Métodos de

Ensamble

M.Sc. Angelo Jonathan Diaz Soto

2025

(+51) 976 760 803 Data Science – Business Intelligence – Big Data – Machine Learning – Artificial Intelligence – Innovation and
www.datayanalytics.com info@datayanalytics.com

Introducción

- El término **ensamblado** se refiere al conjunto de metodologías que permiten combinar varios modelos para dar lugar a un **meta-algoritmo** que mejore los resultados de los modelos individuales que lo forman.
- Los métodos de ensamble de modelos o métodos combinados intentan ayudar a **mejorar el rendimiento de los modelos de Machine Learning** al mejorar su precisión. Este es un proceso mediante el cual se construyen estratégicamente varios modelos de Machine Learning para resolver un problema particular.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Ejemplo

quieres
pero no

buscas

acción
anual.
de cada

Supongamos que **invertir en una empresa**, estas seguro de su rendimiento por lo que a varios expertos para te aconsejen si el precio de la aumentará en más de 6% Estas fueron las respuestas una de las personas consultada:

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Ejemplo

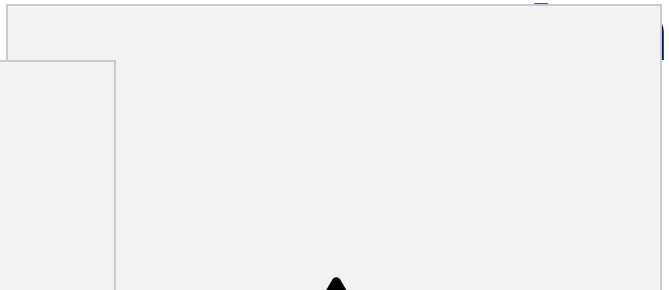
- Dado el amplio espectro de acceso que tiene, probablemente pueda combinar toda la

información y tomar una decisión informada.

- El supuesto utilizado aquí de que todas las predicciones son completamente independientes es ligeramente extremo, ya que se espera que estén **correlacionados**. Sin embargo, se puede mejorar la decisión combinando varios pronósticos.
- El aprendizaje de ensamblado de modelos no es diferente a nuestro ejemplo anterior.

plo

Esta es la



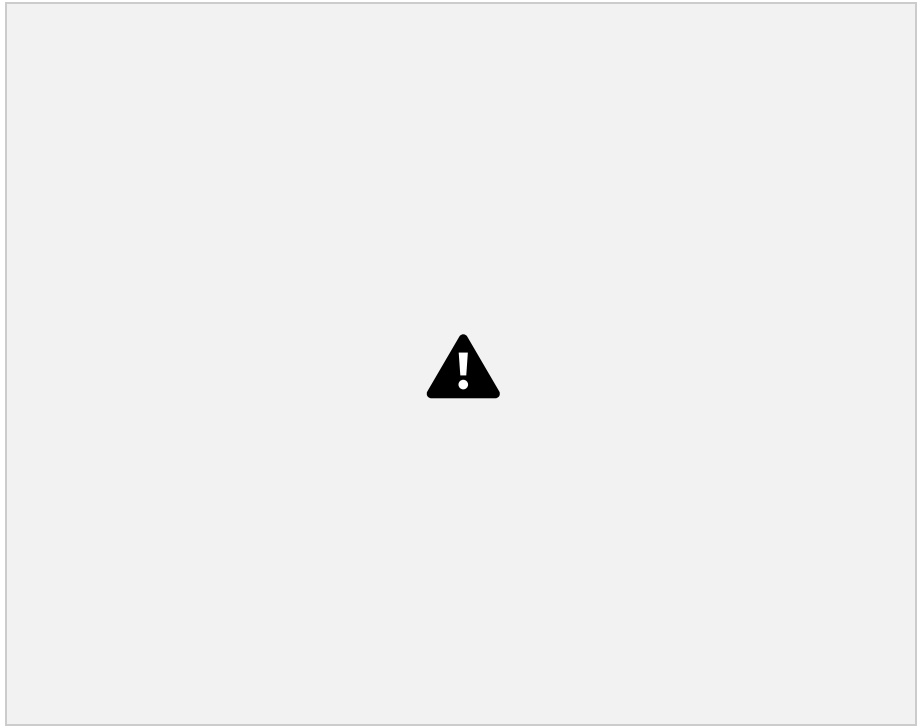
idea básica de un conjunto: **combinar predicciones de varios modelos**, promedia errores idiosincráticos y produce mejores predicciones generales. La siguiente imagen muestra un ejemplo de los esquemas de un conjunto:

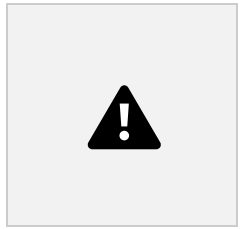
¿Por qué surgen los ensambladores de árboles?

