Machine Learning para Inteligencia Artificial

Ensembles: Bagging y Random Forest

Universidad ORT Uruguay

7 de Mayo, 2025

Índice

Ensembles: combinar modelos

Bagging

Random Forest

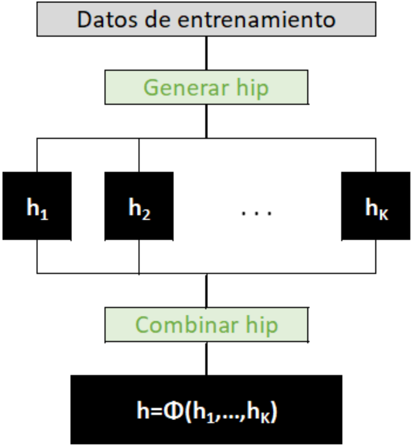
Definición general de ensemble

La técnica de **ensemble** consiste en:

1. Generar varias (K) hipótesis $h_1, \ldots, h_K$
2. Combinarlas en una hipótesis h mediante:

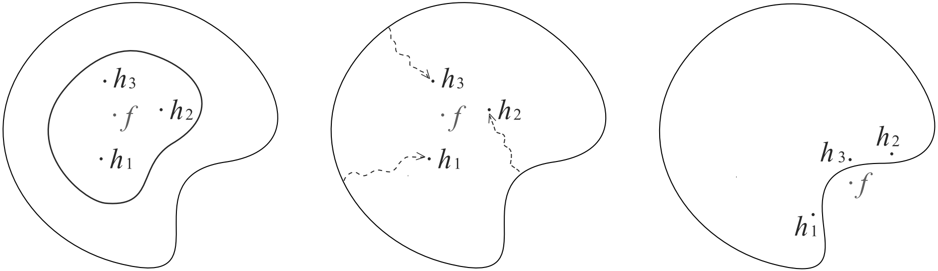
$$h(x) = \boldsymbol{\Phi}(h_1(x), \ldots, h_K(x))$$

La función Φ define cómo combinar las hipótesis y puede ser fija o aprendida durante el entrenamiento.



Tres razones para combinar hipótesis

- Varianza estadística: pequeño cambio en T genera gran cambio en h
- Varianza computacional: óptimo local, algoritmo glotón
- Sesgo inductivo (o de representación)



(a) Statistical (b) Computational (c) Representational

Ensemble: regresión

Promedio

$$h(\boldsymbol{x}) = \frac{1}{K} \sum_{k=1}^{K} h_k(\boldsymbol{x})$$

Suma pesada

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \; h_k(\boldsymbol{x})$$

- Si los h_k son independientes (difícil de conseguir en la práctica):

$$\mathrm{MSE}(h) \leq \frac{1}{K} \sum_{k} \mathrm{MSE}(h_k)$$

- Más aún, si los sesgos de h_k son pequeños $\mathrm{MSE}(h) \ll \frac{1}{K} \sum_k \mathrm{MSE}(h_k)$
- Suma pesada: hay que buscar los pesos $\boldsymbol{w}$ óptimos
- Occam's razor: Promedio suele ser mejor (no over-fitting, comp. simple)

Promedios: Idea intuitiva

- El sesgo del ensemble promedio $h = \frac{1}{K}\sum_k h_k$ satisface

$$\mathrm{B}(h) \leq \frac{1}{K}\sum_{k=1}^{K} \mathrm{B}\left(h_k\right)$$

- Si los $h_1, \ldots, h_K$ son independientes, la varianza satisface

$$\mathrm{V}(h) = \frac{1}{K^2}\sum_{k=1}^{K} \mathrm{V}\left(h_k\right) = \frac{1}{K}\underbrace{\left(\frac{1}{K}\sum_{k=1}^{K} \mathrm{V}\left(h_k\right)\right)}_{\text{promedio de varianzas}}$$

- Entonces la MSE esperada del ensemble promedio satisface

$$\mathrm{MSE}(h) \leq \frac{1}{K}\sum_{k} \mathrm{MSE}(h_k)$$

Ensemble: clasificación

Voto mayoritario de los clasificadores

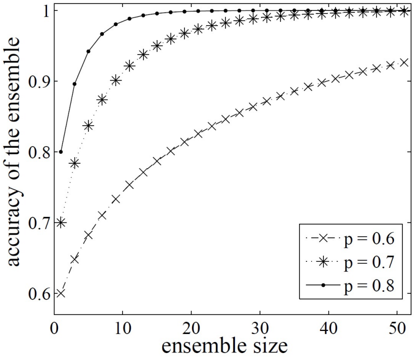
$$h(x) = \arg\max_c \sum_{k=1}^{K} \mathbb{1}(h_k(x) = c)$$

$$\mathbb{1}(u) = \begin{cases} 0 & \text{if } u = \texttt{FALSE} \\ 1 & \text{if } u = \texttt{TRUE} \end{cases}$$

