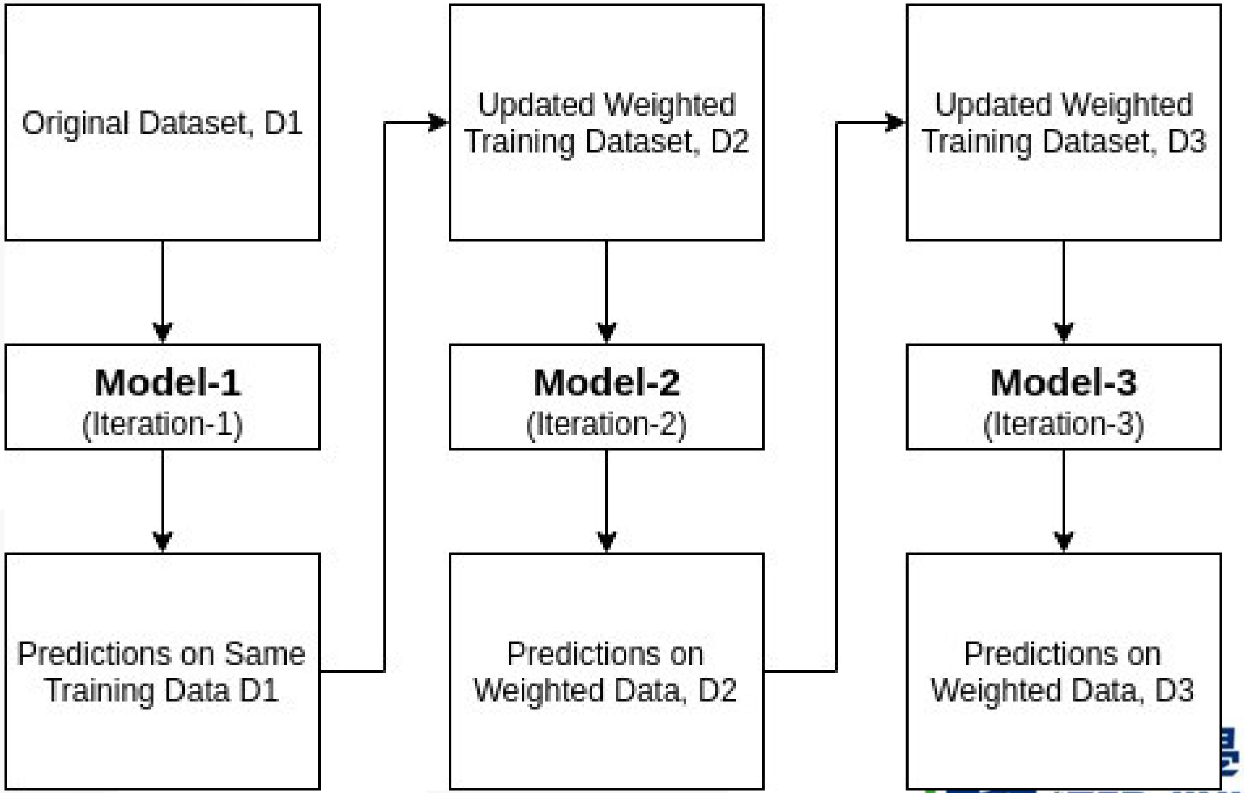
AdaBoost (Adaptive Boosting) Algorithm

- 前一個基本分類器分錯的樣本會得到加強，加權後的全體樣本再次被用來訓練下一個基本分類器。
- 同時，在 每一輪中加入一個新的弱分類器，直到達到某個預定的足夠小的錯誤率或達到預先指定的最大反覆運算次數



AdaBoost (Adaptive Boosting) Algorithm

ITTRAINING

Input :

- A training set $S = ((\mathbf{x}_1, y_1), ..., (\mathbf{x}_m, y_m))$.