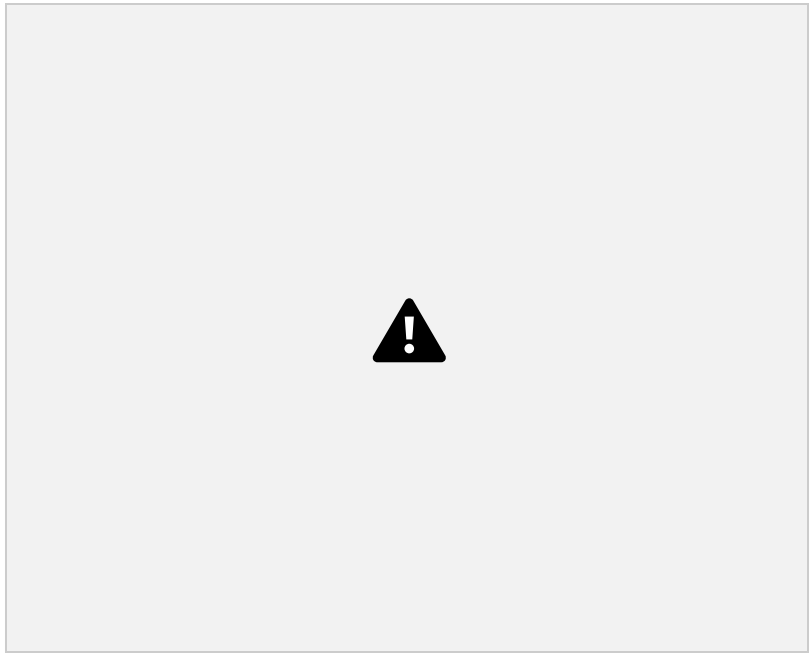
- Así como todos los modelos, un árbol de decisión también sufre de los

problemas de sesgo y varianza. Es decir, cuánto en promedio son los valores predecidos diferentes de los valores reales (sesgo) y cuán diferentes son las predicciones de un modelo en un mismo punto si muestras diferentes se tomarán de la misma población' (varianza).

- ❏ Al construir un árbol pequeño se obtendrá un modelo con baja varianza y alto sesgo. Normalmente, al incrementar la complejidad del modelo, se verá una reducción en el error de predicción debido a un sesgo más bajo en el modelo. En un punto el modelo será muy complejo y se producirá un sobre-ajuste del modelo el cual empezará a sufrir de varianza alta.
- ❏ El modelo óptimo debería mantener un balance entre estos dos tipos de errores. A esto se le conoce como "**trade-off**" (equilibrio) entre errores de sesgo y varianza.
- ❏ El **uso de ensambladores** es una forma de aplicar este "trade-off".



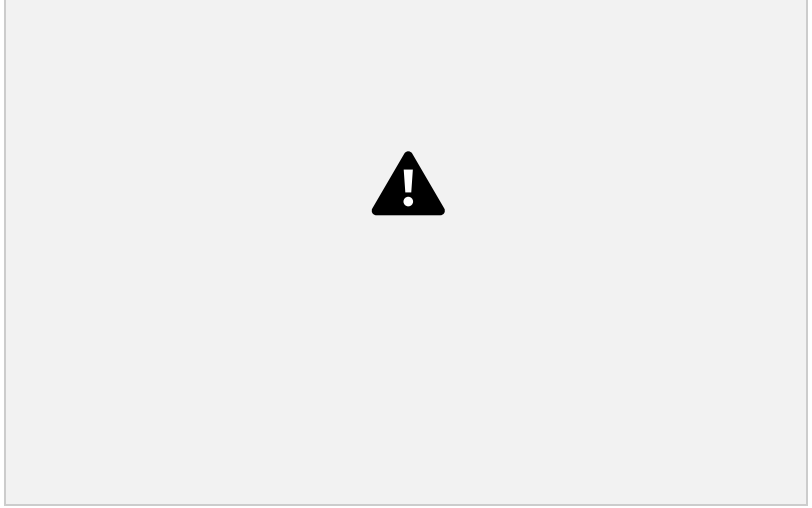




(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Tipos de métodos de ensamble

