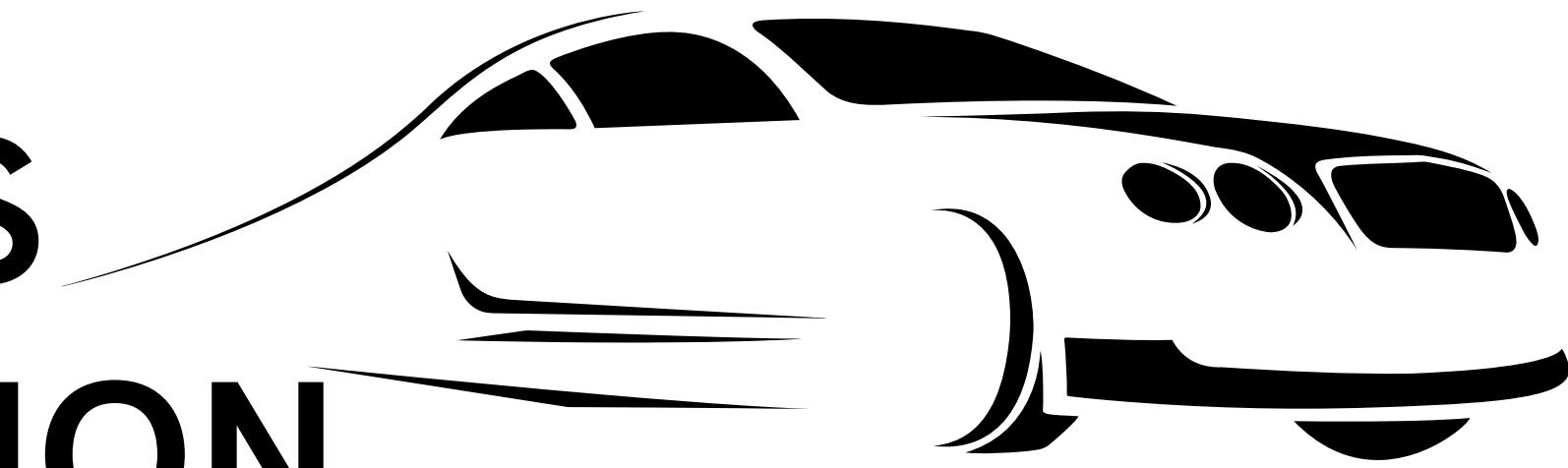


BORIS RENDÓN

USED VEHICLES PRICE PREDICTION

MACHINE LEARNING MODELS



AGENDA

- 1 Explicación del problema
- 2 Limpieza de datos
- 3 Split de data
- 4 Modelos
- 5 Conclusiones

Qué queremos de los modelos?

Predecir el precio de los carros usados

Utilizando modelos de regresión

MAE

Mean Square Error

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i|$$

Limpieza de datos

Mi primer paso es chequear los valores nulos de todas las variables.

Este dataset cuenta con 10 variables(incluyendo la objetivo).

De las 10 variables sólo **tax** y **mpg** contaban con valores nulos pero no sobrepasaron el 10% del total

Limpieza de variables numéricas

Como features numéricos tenemos:

