

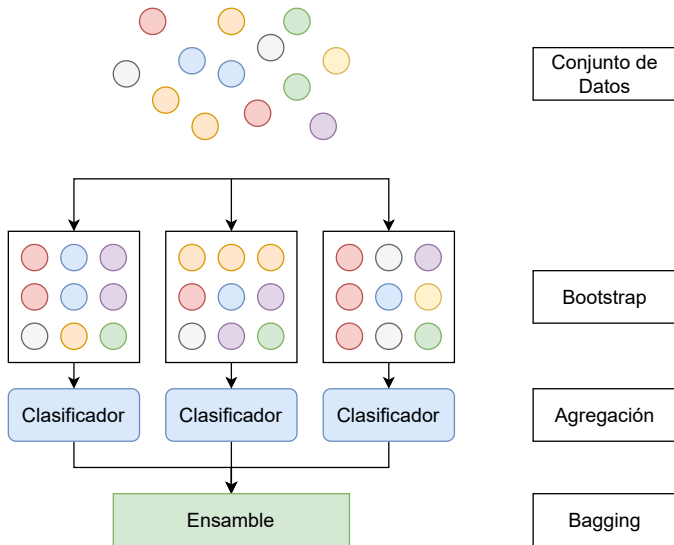
# Ensemble Learning

## Boosting y AdaBoost

Luis Norberto Zúñiga Morales

22 de febrero de 2022

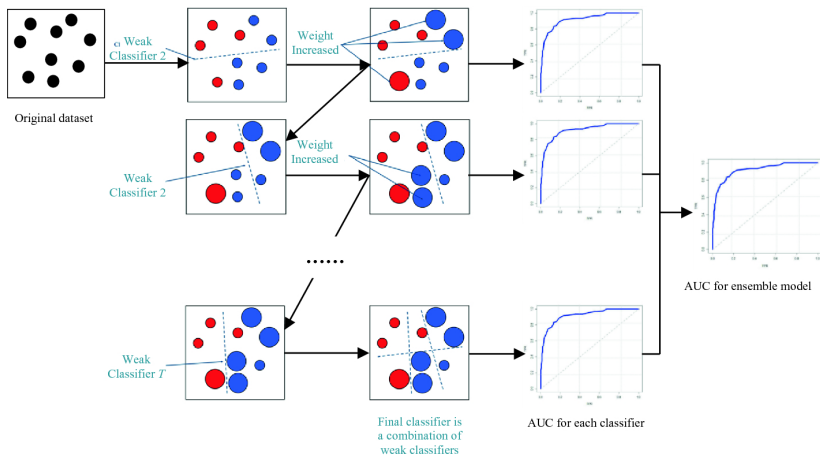
# En clases anteriores...



### Pregunta

¿Se les ocurre alguna otra forma de jugar con los elementos de un ensamble para mejorar su rendimiento?

# Boosting



- **Selecciona muestras aleatorias** las cuales se utilizan para ajustar una serie de modelos de aprendizaje homogéneos en forma secuencial.

- **Selecciona muestras aleatorias** las cuales se utilizan para ajustar una serie de modelos de aprendizaje homogéneos en forma secuencial.
- **Produce nuevos clasificadores** capaces de mejorar el ensamble mediante una mejora al rendimiento en la predicción de instancias históricamente mal asignadas.

# Boosting

- **Selecciona muestras aleatorias** las cuales se utilizan para ajustar una serie de modelos de aprendizaje homogéneos en forma secuencial.
- **Produce nuevos clasificadores** capaces de mejorar el ensamble mediante una mejora al rendimiento en la predicción de instancias históricamente mal asignadas.
- Permite **reducir el sesgo** presente en los modelos base, por lo que es conveniente utilizar aquellos que presenten alto sesgo y baja varianza.

Supongamos lo siguiente:

- Buscamos crear un equipo de fútbol americano para competir en una liga.



Supongamos lo siguiente:

- Buscamos crear un equipo de fútbol americano para competir en una liga.
- **Pregunta:** ¿Cómo realizamos el reclutamiento?

Supongamos lo siguiente:

- Buscamos crear un equipo de fútbol americano para competir en una liga.
- **Pregunta:** ¿Cómo realizamos el reclutamiento?
  - 1 Exploración de jugadores (*scouting*).