1 Votación por mayoría



Stacking

2
2

Bagging

3

Random Forest

4

Boosting

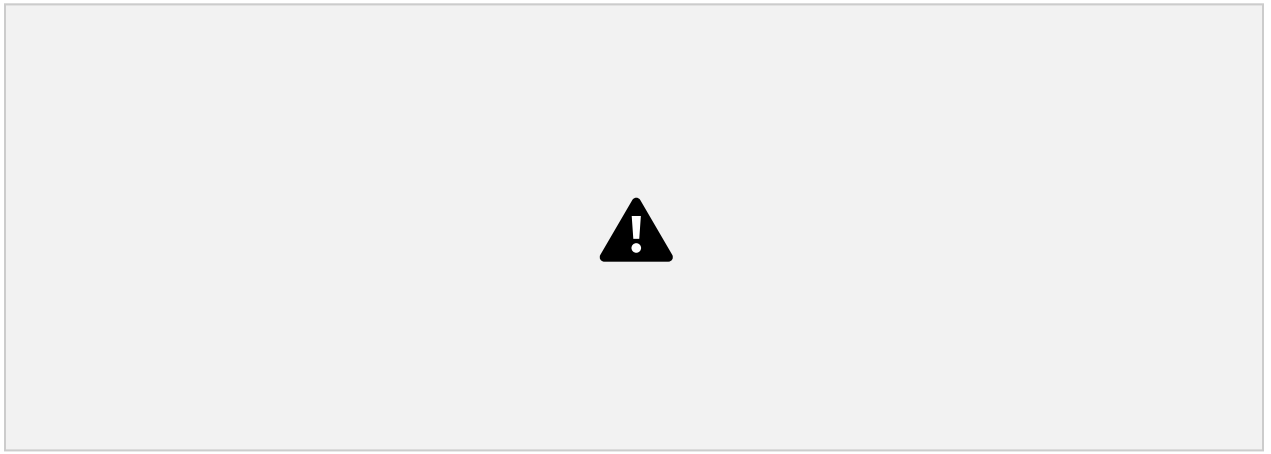
5

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Votación por mayoría

Podemos entrenar varios modelos de aprendizaje automático con los mismos datos. Cuando tengamos datos

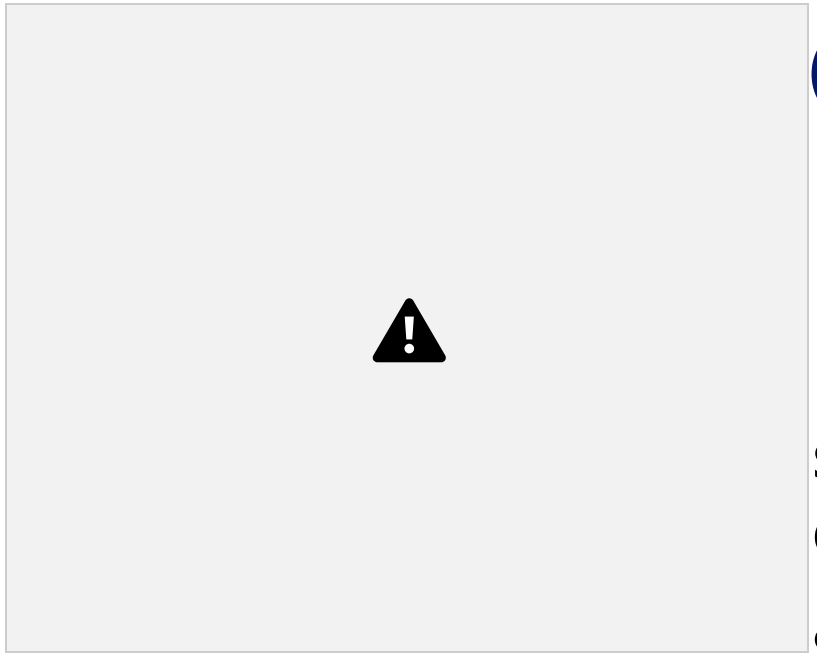
nuevos, obtendremos una predicción de cada modelo. Cada modelo tendrá asociado un voto. De esta forma, propondremos como predicción final lo que voten la mayoría de los modelos.



Votación por mayoría

- Hay otra forma de combinar las votaciones. Cuando los modelos de machine learning dan una probabilidad, podemos usar el «voto suave» **(soft-voting)**.
- Cuando usamos modelos diferentes, los errores se compensan y la predicción combinada generaliza mejor.
- Por eso, **no tiene sentido hacer un ensemble de votación por mayoría con el mismo tipo de modelo**.

(modelos apilados))



- Cuando hablamos de un ensemble de stacking, nos referimos a que estamos apilando modelos.
- Cuando apilamos modelos, lo que en realidad estamos haciendo, es usar la salida de varios modelos como la entrada de varios modelos.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