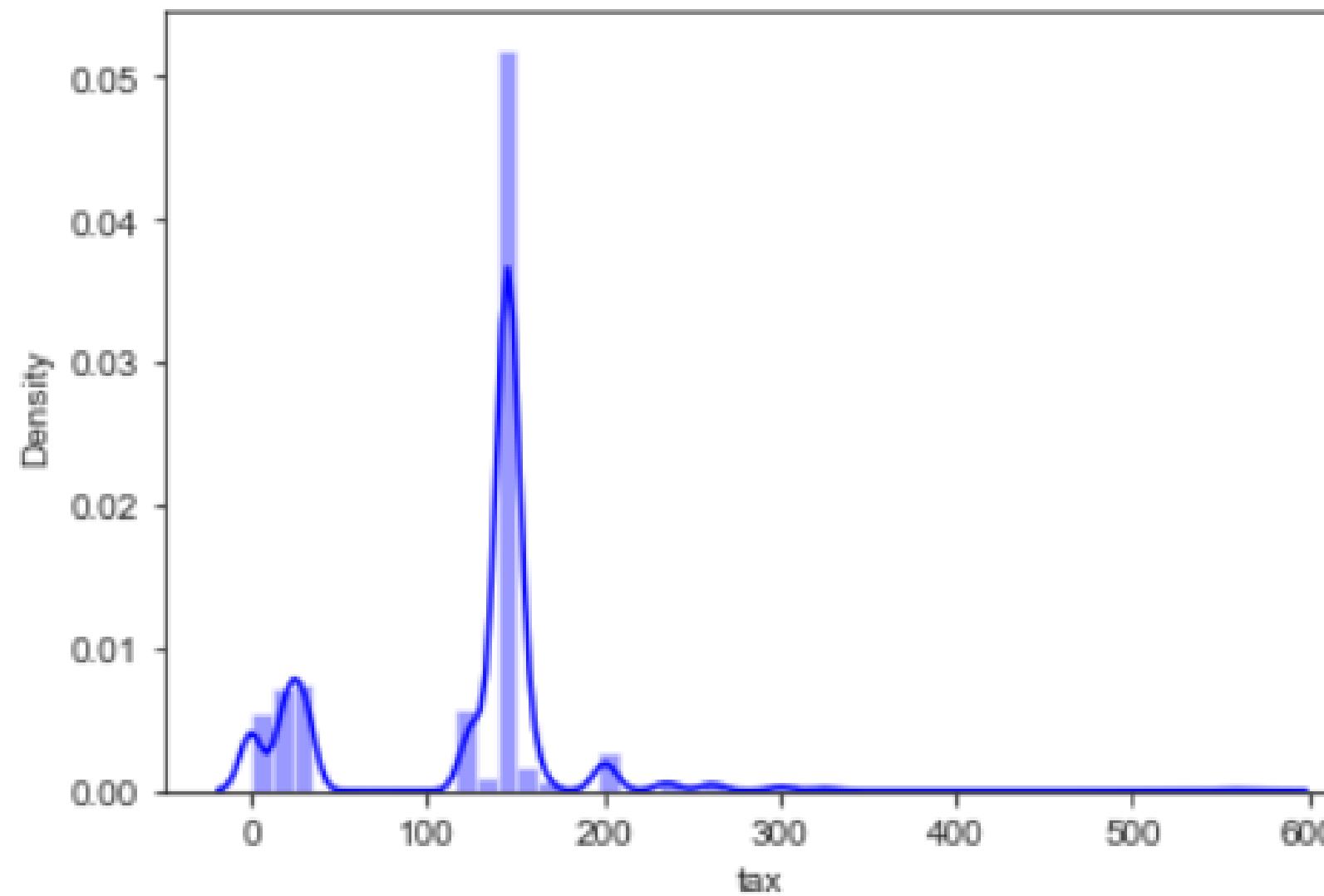
- year
- mileage
- tax
- mpg
- engineSize

Para **year** elegimos solo los modelos arriba del 2012, ya que representan algún valor significativo

