

Sistemas Inteligentes

Boosting y Adaboost

Sistemas Inteligentes

Departament de Ciència de la Computació i Intel·ligència Artificial
 Departament de Ciència de la Computació i Intel·ligència Artificial
 Universitat d'Alacant
 Universidad de Alicante

Tema 10. Boosting y AdaBoost

1

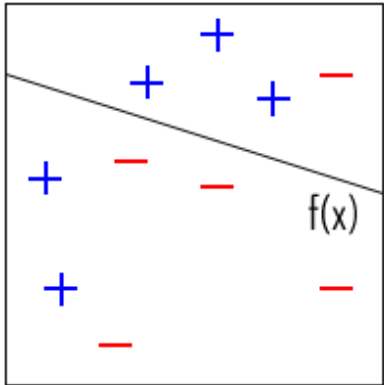
Sistemas Inteligentes

Un poco de notación

Clasificadores supervisados:

1. **Patrones:** $\mathbf{x}_i \in X = R^d$
2. **Clases:** $y_i \in Y = \{-1, 1\}$
3. **Conjunto de entrenamiento:** $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$
4. **Función aprendida:** $f : R^d \rightarrow Y$

$$h_f(x) = \text{sign}(f(x))$$
5. **Clasificador:**



Departament de Ciència de la Computació i Intel·ligència Artificial
 Departament de Ciència de la Computació i Intel·ligència Artificial
 Universitat d'Alacant
 Universidad de Alicante

Tema 10. Boosting y AdaBoost

2

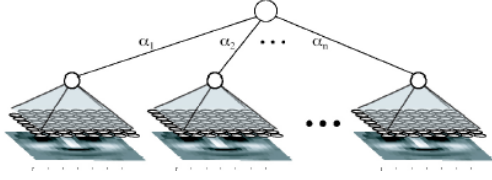
Dept. de Ciència de la Computació i Intel·ligència Artificial
 Dpto. de Ciencia de la Computación e Inteligencia Artificial
 Universitat d'Alacant
 Universidad de Alicante

Sistemas Inteligentes

Combinar clasificadores “débiles”

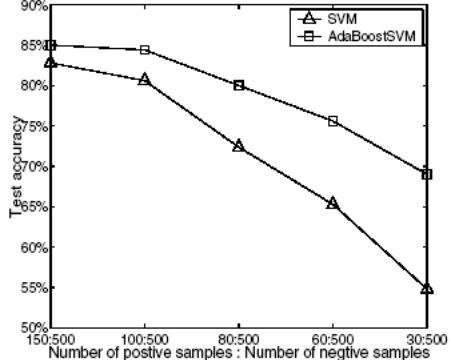
Clasificadores débiles:

Moderadamente precisos (simples y que funcionen al menos mejor que una clasificación aleatoria). El resultado es una hipótesis conjunta (ensemble hypothesis)



Combinación:

- Se puede probar que es posible encontrar un clasificador más preciso combinando muchos clasificadores “débiles”.
- ¿Cómo combinarlos?
 - Bagging**
 - Boosting**



Tema 10. Boosting y AdaBoost

Dept. de Ciència de la Computació i Intel·ligència Artificial
 Dpto. de Ciencia de la Computación e Inteligencia Artificial
 Universitat d'Alacant
 Universidad de Alicante

Sistemas Inteligentes

Bagging. Bootstrap aggregating

Bagging.[Breiman,94] Repeat for $t = 1, \dots, T$:

- Select, at random *with replacement*, N training examples.
- Train learner on selected samples to generate h_t

Final hypothesis is simple vote:

$$H(x) = MAJ(h_1(x), \dots, h_T(x))$$

□ Características:

Ayuda a mejorar clasificadores inestables, como redes neuronales o árboles de decisión (pequeños cambios en el conjunto de entrenamiento llevan a diferentes clasificadores y grandes cambios en el porcentaje de aciertos).

