A Formal View of Boosting

- given training set $(x_1, y_1), \ldots, (x_m, y_m)$
- $y_i \in \{-1, +1\}$ correct label of instance $x_i \in X$
- for $t = 1, \ldots, T$:
  - construct distribution D_t on $\{1, \ldots, m\}$
  - find weak hypothesis ("rule of thumb")

    $$h_t : X \to \{-1, +1\}$$

    with small error ϵ_t on D_t:

    $$\epsilon_t = \Pr_{D_t}[h_t(x_i) \neq y_i]$$
- output final hypothesis H_{final}

AdaBoost

[Freund & Schapire]

- constructing D_t:
  - $D_1(i) = 1/m$
  - given D_t and h_t:

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \cdot \begin{cases} e^{-\alpha_t} & \text{if } y_i = h_t(x_i) \\ e^{\alpha_t} & \text{if } y_i \neq h_t(x_i) \end{cases} > 1$$

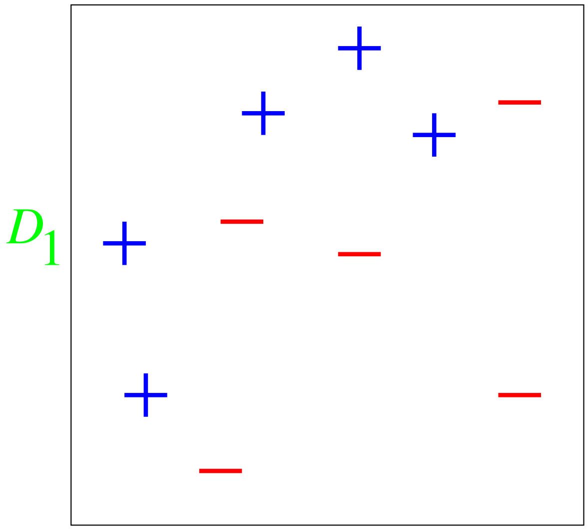
$$= \frac{D_t(i)}{Z_t} \cdot \exp(-\alpha_t \, y_i \, h_t(x_i))$$

where Z_t = normalization constant

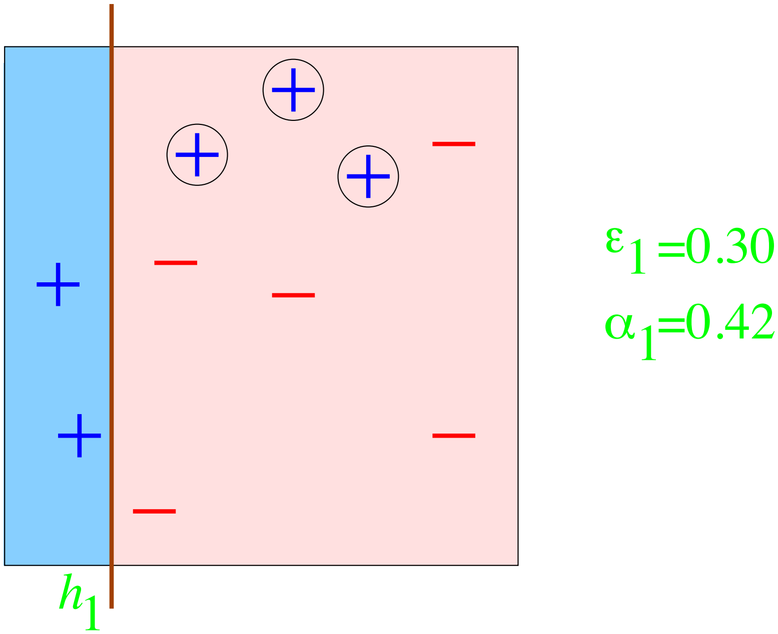
$$\alpha_t = \tfrac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right) > 0$$