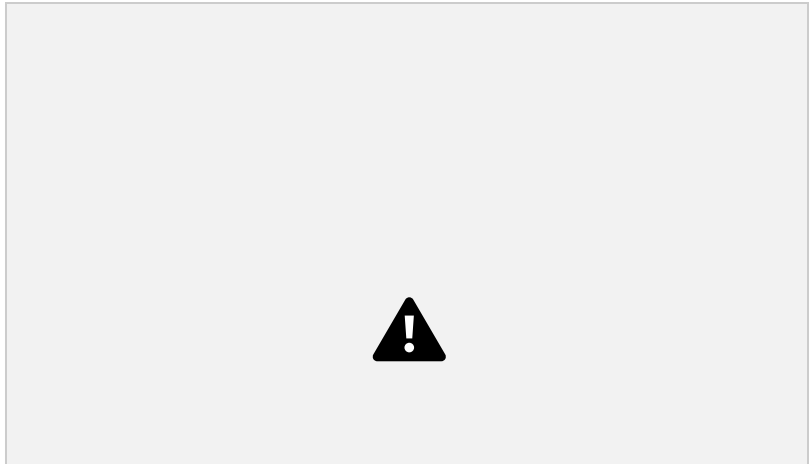
Bagging

El método de **bagging** o bootstrap aggregation es un procedimiento utilizado para **reducir la varianza** de un método de aprendizaje estadístico, usado muy frecuentemente con árboles de decisión. El Bagging es una de las técnicas de también se construcción de conjuntos que Bootstrap. conoce como Agregación extraen Dada una muestra de datos, se varias muestras, **bootstrapped**.

Esta **selección se realiza de manera aleatoria**, es decir, cada variable se puede elegir de la población original, de modo que cada variable es igualmente probable que se seleccione en cada iteración del proceso de arranque.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Bagging

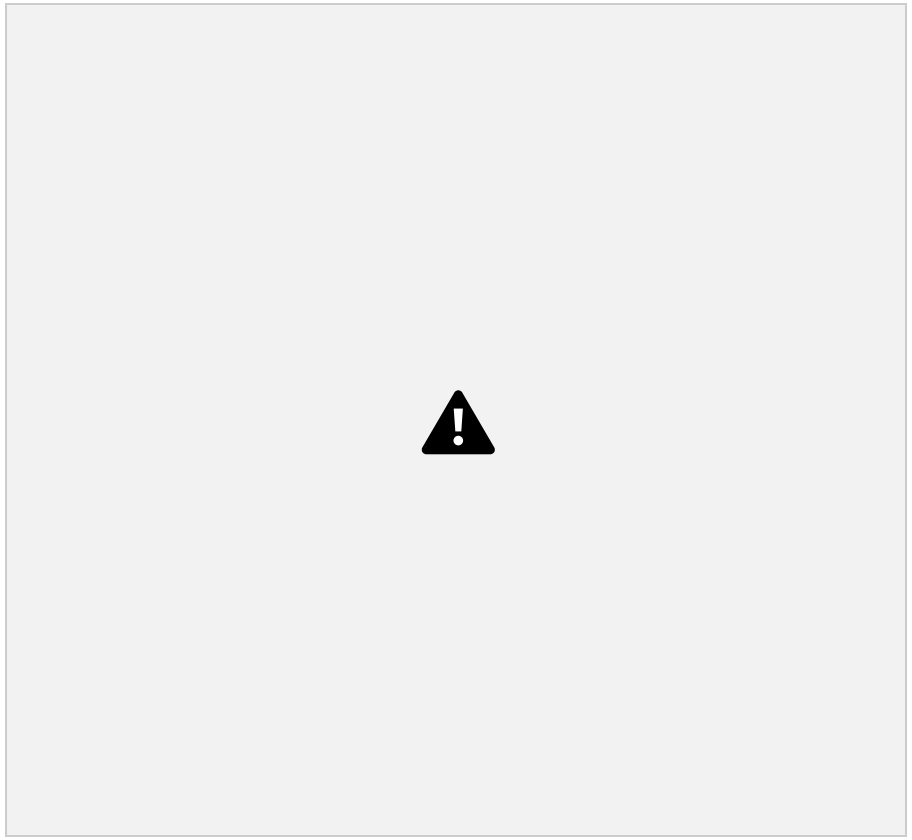


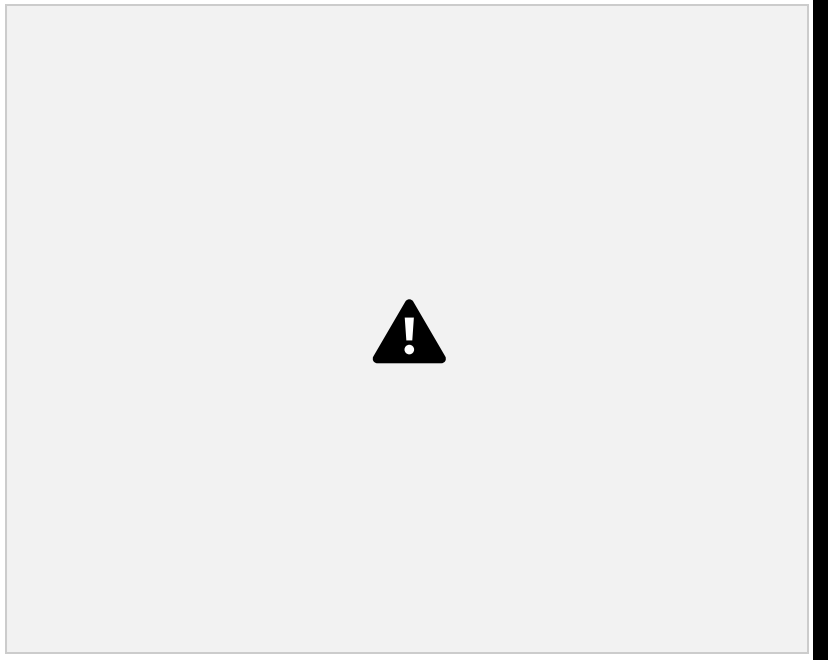
• Una vez que forman las muestras bootstrapped, se entrenan los modelos de manera separada. • Toma en cuenta que las