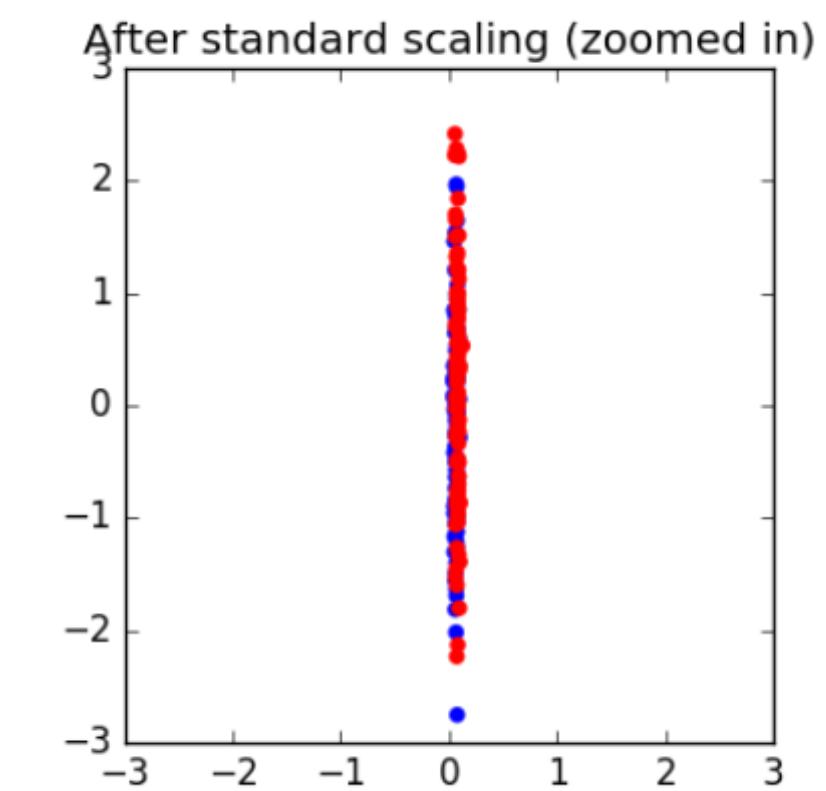
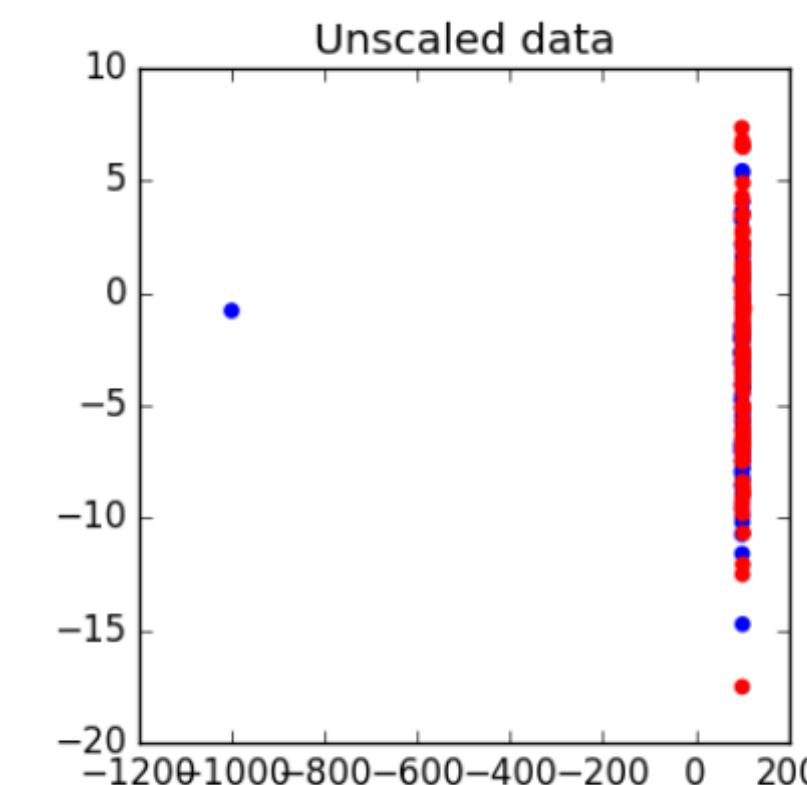
Para **tax** y **mpg** utilizamos una imputación por media a los valores

Distribución de datos

Al revisar las distribuciones nos dimos cuenta que ningún dato contaba con una distribución normal



Por lo que procedimos a usar el `StandardScaler()` de la librería `sklearn` y ya contabamos con la data normalizada



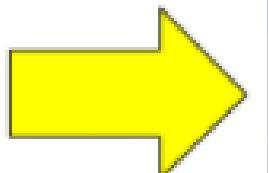
Limpieza de variables categóricas

Como features categóricos tenemos:

- model
- transmission
- fuelType
- make

Decidí botar la variable **model** ya que tenía demasiados valores diferentes y iba a perjudicar mi resultado

