



**U.N. Sede Medellín**

Una universidad con criterio nacional y presencia regional

# CLASIFICADOR DE ENSAMBLE ADABOOST

Descubrimiento de Conocimiento en Bases de Datos

Dilson Ríos

*Especialización Analítica*

Felipe Garcia

*Maestría en Ingeniería*

[minas.medellin.unal.edu.co](http://minas.medellin.unal.edu.co)

Facultad de Minas  
Sede Medellín



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

# Contenido

1. Introducción
2. Métodos de Ensamble
3. Tipos de Combinación
4. Introducción AdaBoost
5. Descripción Matemática
6. Descripción Interna
7. Ejemplo Práctico en Python
8. Conclusiones

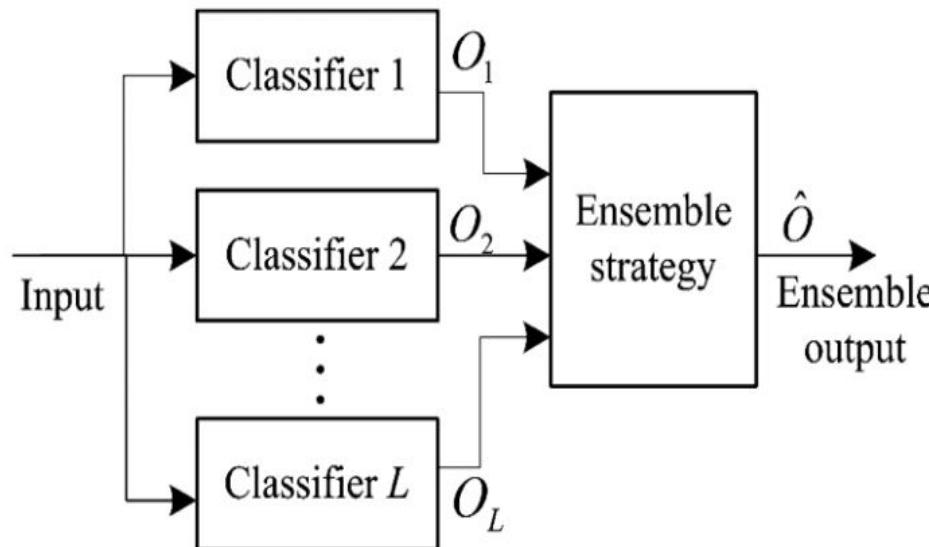
# Introducción

El término boosting hace referencia a un tipo de algoritmos cuya finalidad es encontrar una hipótesis fuerte a partir de utilizar hipótesis simples y débiles.

En este trabajo hablaremos del algoritmo AdaBoost, creado por Freund y Schapire, diseño mejorado del boosting original; y de sus diferentes variantes.



# Métodos de Ensamble



Bagging

**Boosting**

Stacking

# Tipos de Combinación



- Voto mayoritario
- Voto Bayesiano
- Por Ranking
- Combinación lineal
- Por producto
- Pesada
- Stacking
- Seleccionador de regiones/clasificadores

# Historia



AdaBoost es una contracción de “Adaptive Boosting”, en donde el término Adaptive hace alusión a su principal diferencia con su predecesor Baggin.

En términos de funcionalidad son iguales, ambos algoritmos buscan crear un clasificador fuerte cuya base sea la combinación lineal de clasificadores “débiles simples”.

# Descripción Matemática

$$H(x) = \text{sign} \left( \sum_{t=1}^T \alpha_t h_t(x) \right).$$

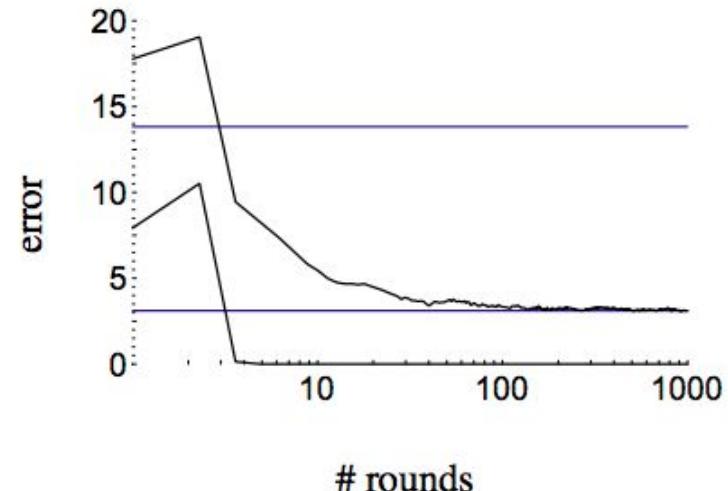
**Figure 1:** The boosting algorithm AdaBoost.