muestras bootstrapped se extraen del conjunto de entrenamiento y los submodelos se prueba utilizando el conjunto de prueba.

- La **predicción** de salida final **se combina** en las proyecciones de todos los submodelos.

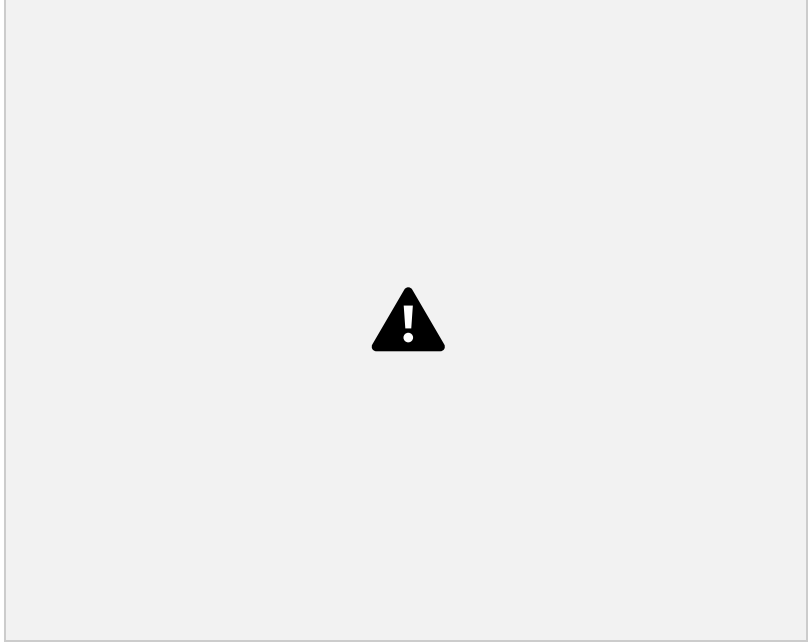
(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com





(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Bagging



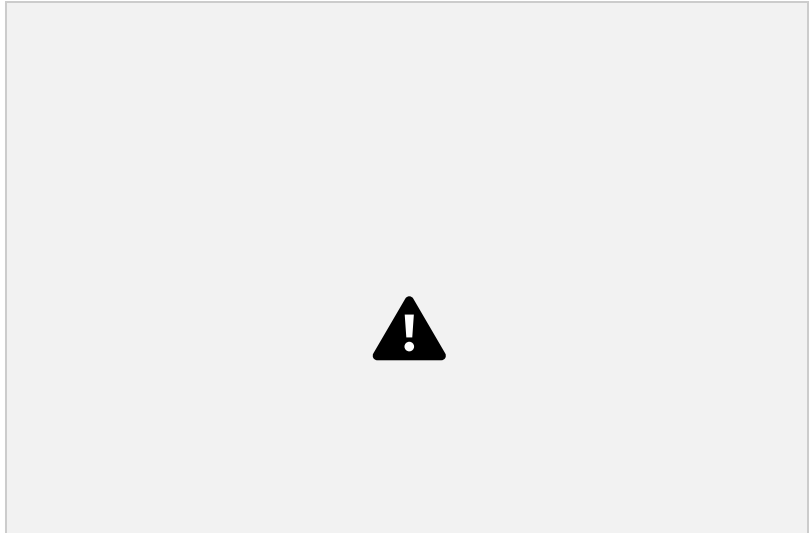
En resumen el algoritmo de Bagging consiste en:

- Crear múltiples subconjuntos de datos.
- Construir múltiples modelos
- Combinar los modelos

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Random Forest

- una



Random forests proporciona mejora a los árboles combinados por bagging en cuanto a que los **decorrelaciona**, teniendo en cuenta sólo un subgrupo de

predictores en cada división.

- Al igual que en el bagging, se construyen un número de árboles de decisión a partir de pseudo-muestras generadas por **bootstrapping**

Random Forest

- Útil para regresión y clasificación.
- Se generan múltiples árboles (a diferencia de CART).
- Cada árbol da una clasificación (vota por una clase), y el resultado es la clase

con mayor número de votos en todo el bosque (forest). Para regresión, se toma el **promedio** de las salidas (predicciones) de todos los árboles.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

¿Cómo se construye un modelo random forest?