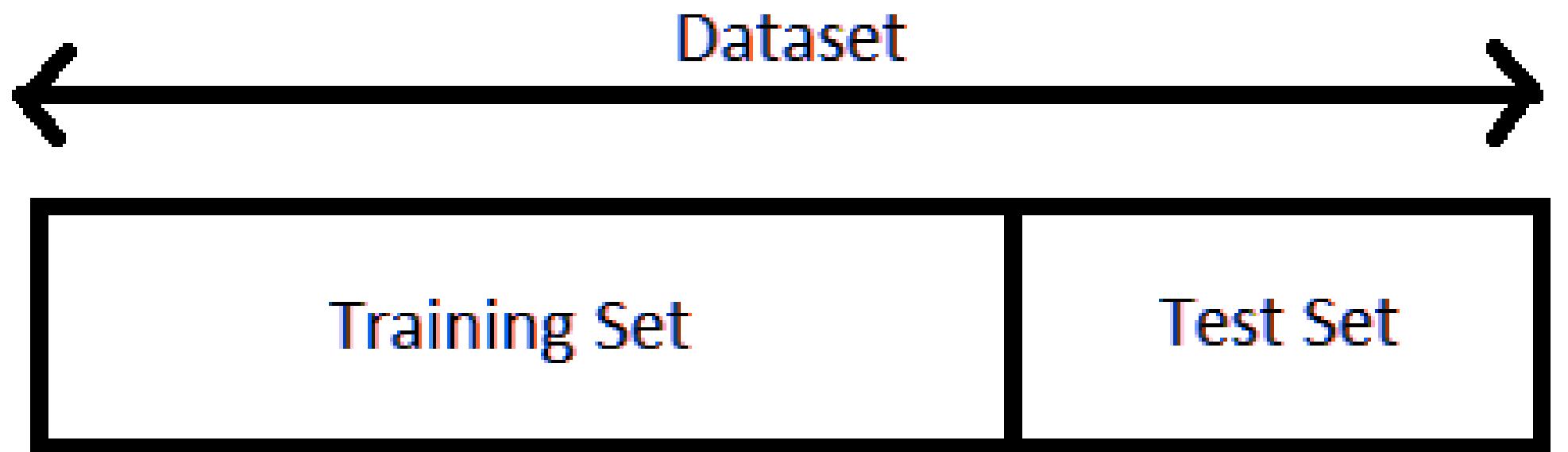
Para las variables categóricas utilizamos **OneHotEncoder** de la librería **sklearn**