Caso binario:

Si los h_k son independientes y tienen exactitud $p > 0.5$, la exactitud de h tiende a 1 cuando $K \to +\infty$

Pero no es esperable en la práctica



Ensemble: clasificación

Voto pesado de los clasificadores

$$h(x) = \arg \max_c \sum_{k=1}^{K} w_k \ \mathbb{1}(h_k(x) = c)$$

Soft-voting: la predicción de las hipótesis es una probabilidad

$$h_k^c(x) = \widehat{\mathrm{Prob}}_k[Y = c \mid X = x]$$

$$h^c(x) = \sum_{k=1}^{K} w_k \ h_k^c(x)$$

$$h(x) = \arg \max_c h^c(x)$$

Bagging (Bootstrap Aggregation)

Ensemble: Promediar reduce la varianza

¿Cómo hacemos para generar K hipótesis diferentes?

Primera idea: bagging

Para $k \in \{1, \ldots, K\}$ tomar muestra bootstrap de T:

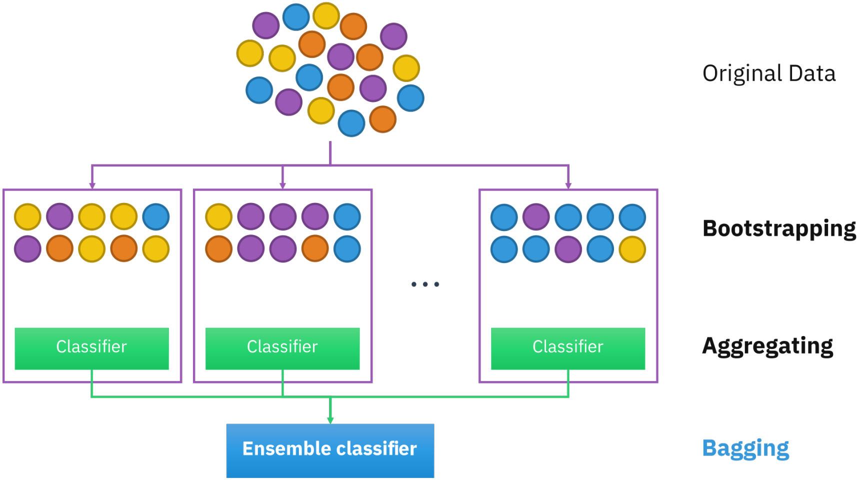
- Construir $B_k \subset S$, tal que[1] $\#B_k = M$, con selección aleatoria con reemplazo

Construir el clasificador (e.g., árbol) h_k usando B_k

Problema: correlación entre los h_k

[1] Se suele usar $\#B_k = \#T$ pero puede ser distinto.

Bagging (Bootstrap Aggregation)



https://en.wikipedia.org/wiki/Bootstrap_aggregating

Bagging (Bootstrap Aggregation)

Algoritmo Bagging

Sea K el número de muestras bootstrap

para $k = 1$ a K repetir
    Tomar muestra bootstrap B_k de tamaño M
    Entrenar h_k en B_k
$h = \Phi(h_1, \ldots, h_K)$

La forma de combinar las hipótesis Φ puede ser:

- Promedio, Suma pesada (regresión)
- Voto mayoritario, Voto pesado, Soft-voting (clasificación)

Bagging (Bootstrap Aggregation)

- Mismo algoritmo - diferentes subconjuntos de datos (muestras bootstrap).
- Mejora el desempeño de algoritmos inestables (alta varianza, sobreajuste).
- Out-of-bag samples (OOB)

	Datos de entrenamiento	Out-of-bag
	1 2 3 4 5 6 7 8 9 10	
Bootstrap 1	10 1 8 2 2 1 5 5 5 2	3 4 6 7 9
Bootstrap 2	10 2 5 9 4 7 8 1 7 1	3 6
Bootstrap 3	4 10 7 2 7 8 10 9 5 8	1 3 6
Bootstrap 4	5 10 6 6 10 6 7 10 10 6	1 2 3 4 8 9

- En un dataset de tamaño N la probabilidad de un dato dado de ser elegido es

$$\mathbb{P}(\text{elegido}) = 1 - \left(1 - \frac{1}{N}\right)^N \approx 1 - \frac{1}{e} = 0.632$$