[redacted]

1 Dado que el número de casos en el conjunto de entrenamiento es *N* . Una muestra [illegible] riamente pero **CON REEMPLAZO**. Esta mu[illegible]namiento para construir el árbol *i*.

Si existen *M* [illegible] variables de entrada, [illegible]

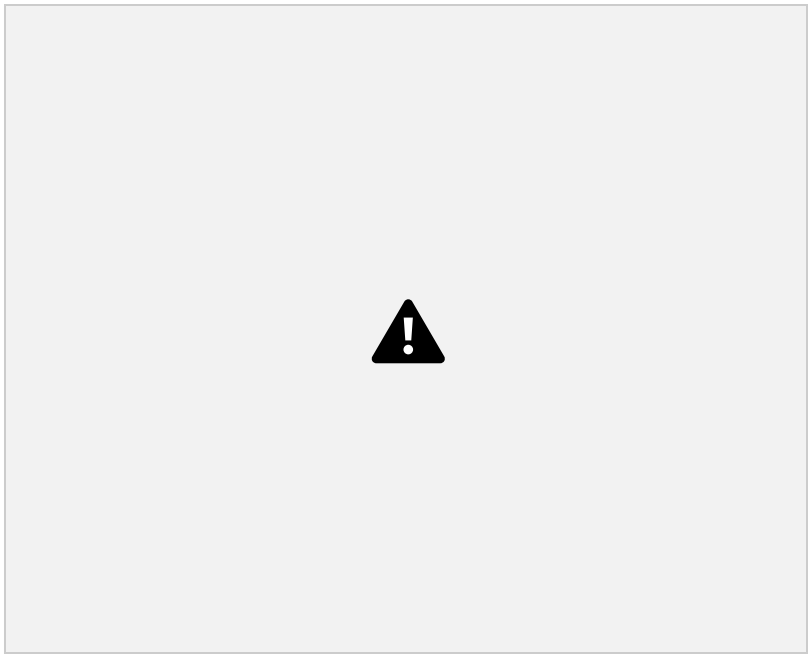
un número $m < M$ se especifica tal que

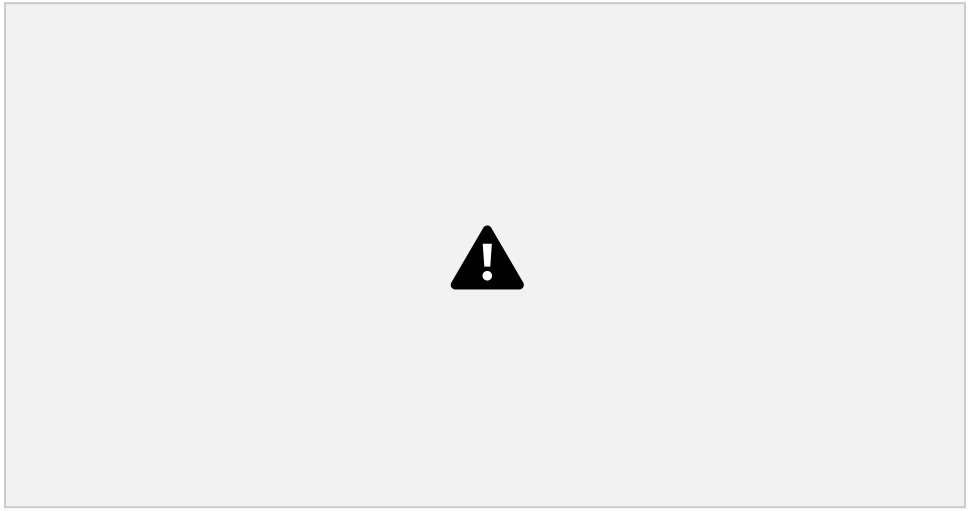
2. para cada nodo, m variables se selecciona aleatoriamente de M . La mejor división de estos m atributos es usado para ramificar el árbol. El valor m se mantiene constante durante la generación de todo el bosque. Cada árbol crece hasta su máxima extensión posible y **NO hay proceso de**
3. **poda.**
4. Nuevas instancias se predicen a partir de la **agregación de las predicciones** de los x árboles (i.e., mayoría de votos para clasificación, promedio para regresión)

Out of bag samples y out of bag error

- El proceso de muestreo de los datos con reemplazo se denomina bootstrap.