Color	Red	Yellow	Green
Red	1	0	0
Red	1	0	0
Yellow	0	1	0
Green	0	0	1
Yellow			

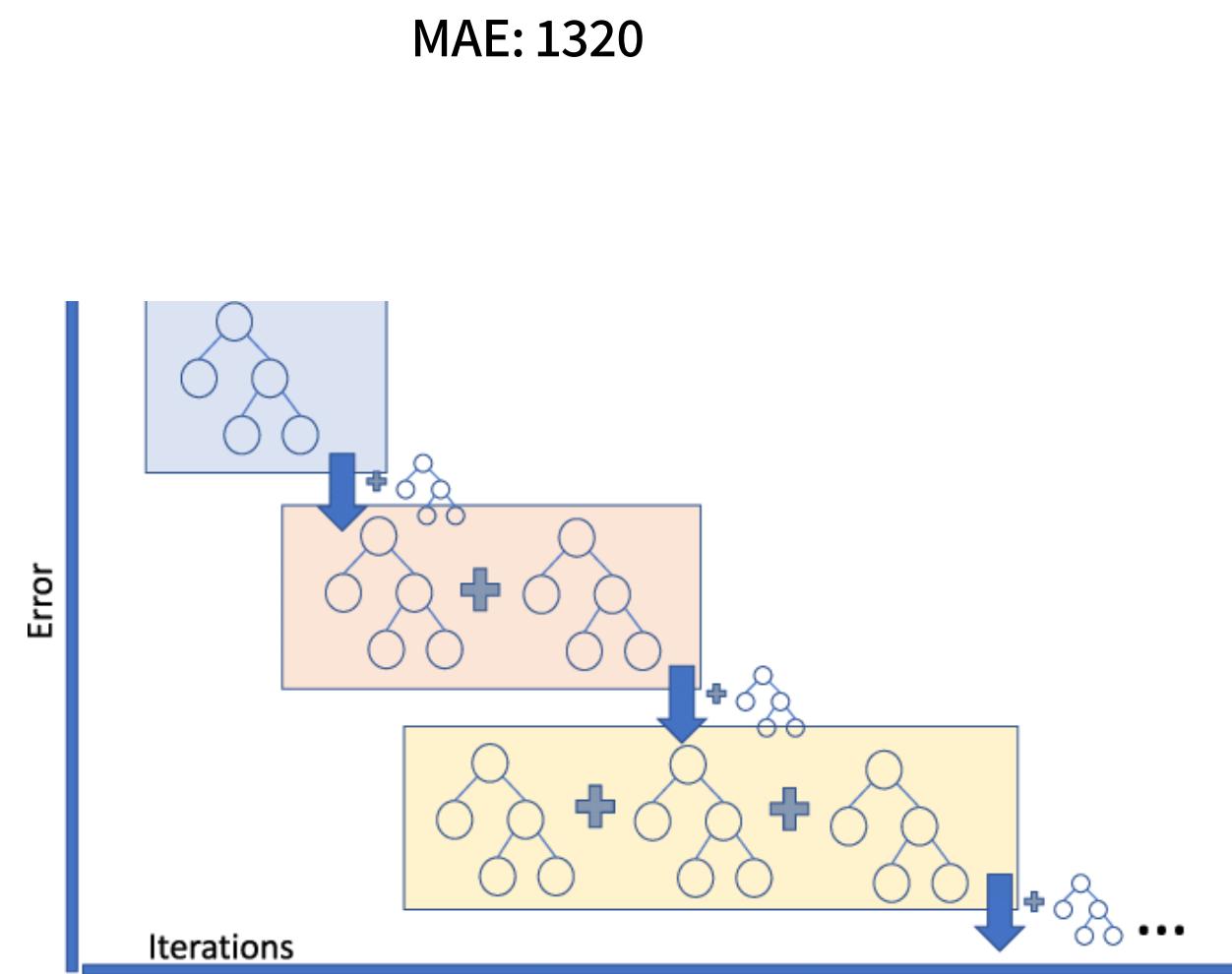
Split de data

Dedicamos el 30% de la data para testing y 70% para training, también utilizamos shuffle=True, que nos sirve para evitar que exista un overfitting.