Supongamos lo siguiente:

- Buscamos crear un equipo de fútbol americano para competir en una liga.
- **Pregunta:** ¿Cómo realizamos el reclutamiento?
  - 1 Exploración de jugadores (*scouting*).
  - 2 Selección de los mejores jugadores(*draft*).

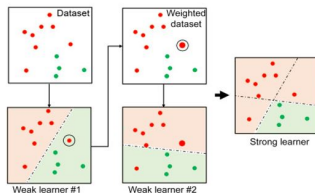
Supongamos lo siguiente:

- Buscamos crear un equipo de fútbol americano para competir en una liga.
- **Pregunta:** ¿Cómo realizamos el reclutamiento?
  - 1 Exploración de jugadores (*scouting*).
  - 2 Selección de los mejores jugadores(*draft*).
  - 3 Asignamos un rol según su importancia.

# AdaBoost

Idea básica [1, 2]:

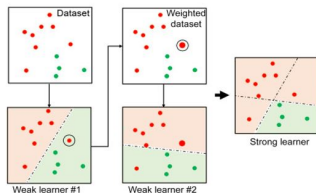
- Se crea un modelo base y se entrena.



# AdaBoost

Idea básica [1, 2]:

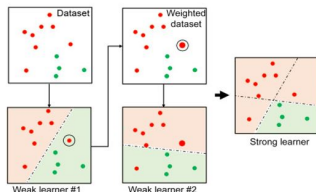
- Se crea un modelo base y se entrena.
- El algoritmo aumenta el peso de los puntos mal clasificados y de los modelos base.



# AdaBoost

Idea básica [1, 2]:

- Se crea un modelo base y se entrena.
- El algoritmo aumenta el peso de los puntos mal clasificados y de los modelos base.
- Se entrena un nuevo modelo base con los pesos actualizados.



Utilizaremos la idea de Rojas [3] para el desarrollo del modelo.

- El primer paso consiste en probar los clasificadores del ensamble por medio de un conjunto de entrenamiento  $T$  de  $N$  puntos  $(\mathbf{x}_i, y_i)$ .



Utilizaremos la idea de Rojas [3] para el desarrollo del modelo.

- El primer paso consiste en probar los clasificadores del ensamble por medio de un conjunto de entrenamiento  $T$  de  $N$  puntos  $(\mathbf{x}_i, y_i)$ .
- A cada modelo se le asigna un costo:
  - $e^{\beta}$  cada vez que el clasificador falla,  $\beta > 0$
  - $e^{-\beta}$  si el clasificador predice la clase correcta,  $\beta > 0$

Utilizaremos la idea de Rojas [3] para el desarrollo del modelo.

- El primer paso consiste en probar los clasificadores del ensamble por medio de un conjunto de entrenamiento  $T$  de  $N$  puntos  $(\mathbf{x}_i, y_i)$ .
- A cada modelo se le asigna un costo:
  - $e^{\beta}$  cada vez que el clasificador falla,  $\beta > 0$
  - $e^{-\beta}$  si el clasificador predice la clase correcta,  $\beta > 0$
- A este tipo de error ( $e^{\beta}$ ) se le conoce como función de pérdida exponencial.

Cuando se prueban los  $k$  clasificadores, se construye una matriz  $S$  que alberga los resultados de aciertos (0) y fallos (1) de cada uno de ellos.

	$\phi_1$	$\phi_2$	...	$\phi_k$
$\mathbf{x}_1$	0	1	...	1
$\mathbf{x}_2$	0	1	..	0
$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\mathbf{x}_k$	1	1	...	0

### Pregunta

¿Cómo se elige el nuevo elemento del ensamble?

### Pregunta

¿Cómo se elige el nuevo elemento del ensamble?

### Respuesta

Con un modelo iterativo:

- En cada paso se agregan funciones al ensamble.
- Se jerarquizan todos los posibles clasificadores para elegir al mejor de ellos.

En la  $m$ -ésima iteración, ya se han incluido  $m - 1$  clasificadores en el ensamble, por lo que en este paso se debe agregar uno más. Hasta el momento, se tiene la siguiente combinación lineal:

$$C_{m-1}(\mathbf{x}_i) = \alpha_1 \phi_1(\mathbf{x}_i) + \dots + \alpha_{m-1} \phi_{m-1}(\mathbf{x}_i) \quad (1)$$

y se desea llegar a

$$C_m(\mathbf{x}_i) = C_{m-1}(\mathbf{x}_i) + \alpha_m \phi_m(\mathbf{x}_i) \quad (2)$$

En la primera iteración,  $C_{m-1} = 0$ .