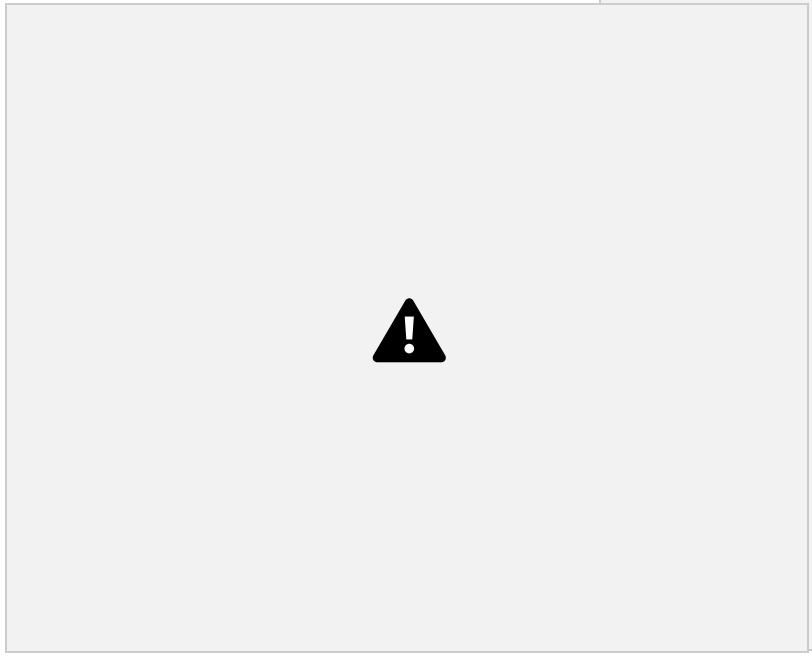
- Un tercio de los datos no se usan para el entrenamiento y pueden ser usados para test.
- Este conjunto se denomina out of bag (OOB)samples.





Out of bag samples y out of bag error

- El error estimado en estos



out of bag samples se conoce como out of bag error (OOB error).

- Usar este conjunto de test (OOB) es tan preciso como

si se usara un conjunto de test del mismo tamaño que el de entrenamiento.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Random Forest: Hyper-parámetros

El hyper-parámetro más importante para ajustar **es el número de variables candidatas a seleccionar para evaluar cada ramificac** lgunos adicionales que deben c

ntree: número de árboles en el bosque. Se quiere

estabilizar el error, pero usar demasiados árboles puede ser innecesariamente ineficiente.

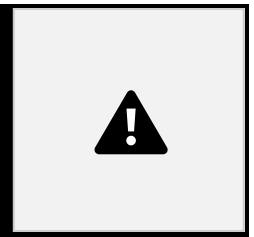
mtry: número de variables aleatorias como candidatas en cada ramificación. **nodesize:** mínimo número de muestras dentro de los nodos terminales.Equilibrio entre bias-varianza

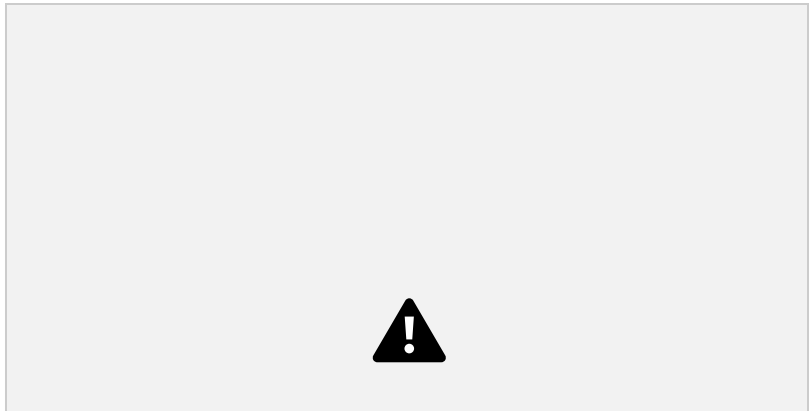
sampsize: el número de muestras sobre las cuales entrenar. El valor por defecto es 63.25%. Valores más bajos podrían introducir sesgo y reducir el tiempo.Valores más altos podrían incrementar el rendimiento del modelo pero a riesgo de causar overfitting. Generalmente se mantiene en el rango 60-80%.

maxnodes: máximo número de nodos terminales.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Ventajas de Random Forest





- Puede manejar hasta miles de variables de entrada e identificar las más significativas. **Método de**

reducción de dimensionalidad.