Tema 10. Boosting y AdaBoost

Boosting vs Bagging

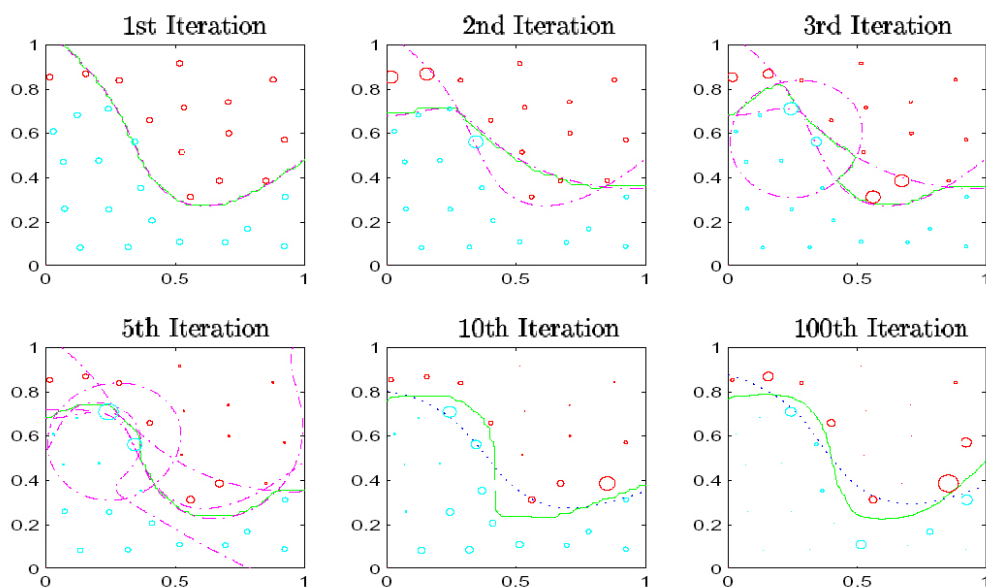
Muestreo ponderado (ejemplos):

- En lugar de hacer un muestreo aleatorio de los datos de entrenamiento, se ponderan las muestras para concentrar el aprendizaje en los **ejemplos más difíciles**.
- Intuitivamente, **los ejemplos más cercanos a la frontera de decisión** son más difíciles de clasificar, y recibirán pesos más altos.

Votos ponderados (clasificadores):

- En lugar de combinar los clasificadores con el mismo peso en el voto, se usa un **voto ponderado**.
- Esta es la **regla de combinación** para el conjunto de clasificadores débiles.
- En conjunción con la estrategia de muestreo anterior, esto produce un **clasificador más fuerte**.

Boosting: explicación intuitiva



Adaboost

AdaBoost. Adaptive Boosting [Freund, Schapire, 96]

- Initialize distribution over training set $D_1(i) = 1/N$.
- For $t = 1, \dots, T$
 1. Train weak learner using distribution D_t and obtain h_t .
 2. Choose a weight (confidence value) $\alpha_t \in \mathbb{R}$.
 3. Update distribution over training set:

$$D_{t+1}(i) = \frac{D_t(i)e^{-\alpha_t y_i h_t(x_i)}}{Z_t}$$

- Set $H(x) = \text{sign}(f(x)) = \text{sign}\left(\sum_{i=1}^T \alpha_t h_t(x)\right)$

Notación:

1. **i** indexa ejemplos, mientras que **t** indexa clasificadores (débiles)
2. **D_t** depende de la complejidad de los ejemplos. ¿Cómo usarla?
3. **a_t** depende del error **e_t** asociado a la **h_t**
4. **Z_t** es una constante de normalización.

Tema 10. Boosting y AdaBoost

7

Adaboost

Algorithm 1 Adaboost

```

1: procedure ADABOOST( $X, Y$ )
2:    $D_1(i) = 1/N$            ▷ Indica como de difícil es de clasificar cada punto  $i$ 
3:   for  $t = 1 \rightarrow T$  do       ▷  $T$  es el número de clasificadores débiles a usar
4:     Entrenar  $h_t$  teniendo en cuenta  $D_t$ 
5:     Start
6:       for  $k = 1 \rightarrow A$  do     ▷  $A$  = num. de pruebas aleatorias
7:          $F_p = \text{generaPlanoAlAzar}()$ 
8:          $\epsilon_t = P_{D_t}(h_t(x_i) \neq y_i) \rightarrow \epsilon_{t,k} = \sum_{i=1}^N D_t(i) \cdot (F_k(x) \neq y(x))$ 
9:         return  $\langle F_p | \min(\epsilon_{t,k}) \rangle$ 
10:      End
11:      Del  $h_t$  anterior obtener su valor de confianza  $\alpha_t \in \mathbb{R}$ 
12:      Start
13:         $\alpha_t = 1/2 \log_2 \left( \frac{1-\epsilon_t}{\epsilon_t} \right)$ 
14:      End
15:      Actualizar  $D_{t+1}$ 
16:      Start
17:         $D_{t+1} = \frac{D_t(i) \cdot e^{-\alpha_t + y_i h_t(x_i)}}{Z_t}$ 
18:      End
19:       $Z_t = \sum_i D_t(i)$ 
20:      return  $H(x) = \text{sign}(\sum_t \alpha_t \cdot h_t(x))$ 

