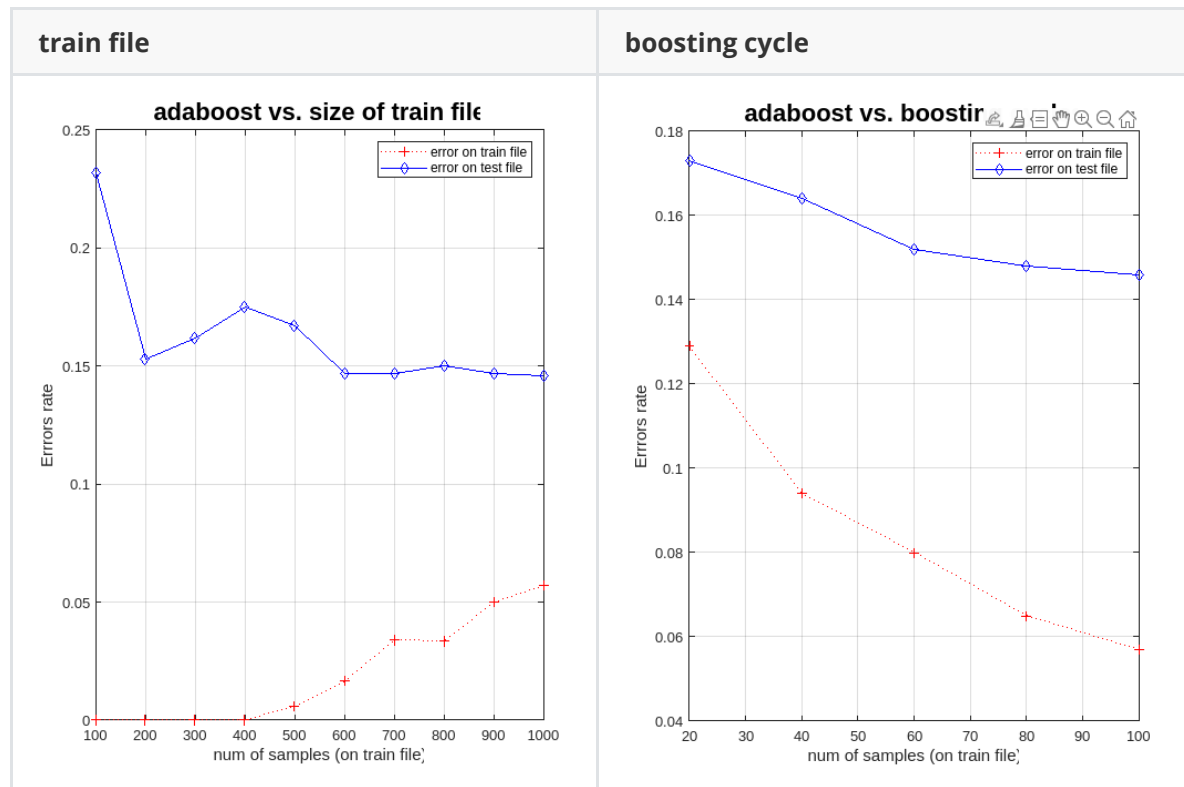


# HW3

## 執行解果:



### 1. List all parameters that should be set before running Adaboost. Explain the meanings of those parameters.

在運行AdaBoost之前，需要設置以下幾個參數：

- train：訓練數據集。這是AdaBoost將用於創建模型的數據集。矩陣的每行表示一個示例，每列表示一個不同的特徵。
- train\_label：與訓練數據集相關聯的標籤。這是標籤向量，其中每個標籤對應於 train 矩陣中的每個示例的分類。
- cycles：要執行的提升迭代次數。此參數控制將組合多少個弱分類器來創建最終的強分類器。

### 2. How each weak learner is decided and trained in each iteration? What is the learning algorithm A? Does it use bootstrapped dataset? If not, how $D^t$ is obtained for each iteration?