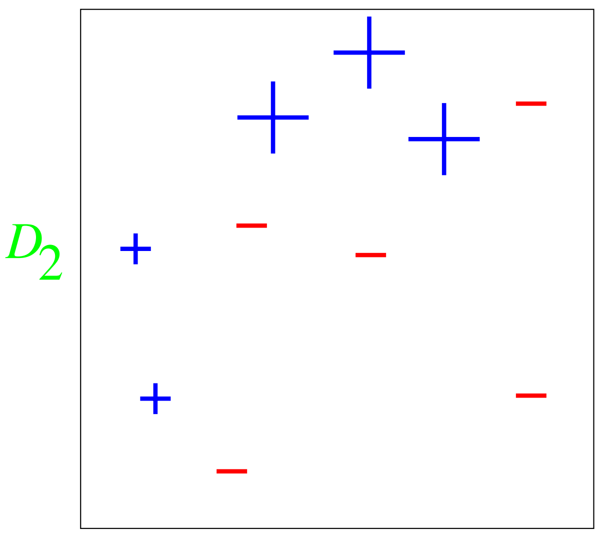
- final hypothesis:
  - $H_{\text{final}}(x) = \text{sign}\left(\sum_t \alpha_t h_t(x)\right)$

Toy Example

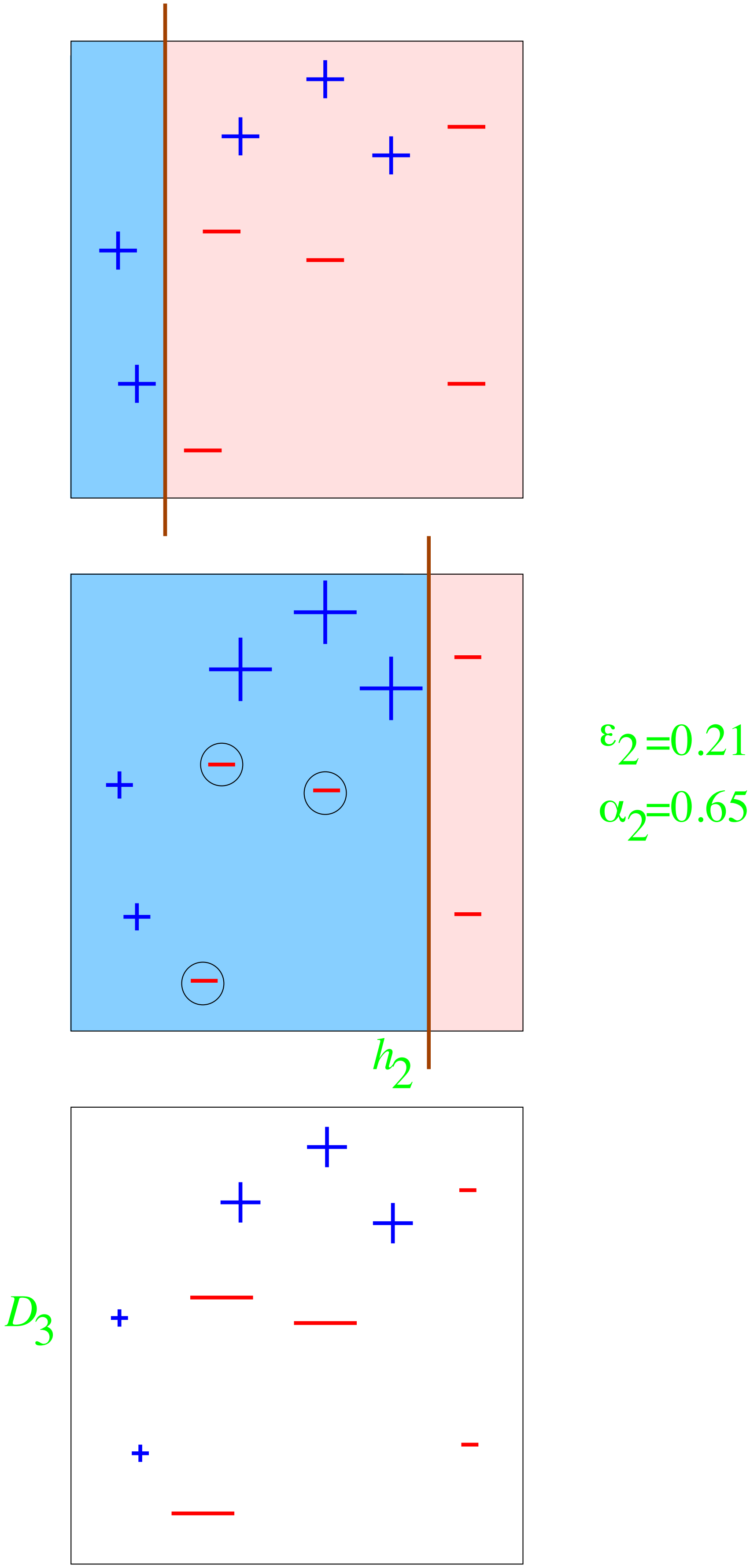


Round 1

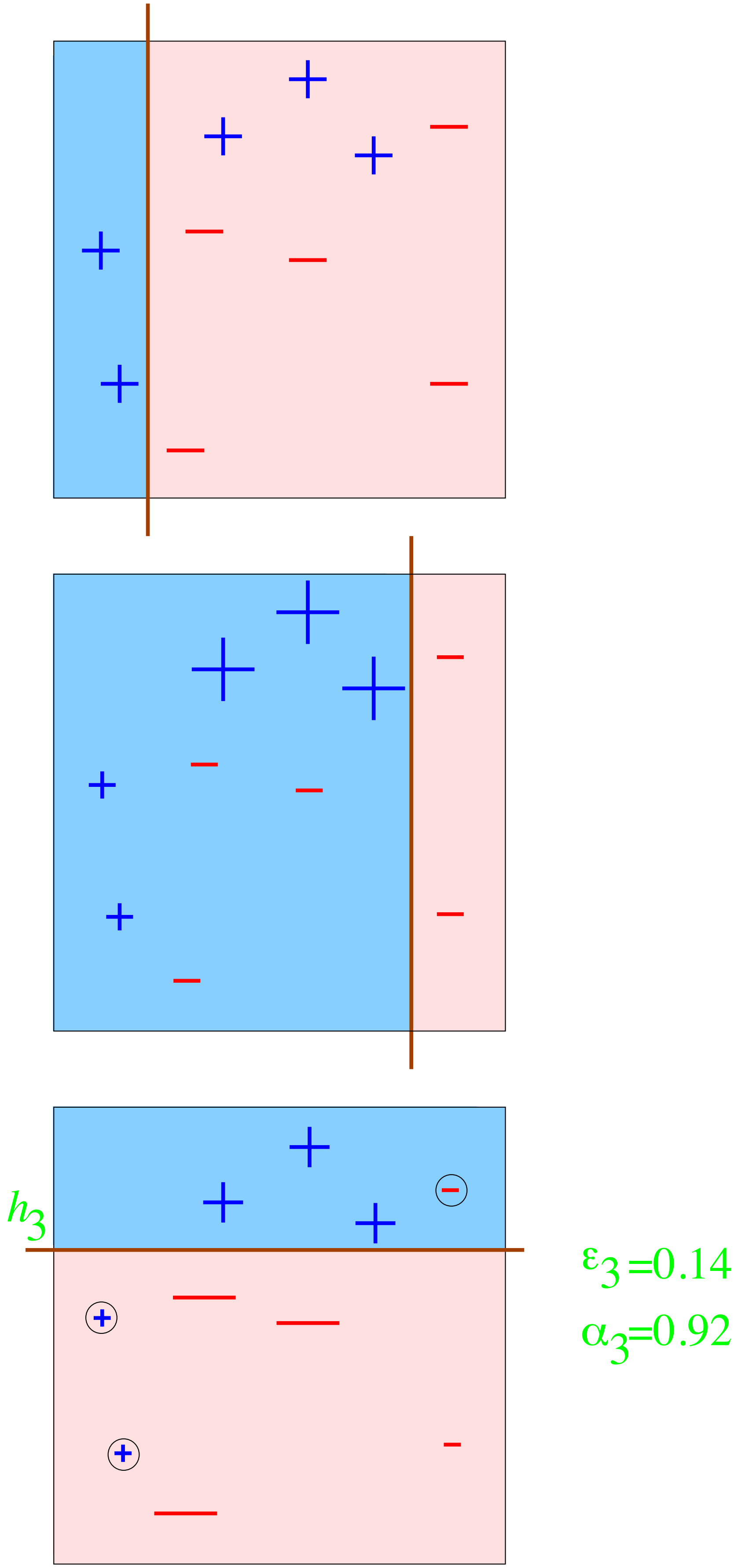




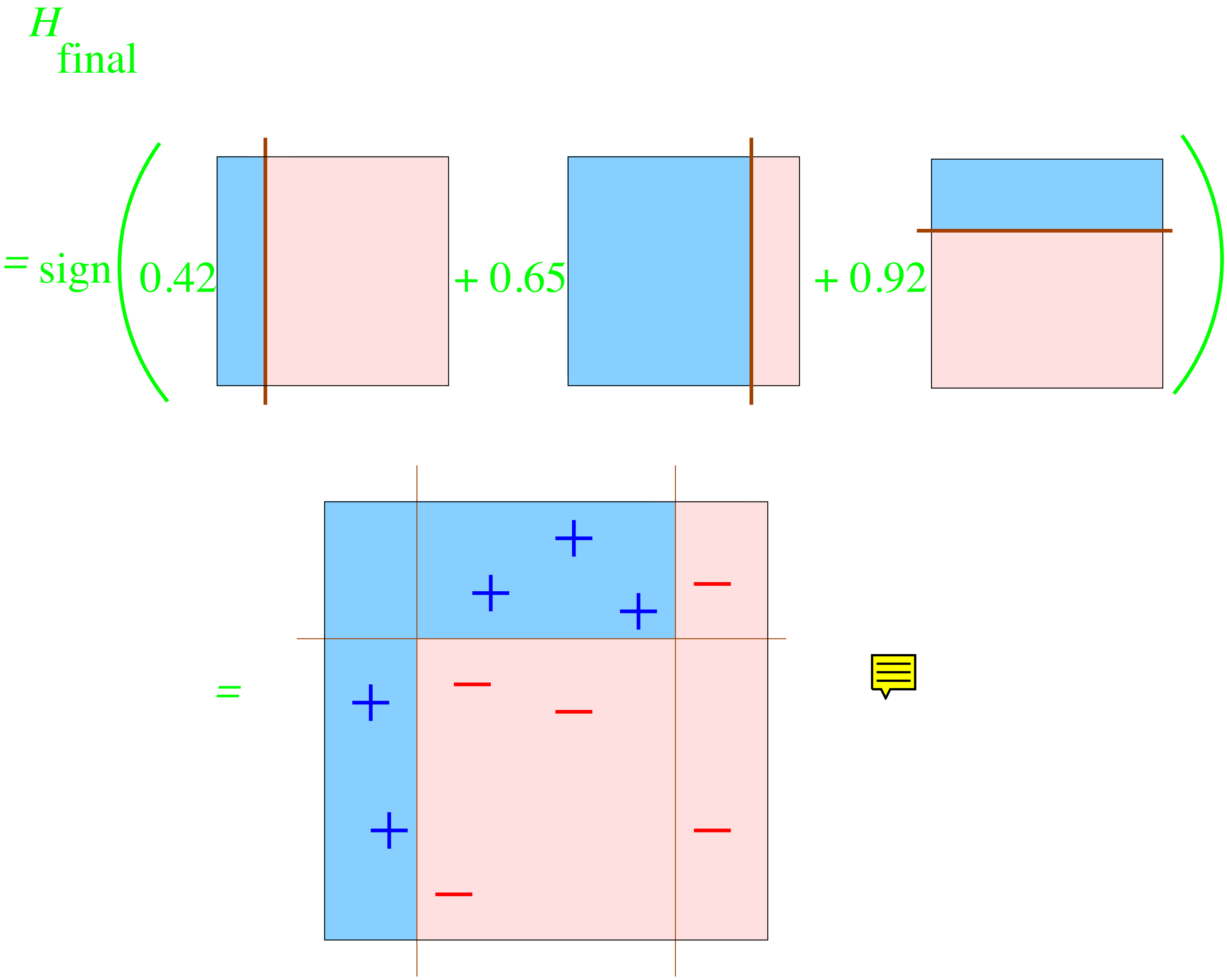
Round 2



Round 3



Final Hypothesis



* See demo at
`www.research.att.com/~yoav/adaboost`

Analyzing the training error

- Theorem:
  - run AdaBoost
  - let $\epsilon_t = 1/2 - \gamma_t$
  - then