El error total del clasificador extendido se define como la pérdida exponencial:

$$E = \sum_{i=1}^N \exp[-y_i(C_{m-1}(\mathbf{x}_i) + \alpha_m \phi_m(\mathbf{x}_i))] \quad (3)$$

donde los valores óptimos de  $\alpha_m$  y  $\phi_m$  se encuentran pendientes de ser encontrados.

### Ecuación 3

$$E = \sum_{i=1}^N \exp[-y_i(C_{m-1}(\mathbf{x}_i) + \alpha_m \phi_m(\mathbf{x}_i))]$$

La Ec. (3) se puede expresar de la siguiente forma:

$$E = \sum_{i=1}^N w_i^{(m)} \exp[-y_i \alpha_m \phi_m(\mathbf{x}_i)] \quad (4)$$

donde

$$w_i^{(m)} = \exp[-y_i C_{m-1}(\mathbf{x}_i)] \quad (5)$$

para  $i = 1, \dots, N$ .



### Ecuación 5

$$w_i^{(m)} = \exp[-y_i C_{m-1}(\mathbf{x}_i)]$$

- En la primera iteración,  $w_i^{(1)} = 1$ .
- Posteriormente, para cada nueva iteración, el vector  $w^{(m)}$  representa el peso asignado a cada dato en el conjunto de entrenamiento en la  $m$ -ésima iteración.

### Ecuación 4

$$E = \sum_{i=1}^N w_i^{(m)} \exp[-y_i \alpha_m \phi_m(\mathbf{x}_i)]$$

La suma en la Ec. (4) se puede dividir de la siguiente manera:

$$E = \sum_{y_i = \phi_m(\mathbf{x}_i)} w_i^{(m)} \exp(-\alpha_m) + \sum_{y_i \neq \phi_m(\mathbf{x}_i)} w_i^{(m)} \exp(\alpha_m) \quad (6)$$

lo cual nos indica que el error total es el error ponderado de los aciertos ( $e^{-\beta}$ ) y el error ponderado de los fallos ( $e^{\beta}$ ).

### Ecuación 6

$$E = \sum_{y_i = \phi_m(\mathbf{x}_i)} w_i^{(m)} \exp(-\alpha_m) + \sum_{y_i \neq \phi_m(\mathbf{x}_i)} w_i^{(m)} \exp(\alpha_m)$$

La primera sumatoria se puede representar como  $W_c \exp(-\alpha_m)$  y la segunda como  $W_e \exp(\alpha_m)$ :

$$E = W_c \exp(-\alpha_m) + W_e \exp(\alpha_m) \quad (7)$$

### Ecuación 7

$$E = W_c \exp(-\alpha_m) + W_e \exp(\alpha_m)$$

Para la selección de  $\phi_m$ , el valor exacto de  $\alpha_m > 0$  es irrelevante, ya que para  $\alpha_m$  fija, minimizar  $E$  es equivalente a minimizar  $e^{\alpha_m} E$  y por que

$$\exp(\alpha_m) E = W_c + W_e \exp(2\alpha_m) \quad (8)$$

Dado que  $\exp(2\alpha_m) > 1$ , la Ec. (8) se puede reescribir como

$$\exp(\alpha_m) E = (W_c + W_e) + W_e(\exp(2\alpha_m) - 1) \quad (9)$$

### Ejercicio #1

Verificar que la ecuación

$$\exp(\alpha_m)E = W_c + W_e \exp(2\alpha_m)$$

es igual a

$$\exp(\alpha_m)E = (W_c + W_e) + W_e(\exp(2\alpha_m) - 1)$$

### Ecuación 9

$$\exp(\alpha_m)E = (W_c + W_e) + W_e(\exp(2\alpha_m) - 1)$$

- Noten que  $(W_c + W_e)$  es la suma total de todos los pesos de los datos en el conjunto de datos, una constante en la iteración en turno.
- El segundo término de la Ec. (9) se minimiza cuando en la  $m$ -ésima iteración se elige el clasificador con el menor error total  $W_e$ , i.e., el que tenga la menor proporción del error ponderado.
- En pocas palabras, tiene sentido elegir al que menor error tenga dentro de los candidatos para que entre al ensamble.