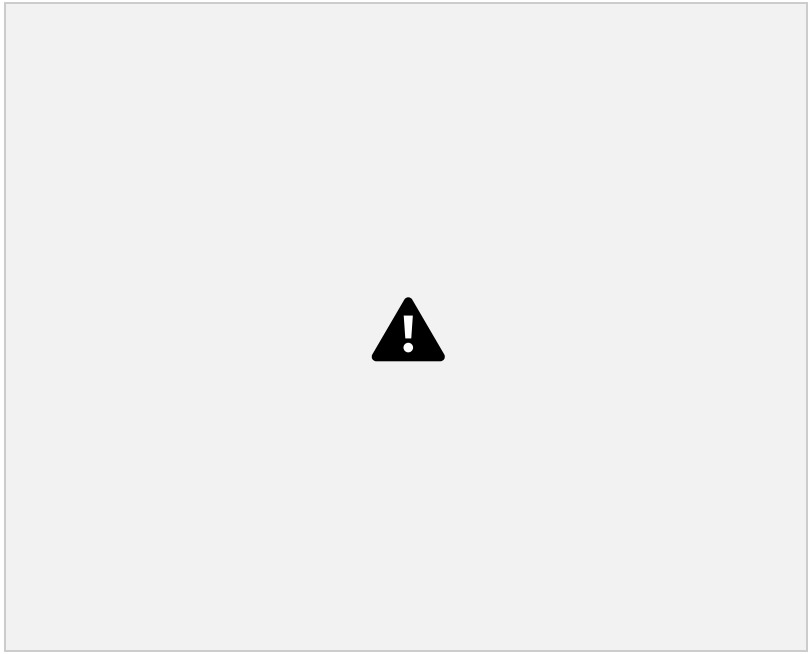
➢ Una de las salidas del modelo es la **importancia de variables**.

➢ Incorpora métodos efectivos para **estimar valores faltantes**.

Desventajas de Random Forest

- Pérdida de interpretación
- **Bueno para clasificación**, no tanto para regresión. Las predicciones no son de naturaleza continua.
- En regresión, no puede predecir más allá del rango de valores del conjunto de entrenamiento.
- Poco control en lo que hace el modelo (**modelo caja** negra para modeladores estadísticos)

Boosting

- 

Boosting funciona de manera parecida al bagging en cuanto a que combina un gran número de árboles, a excepción de que **los árboles se construyen de manera secuencial.**

- Otra diferencia es que boosting **no utiliza remuestreo por bootstrapping**, sino que cada árbol se genera utilizando una versión modificada del set de datos original.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Principales parámetros del Boosting

- **Número de árboles** (B). A diferencia del bagging y random forests, boosting puede sobre ajustarse a los datos si el número de árboles es demasiado alto. B se selecciona por validación cruzada. **Número de divisiones** (d) en cada árbol, que controla el nivel de

complejidad. Un valor de $d = 1$ (cada árbol contiene una única división, es decir, un único predictor) suele dar buenos resultados. **Parámetro de penalización** (λ), que controla el ritmo con el que boosting aprende. Valores comunes para este parámetro suelen ser 0,01 o 0,001, aunque la decisión depende del problema en cuestión.

(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com

Algunos de los algoritmos de boosting más utilizados son:

AdaBoost

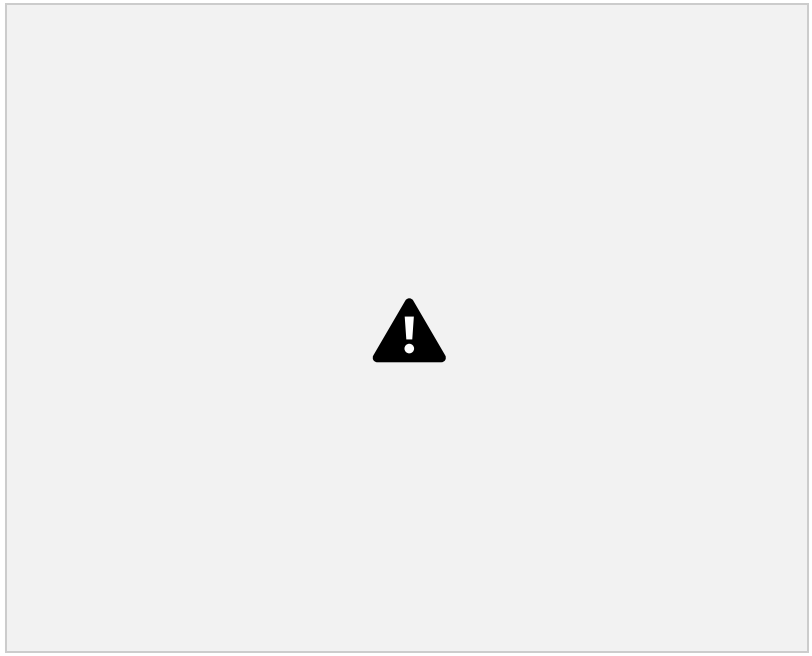
- Gradient Boosting
- Stochastic Gradient

Referencias Bibliográficas

Valiant, L. G. (1984). A Theory of the Learnable. Communications of the ACM, 27(11), 1134-1142.

https://doi.org/10.1145/800057.808710

- An Introduction to Statistical Learning: with Applications in R (2013). James G., Witten D., Hastie T., Tibshirani R.
- The Elements of Statistical Learning (2009). Hastie, Trevor, Tibshirani, Robert, Friedman, Jerome.
- Stochastic Gradient Boosting (1999). Jerome H. Friedman.



(+51) 976 760 803 www.datayanalytics.com info@datayanalytics.com