- Conjunto de Clasificadores Débiles.
- Pesos de los Clasificadores.
- Función de toma de decisión

# Optimización del Error

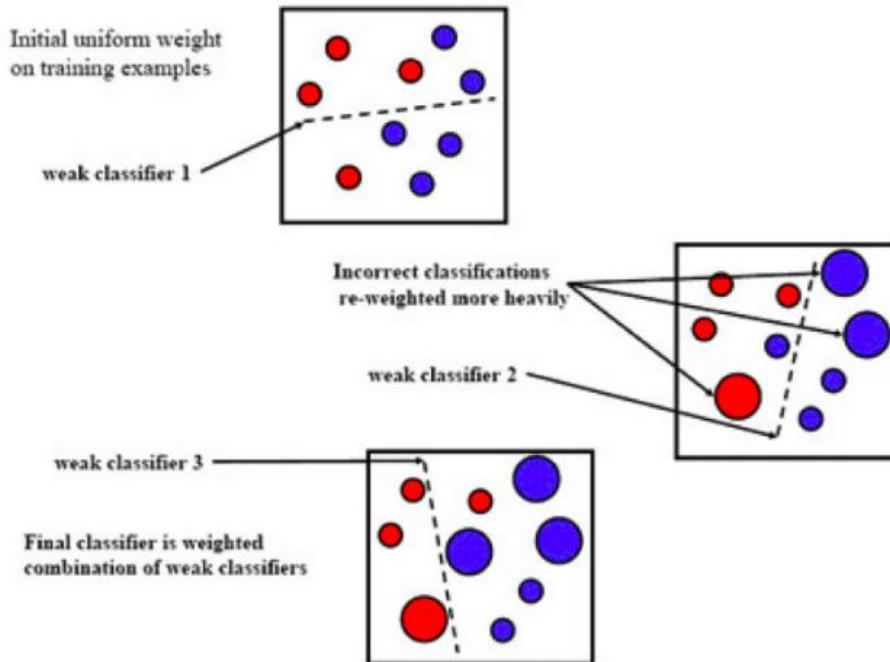
Algoritmo de Descenso de Gradiente para la optimización del error.

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



# Funcionamiento Interno

## Boosting



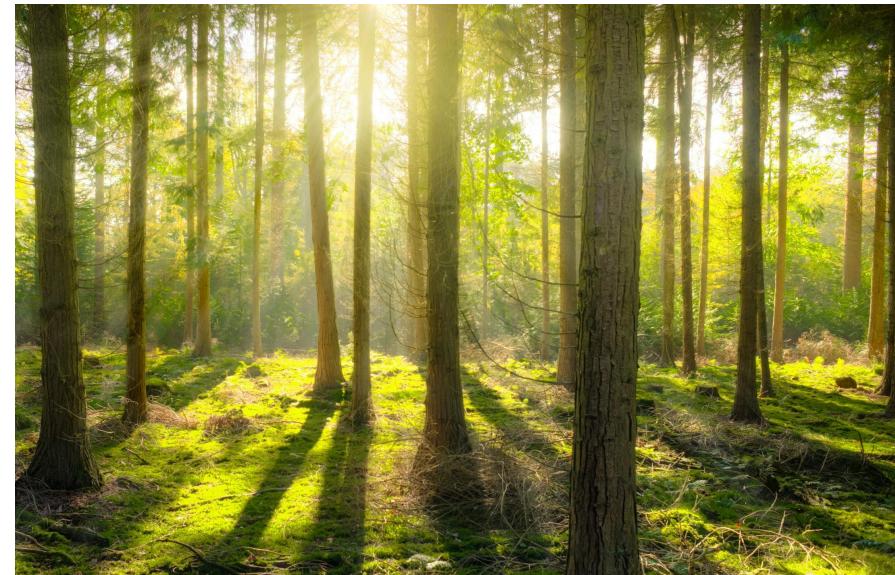
$$H(x) = \text{sign}(\alpha_1 h_1(x) + \alpha_2 h_2(x) + \alpha_3 h_3(x))$$

Ajuste adaptativo de los pesos de los clasificadores sobre las regiones y/o puntos donde se encuentran errores.