Bagging (Bootstrap Aggregation)

Out-of-bag error (OOB error):

- Sea S_{OOB} las instancias que han sido OOB para algún h_k.
- Sea $(\boldsymbol{x}, y)$ una instancia dada en S_{OOB}.
- Tomar los modelos h_k que no hayan sido entrenados con $(\boldsymbol{x}, y)$.
- El error OOB en $(\boldsymbol{x}, y)$:

  Voto mayoritario/promedio de estos $h_k(\boldsymbol{x})$ vs. el valor real y.
- El error OOB en S_{OOB}:

  Promediar el error OOB de $(\boldsymbol{x}, y)$ para todas las instancias en S_{OOB}.

Árboles de decisión

Ventajas

- Alto poder *explicativo*
- Representación visual

Desventajas

- Suelen tener menor *exactitud* que otros modelos
- Poco *robustos* o *alta varianza*
  - un pequeño cambio en los datos puede generar un gran cambio en el Árbol

Bajar la varianza ...

Ensemble: Random Forest

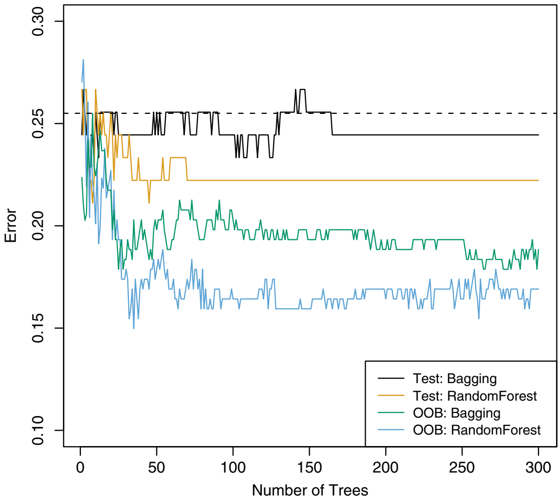
Para $k \in \{1, \ldots, K\}$

- Construir B_k como en bagging
- Elegir aleatoriamente un conjunto A_k de atributos con $\#A_k \ll \#\text{Atributos}$
- Usualmente $A_k \sim \sqrt{\#\text{Atributos}}$
- Construir el árbol h_k usando B_k y A_k

Generaliza mejor que bagging

Ensemble: Bagging vs RF

- Heart dataset (clasificación binaria) $N = 303$ pacientes.
- Sí indica la presencia de enfermedad cardíaca.
- No significa que no hay enfermedad cardíaca.
- Hay 13 predictores que incluyen edad, sexo, colesterol, etc.



Ensemble: Random Forest y Variable Importance

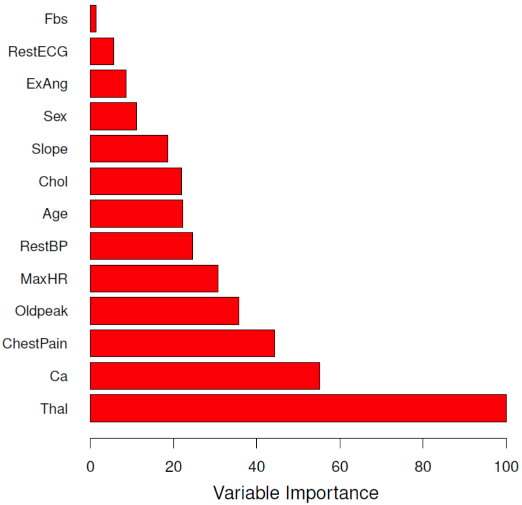
En los árboles tenemos una manera directa de medir la importancia atributos:

- Cada nodo divide un dataset en base a un atributo.
- Se basa en la impureza (Gini/entropía en clasificación, MSE en regresión).
- Cuánto contribuye un atributo a disminuir la impureza ponderada:

  disminución media en las impurezas ponderadas para todos los nodos que se dividieron con dicho atributo.

- En Random Forest promediar la disminución de impurezas sobre los árboles.

Ensemble: Random Forest y Variable Importance



Bibliografía

- An introduction to statistical learning with applications in Python. Cap 8.
- Machine Learning - A First Course for Engineers and Scientists. Capítulo 7.
- Raschka, S. Introduction to Machine Learning. Lecture 7. (2021)
- M. Stamp. Introduction to ML with Applications to Information Security, 7.4 y 7.5.
- Zhi-Hua Zhou. Ensemble Methods: Foundations and Algorithms.