Initialization :

- Maximum number of iterations T;   T: number of weak learners
- initialize the weight distribution $\forall i \in \{1, \dots, m\}, D^{(1)}(i) = \frac{1}{m}$.

for $t = 1, \dots, T$ **do**

- Learn a classifier $f_t : \mathbb{R}^d \rightarrow \{-1, +1\}$ using distribution $D^{(t)}$
- Set $\epsilon_t = \sum_{i: f_t(\mathbf{x}_i) \neq y_i} D^{(t)}(i)$ $\boxed{\varepsilon_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i].}$
- Choose $a_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$ make misclassified cases to be updated with increased weights after an iteration.
- Update the weight distribution over examples

Only with low weighted error → $\forall i \in \{1, \dots, m\}, D^{(t+1)}(i) = \frac{D^{(t)}(i) e^{-a_t y_i f_t(\mathbf{x}_i)}}{Z^{(t)}}$

where $Z^{(t)} = \sum_{i=1}^{m} D^{(t)}(i) e^{-a_t y_i f_t(\mathbf{x}_i)}$ is a normalization factor such that $D^{(t+1)}$ remains a distribution.

Output :   The voted classifier $\forall \mathbf{x}, F(\mathbf{x}) = \text{sign}\left(\sum_{t=1}^{T} a_t f_t(\mathbf{x})\right)$

IT學院 TRAINING

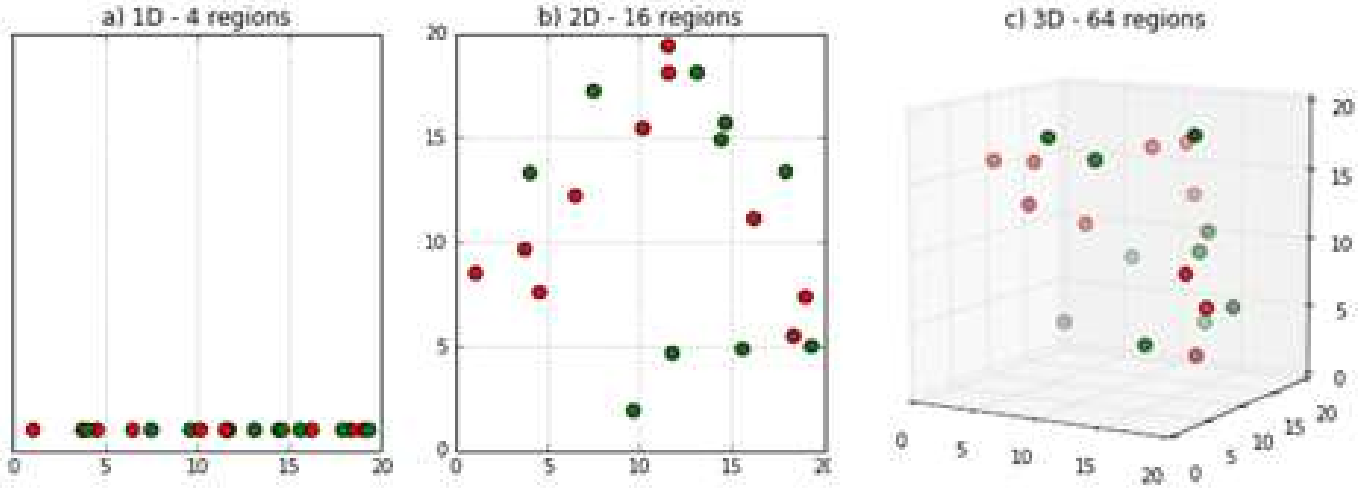
The Ideal Set of Features

1. **High Variance:** features with a lot of variance contain a lot of potential signal
2. **Uncorrelated:** features that are highly correlated with each other are less useful and in certain cases downright harmful (when the correlation is so high as to cause multicollinearity).
3. **Not That Many:** a low number of features relative to our number of target variable observations

PCA: Dimension Reduction

- 主成分分析(Principal component analysis，PCA)，是一種特徵提取的技術。假如訓練資料集裡有很多個特徵數據，我們可以利用主成分分析(Principal component analysis，PCA)來選取最有影響力的幾個特徵來做分類器模型的訓練而不需要使用所有的特徵。
- 因為用了大量特徵，可能會
  1. 發生overfitting
  2. 分類或分群發生困難
     - 如在分群時，在高維度時在計算 Euclidean distance 時, 會發現每筆資料會都是相近的

curse of dimensionality



特徵在高維度時, 資料反而變得稀疏 ➔ 特徵增加你其實要增加更多資料

利用PCA降維也可以避免發生維度災難(curse of dimensionality)的問題

	x1	x2
a	1	0
b	1	0
c	1	0
d	0	1
e	0	1

	y1	y2	y3	y4	y5
a	1	0	0	0	0
b	0	1	0	0	0
c	0	0	1	0	0
d	0	0	0	1	0
e	0	0	0	0	1