# AdaBoost con Árboles de Decisión

Como se ha mencionado AdaBoost es un meta-algoritmo de ensamble que construye un clasificador robusto a partir de clasificadores débiles.

Una de las opciones es utilizar **ÁRBOLES DE DECISIÓN** como clasificadores débiles usando el algoritmo **C4.5**



# Árboles de Decisión con C4.5

Creación de los árboles basados en el concepto de **Entropía de la Información**.

Cada nodo se crea buscando la mejor división de clases posibles.



# Algoritmo C4.5

## Mejoras sobre ID3

- Manejo de datos continuos y discretos.
- Manejo de atributos con valores faltantes.
- Podado del árbol generado.

## Mejoras en el C5.0 vs C4.5

- Mejora en el manejo de memoria.
- Mejora en el manejo de velocidad.
- Arboles de decisión mas pequeños

# Ejemplo Práctico

Clasificación multiclas de un dataset de Vinos utilizando AdaBoost con Árboles de Decisión

## Repositorio GitHub

<https://github.com/FelipeGarcia911/adaboost/blob/master/Ejemplo%20Practico/AdaBoostClassifier.ipynb>



# Ventajas y desventajas AdaBoost

## VENTAJAS

- Sencillo de programar.
- El único parámetro a establecer son las iteraciones.
- El clasificador débil no requiere conocimiento previo.
- Versátil y rápido

## DESVENTAJAS

- Clasificadores débiles complejos pueden llevar a overfitting.
- Clasificadores débiles demasiado débiles pueden producir un bajo margen y overfitting.
- Es vulnerable al ruido.

# Conclusiones

- La unión de varios clasificadores débiles nos permite generar uno más robusto, el cual disminuye los efectos de sesgo y errores al realizar la combinación de dos o más clasificadores.
- AdaBoost puede ser utilizado con diferentes tipos de clasificadores, no solo con Árboles de Decisión.
- Debe analizarse el tipo de clasificador y tipo de problema a afrontar para determinar cuáles son los mejores hiper-parámetros a configurar en el clasificador base usado en AdaBoost.
- Los Árboles de Decisión son clasificadores que han evolucionado del ID3 hasta el C5.0, ofreciendo avances significativos en cuanto a optimización y rendimiento.

# Bibliografía

## References

- [1]Y. Freund and R. Schapire, "A Short Introduction to Boosting", *Machine Learning*, vol. 37, no. 3, pp. 277-296, 1999.
- [2]E. Morales and H. Escalante, "Ensambles de Clasificadores", *Ccc.inaoep.mx*, 2018. [Online]. Available: <https://ccc.inaoep.mx/~emorales/Cursos/Aprendizaje2/Acetatos/ensambles.pdf>. [Accessed: 21- May- 2018].
- [3]B. HSSINA, A. MERBOUHA, H. EZZIKOURI and M. ERRITALI, "A comparative study of decision tree ID3 and C4.5", *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 2, 2014.
- [4]"Free stock photos · Pexels", *Pexels.com*, 2018. [Online]. Available: <https://www.pexels.com/>. [Accessed: 21- May- 2018].
- [5]"jdvelasq/Python-for-predictive-analytics", *GitHub*, 2018. [Online]. Available: <https://github.com/jdvelasq/Python-for-predictive-analytics>. [Accessed: 21- May- 2018].

**Facultad de Minas**  
Sede Medellín



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

- *Dependencias, Unidades básicas académicas o Áreas curriculares  
Secretarías, Facultades, Institutos o Centros*

*Dirección  
Medellín, Colombia  
(+57 4) 430 90 00  
correo@unal.edu.co  
minas.medellin.unal.edu.co*