Ya que se eligió el  $m$ -ésimo miembro del ensamble, se debe determinar su peso  $\alpha_m$ .

### Ejercicio #2

Determinar

$$\frac{d}{d\alpha_m} W_c \exp(-\alpha_m) + W_e \exp(\alpha_m)$$

donde

$$W_c = \sum_{y_i = \phi_m(\mathbf{x}_i)} \exp[-y_i C_{m-1}(\mathbf{x}_i)]$$

y

$$W_e = \sum_{y_i \neq \phi_m(\mathbf{x}_i)} \exp[-y_i C_{m-1}(\mathbf{x}_i)]$$

### Solución

$$\frac{dE}{d\alpha_m} = -W_c \exp(-\alpha_m) + W_e \exp(\alpha_m)$$



### Ejercicio #3

Igualen a cero y multipliquen por  $\exp(\alpha_m)$ :

$$-W_c \exp(-\alpha_m) + W_e \exp(\alpha_m)$$

### Ejercicio #3

Igualen a cero y multipliquen por  $\exp(\alpha_m)$ :

$$-W_c \exp(-\alpha_m) + W_e \exp(\alpha_m)$$

### Solución

Igualando a cero y multiplicando por  $\exp(\alpha_m)$  se obtiene que

$$-W_c + W_e \exp(2\alpha_m) = 0 \quad (10)$$

por lo que el óptimo se encuentra dado por

$$\alpha_m = \frac{1}{2} \ln \left( \frac{W_c}{W_e} \right) \quad (11)$$

### Ecuación 17

$$\alpha_m = \frac{1}{2} \ln \left( \frac{W_c}{W_e} \right)$$

Recordando que  $W$  es la suma total de los pesos, la Ec. (11) se puede escribir como

$$\alpha_m = \frac{1}{2} \ln \left( \frac{W - W_e}{W_e} \right) = \frac{1}{2} \ln \left( \frac{1 - e_m}{e_m} \right) \quad (12)$$

donde  $e_m = W_e / W$  representa la razón del error dados los pesos de todos los puntos del conjunto de datos.

### Ejercicio #4

Verificar que

$$\alpha_m = \frac{1}{2} \ln \left( \frac{W_c}{W_e} \right)$$

es igual a

$$\alpha_m = \frac{1}{2} \ln \left( \frac{W - W_e}{W_e} \right) = \frac{1}{2} \ln \left( \frac{1 - e_m}{e_m} \right)$$

donde  $e_m = W_e / W$ .

1. De la lista de posibles clasificadores elegir al  $\phi_m$  que minimice

$$W_e = \sum_{y \neq \phi_m(\mathbf{x}_i)} w_i^{(m)}$$

2. Establecer el peso  $\alpha_m$  del clasificador a

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

3. Actualizar los pesos de los datos para la siguiente iteración. Si  $\phi(\mathbf{x}_i)$  es un fallo

$$w_i^{m+1} = w_i^m e^{\alpha_m} = w_i^m \sqrt{\frac{1 - e_m}{e_m}}$$

Si es un acierto

$$w_i^{m+1} = w_i^m e^{-\alpha_m} = w_i^m \sqrt{\frac{e_m}{1 - e_m}}$$

4. Para la selección del conjunto de datos, AdaBoost tiene dos opciones:
- Realizar un muestreo aleatorio considerando los pesos de cada dato, i.e., un muestreo con importancia.
  - Utilizar todo el conjunto de datos original, considerando los pesos como una manera de aumentar o disminuir el error del ensamble.

5. El algoritmo se detiene cuando el número de modelos base es alcanzado, o bien, cuando encuentra un modelo perfecto.
6. Para realizar las predicciones, AdaBoost simplemente computa las predicciones de todos los predictores y las pondera por medios de los pesos  $\alpha_j$ . La clase predicha es aquella que recibe la mayoría de votos ponderados:

$$\hat{y}_i = \psi(\mathbf{x}_i) = \text{sign} \left( \sum_{j=1}^M \alpha_j \phi_j(\mathbf{x}_i) \right)$$

- [1] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. *Machine Learning: Proceedings of the Thirteenth International Conference*, 1996.
- [2] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, aug 1997.
- [3] Raul Rojas. Adaboost and the super bowl of classifiers a tutorial introduction to adaptive boosting. Technical report, Freie Universität Berlin, 2009.