```

Tema 10. Boosting y AdaBoost

8

Sistemas Inteligentes

Construyendo y usando D_t

- Entrenar un clasificador débil usando D_t y obtener h_t
 - Normalmente se muestrean los ejemplos de entrenamiento usando D_t (muestreo por importancia)
 - Inicialmente, cuando $T=1$ todos los ejemplos son igualmente probables.
 - En las siguientes iteraciones, es más probable seleccionar los ejemplos más difíciles (los que hacen fallar al clasificador).
- Escoger un valor de confianza α_t
 - Sea ϵ_t el error asociado a h_t

$$\epsilon_t = Pr_{D_t}[h_t(\mathbf{x}_i) \neq y_i]$$
 - El valor de α_t surge de intentar optimizar dicho error y es:

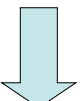
$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right)$$

Tema 10. Boosting y AdaBoost

Sistemas Inteligentes

Construyendo y usando D_t

- Actualizar la distribución D :
 - Inicialmente, cuando $T=1$ todos los ejemplos son igualmente probables.
 - En las siguientes iteraciones, es más probable seleccionar los ejemplos más difíciles (los que hacen fallar al clasificador).
$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \cdot A$$



$$\begin{aligned}
 \text{if } h_t(\mathbf{x}_i) = y_i &\implies A = e^{-\alpha_t} \\
 \text{if } h_t(\mathbf{x}_i) \neq y_i &\implies A = e^{\alpha_t}
 \end{aligned}$$

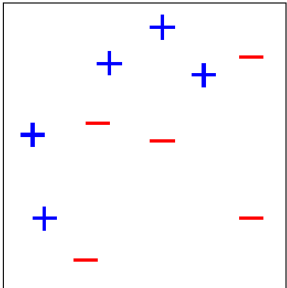
Tema 10. Boosting y AdaBoost

Sistemas Inteligentes

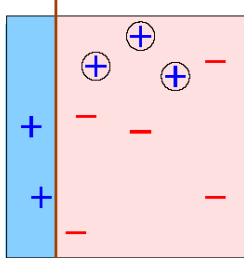
Boosting: ejemplo

Round 1

D_1

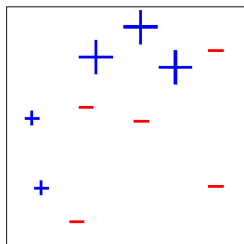


h_1



$\epsilon_1=0.30$
 $\alpha_1=0.42$

D_2



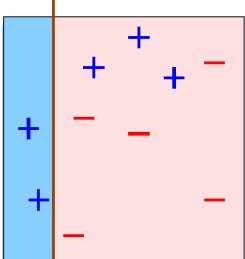
Tema 10. Boosting y AdaBoost

11

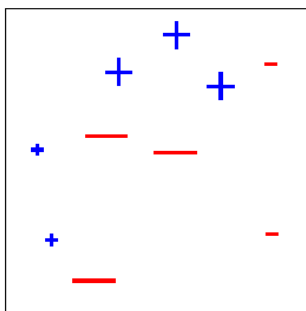
Sistemas Inteligentes

Boosting: ejemplo

Round 2



h_2



$\epsilon_2=0.21$
 $\alpha_2=0.65$

Tema 10. Boosting y AdaBoost

12

Sistemas Inteligentes

Boosting: ejemplo

Round 3

$\epsilon_3 = 0.14$
 $\alpha_3 = 0.92$

Tema 10. Boosting y AdaBoost

Sistemas Inteligentes

Boosting: ejemplo

Final Hypothesis

H_{final}

$= \text{sign} \left(0.42 \begin{array}{|c|} \hline \text{blue} \\ \hline \end{array} + 0.65 \begin{array}{|c|} \hline \text{blue} \\ \hline \end{array} + 0.92 \begin{array}{|c|} \hline \text{blue} \\ \hline \end{array} \right)$

Tema 10. Boosting y AdaBoost

Bibliografía

- Duda, Hart & Stork. [Pattern Classification](#). Wiley 2001. Chapter 9.
- Hastie, Tibshirani, Friedman, [The Elements of Machine Learning](#). Springer Series in Statistics. 2001. Chapter 10.
- www.boosting.org