$$\begin{aligned}\text{training error}(H_{\text{final}}) &\le \prod_t \left[2\sqrt{\epsilon_t(1-\epsilon_t)}\right] \\ &= \prod_t \sqrt{1-4\gamma_t^2} \\ &\le \exp\left(-2\sum_t \gamma_t^2\right)\end{aligned}$$

- so: if $\forall t: \ \gamma_t \ge \gamma > 0$

  then training error$(H_{\text{final}}) \le e^{-2\gamma^2 T}$

- adaptive:
  - does not need to know γ or T a priori
  - can exploit $\gamma_t \gg \gamma$

Proof

- let $f(x) = \sum_t \alpha_t h_t(x) \Rightarrow H_{\text{final}}(x) = \text{sign}(f(x))$
- *Step 1*: unwrapping recursion:

$$D_{\text{final}}(i) = \frac{1}{m} \cdot \frac{\exp\left(-y_i \sum_t \alpha_t h_t(x_i)\right)}{\prod_t Z_t}$$

$$= \frac{1}{m} \cdot \frac{e^{-y_i f(x_i)}}{\prod_t Z_t}$$

- *Step 2*: training error$(H_{\text{final}}) \leq \prod_t Z_t$
- Proof:
  - $H_{\text{final}}(x) \neq y \Rightarrow y f(x) \leq 0 \Rightarrow e^{-y f(x)} \geq 1$
  - so:

$$\text{training error}(H_{\text{final}}) = \frac{1}{m} \sum_i \begin{cases} 1 & \text{if } y_i \neq H_{\text{final}}(x_i) \\ 0 & \text{else} \end{cases}$$

$$\leq \frac{1}{m} \sum_i e^{-y_i f(x_i)}$$

$$= \sum_i D_{\text{final}}(i) \prod_t Z_t$$

$$= \prod_t Z_t$$

Proof (cont.)

- ***Step 3***: $Z_t = 2\sqrt{\epsilon_t(1-\epsilon_t)}$
- Proof:

$$
\begin{aligned}
Z_t &= \sum_i D_t(i)\exp(-\alpha_t\, y_i\, h_t(x_i)) \\
&= \sum_{i:y_i \neq h_t(x_i)} D_t(i)e^{\alpha_t} + \sum_{i:y_i = h_t(x_i)} D_t(i)e^{-\alpha_t} \\
&= \epsilon_t\, e^{\alpha_t} + (1-\epsilon_t)\, e^{-\alpha_t} \\
&= 2\sqrt{\epsilon_t(1-\epsilon_t)}
\end{aligned}
$$

UCI Experiments

[Freund & Schapire]

- tested AdaBoost on UCI benchmarks
- used:
  - C4.5 (Quinlan's decision tree algorithm)
  - "decision stumps": very simple rules of thumb that test on single attributes