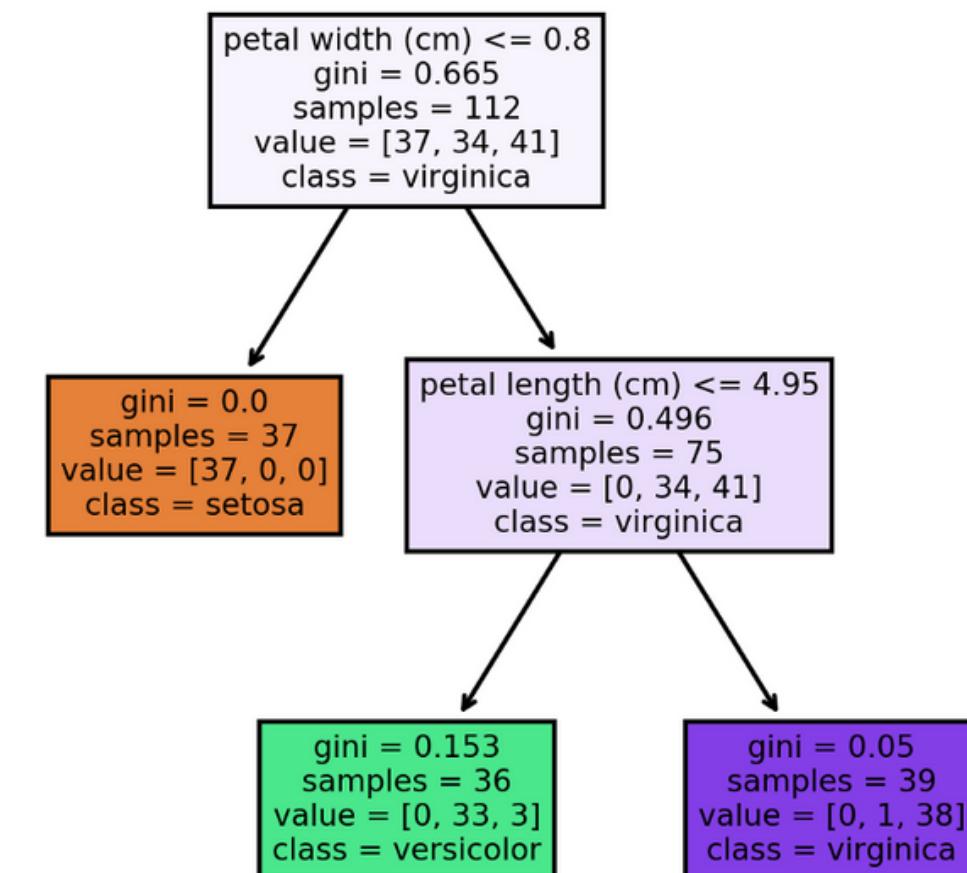
Modelos

Gradient Boosting Regressor



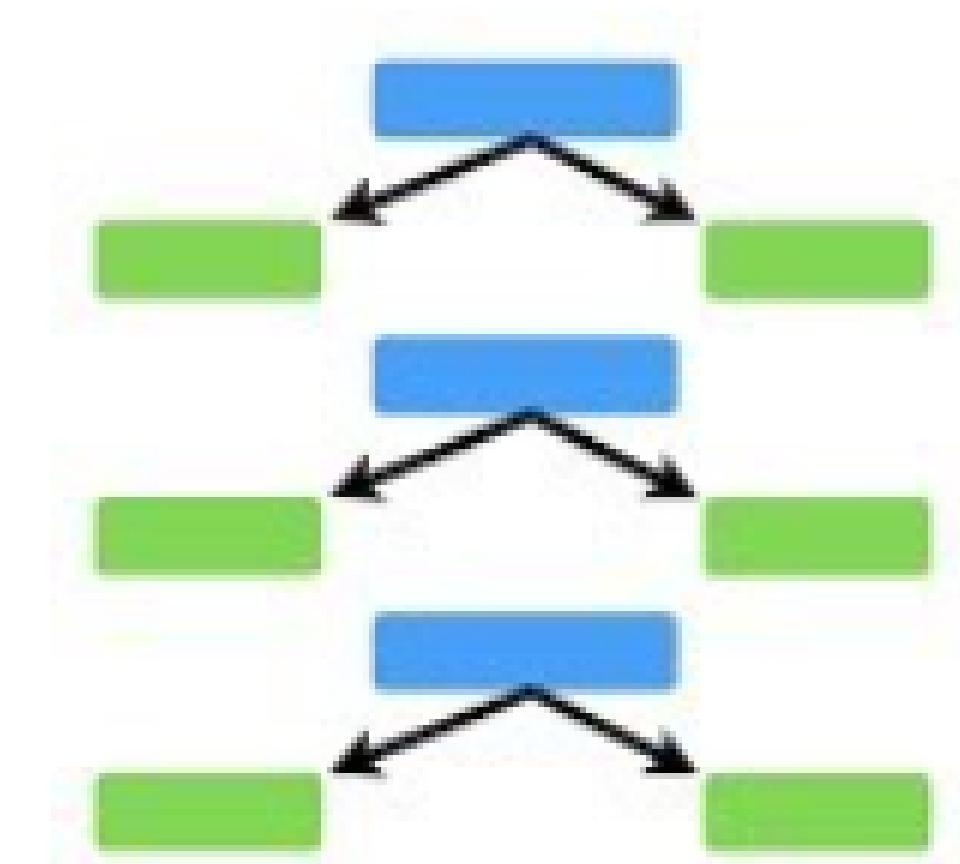
Decision Tree

MAE: 2675



Adaboost

MAE: 7900



Conclusión

Era de esperarse que el gradient Boosting sea el mejor algoritmo

Gradient boosting aunque es similar a AdaBoost, es un algoritmo mucho más robusto y nos beneficiamos de poder ponerle profundidad a los árboles

El mejor modelo tiene los hiper parámetros de 450 estimadores, random state de 42 , learning rate de 0.1 y un max depth de 8