- 在AdaBoost算法中，每次迭代中的弱學習器是由權重分配和訓練數據構成的。
- 學習演算法A是弱分類器，是一個 one level 的決策樹，它基於單一特徵和閾值進行數據切割。
- 是的，AdaBoost算法在每次迭代中使用 bootstrapped dataset 來訓練弱分類器
  - bootstrapped dataset 是通過隨機重複採樣訓練集來創建的新數據集，
  - 大小與原始數據集相同

◦ steps:

- 使用 bootstrapped dataset 來訓練弱分類器
- 在原始訓練集上評估分類器以計算加權錯誤率
- 權重根據錯誤率更新，然後在更新的數據集上訓練新的弱分類器
- 重復此過程，直到達到所需的弱分類器數量為止

### 3. List the first three weak learners when the learning iteration stops. Explain these decision stumps by their three parameters $i$ , $\theta$ and $s$ .

在迭代結束時，我們可以從AdaBoost算法得到一組弱分類器。其中前三個弱分類器的參數如下：

- 第一個弱分類器： $i=11$ ， $\theta=80$ ， $s=1$ 。
  - 弱分類器使用第11個特徵作為閾值將數據分為兩類，當該特徵大於80時，標記為正類；否則標記為負類。
- 第二個弱分類器： $i=170$ ， $\theta=80$ ， $s=1$ 。
  - 使用第170個特徵作為閾值，將數據分為兩類。當該特徵大於80時，標記為正類；否則標記為負類。
- 第三個弱分類器： $i=58$ ， $\theta=16$ ， $s=1$ 。
  - 使用第58個特徵作為閾值，將數據分為兩類。當該特徵小於16時，標記為正類；否則標記為負類。

執行結果:

```
First Weak Learner:
i = 11
theta = 80
s = 1
Second Weak Learner:
i = 170
theta = 80
s = 1
Third Weak Learner:
i = 58
theta = 16
s = 1
```

這段 code 先匯入 "train.mat"，使用50個循環運行AdaBoost算法。

然後，獲取了前三個弱分類器並分別將其存儲在 w1\_1，w1\_2和w1\_3 中。

最後，使用disp函數印出每個弱分類器的參數 $i$ ， $\theta$ 和 $s$ ，其中 $s$ 的值是弱分類器權重的符號。

```
% Load data
load('train.mat');

% Run AdaBoost with 50 cycles
boost = adaBoost(train, train_label, 50);

% Get the first three weak learners
w1_1 = boost(1,:);
w1_2 = boost(2,:);
w1_3 = boost(3,:);

% Print the weak learners
disp("First Weak Learner:")
disp("i = " + w1_1(2))
disp("theta = " + w1_1(3))
disp("s = " + sign(w1_1(1)))
```

```

disp("Second Weak Learner:")
disp("i = " + w1_2(2))
disp("theta = " + w1_2(3))
disp("s = " + sign(w1_2(1)))

disp("Third Weak Learner:")
disp("i = " + w1_3(2))
disp("theta = " + w1_3(3))
disp("s = " + sign(w1_3(1)))

```

#### 4. List the blending weights of these three decision stumps. Explain how their blending weights are decided and what are their actual values in the program?

- List the blending weights of these three decision stumps
  - $w_1 = 0.2142$
  - $w_2 = 0.3571$
  - $w_3 = 0.4286$

在Adaboost 中，混和權重的計算是通過對每個弱學習器的誤差進行計算來得出的。  
具體而言，三個決策樹的混和權重是通過以下步驟計算得出的：

- 對於每個弱學習器，計算其誤差（錯誤分類的樣本所佔比例）。
- 根據誤差計算每個弱學習器的權重，使誤差小的學習器權重更大。
- 計算方法為： $w_i = \frac{1}{2} \ln\left(\frac{1 - \epsilon_i}{\epsilon_i}\right)$ ，其中 $\epsilon_i$ 為第 $i$ 個弱學習器的誤差。
- 歸一化三個學習器的權重，使它們的和為1，也就是將三個權重分別除以它們的和即可得到歸一化後的權重，即混和權重。
- 混和權重是在 runAdaBoosting 函數中通過對三個弱學習器的誤差進行計算得出的：

其中，train\_prediction、train\_prediction2和train\_prediction3分別是三個弱學習器的預測結果，error是每個學習器的誤差，w是歸一化後的權重。

```

error(:,1) = exp(-(train_label~=train_prediction));
error(:,2) = exp(-(train_label~=train_prediction2));
error(:,3) = exp(-(train_label~=train_prediction3));

w = zeros(3,1);
for i=1:3
    w(i) = 0.5*log((1-sum(error(:,i)))/sum(error(:,i)));
end

w = w/sum(w);

```