MACHINE LEARNING CON PYTHON

LUIS CHACON MONTALVAN

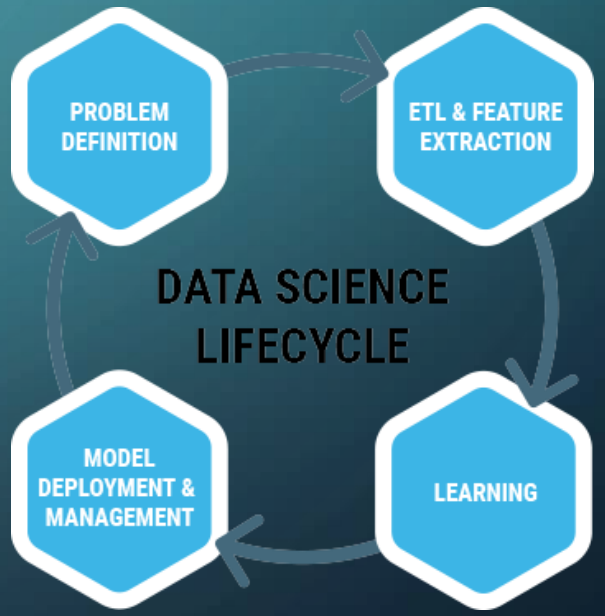
MODELIZACIÓN

¿Que es un modelo?

No existe una definición única de modelo, y muchas veces depende de la bibliografía que consultas.

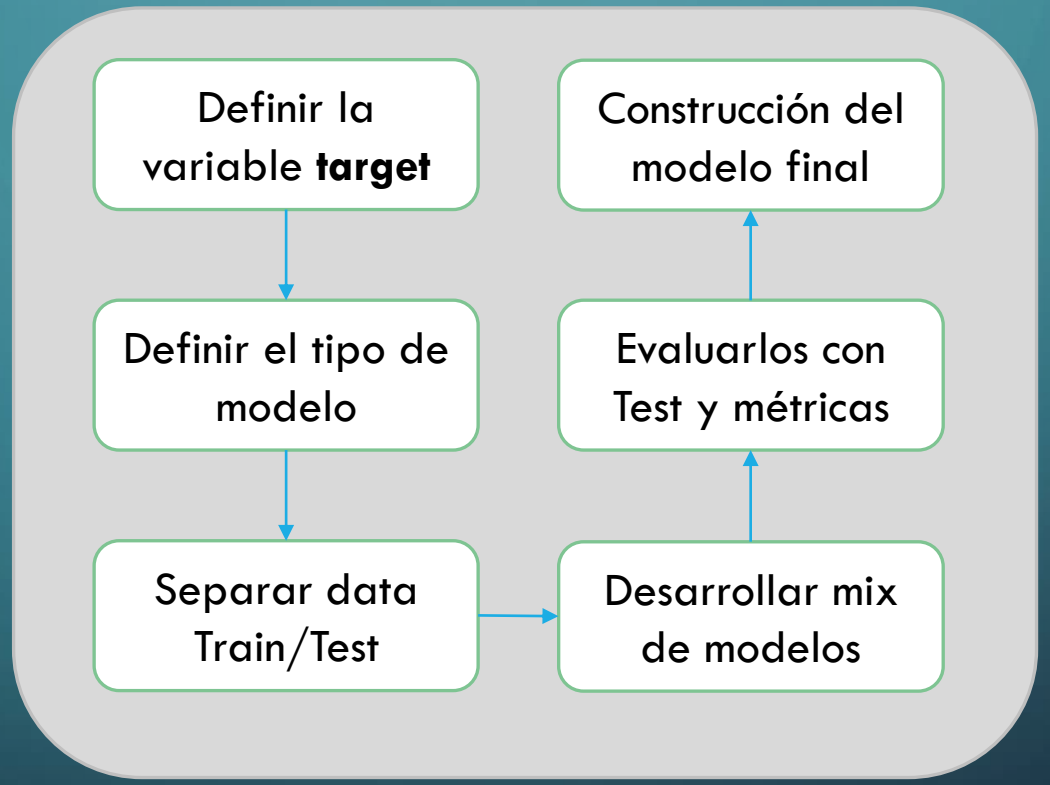
- Modelos determinísticos
- Modelos probabilísticos
- Modelos de Machine Learning

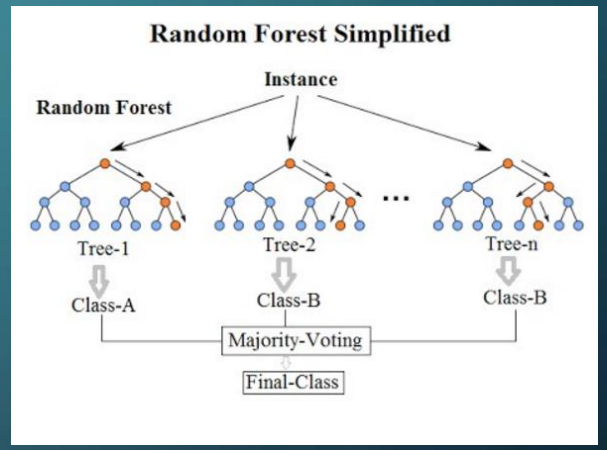
El modelado predictivo es un proceso que utiliza la extracción de datos y la probabilidad para pronosticar resultados.
Cada modelo se compone de una serie de predictores, que son variables que probablemente influyan en los resultados futuros. Una vez que se han recopilado datos para los predictores relevantes, se formula un modelo estadístico. El modelo puede emplear una ecuación lineal simple, o puede ser una red neuronal compleja, diseñada por un software sofisticado.



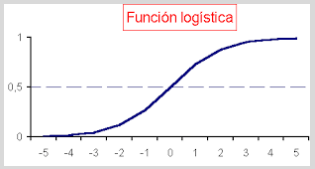
MODELIZACION

- Identificamos un problema
- ETL datos
- Entendimiento de los datos
- Arte de modelar

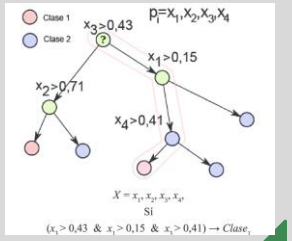




MODELOS DE CLASIFICACIÓN

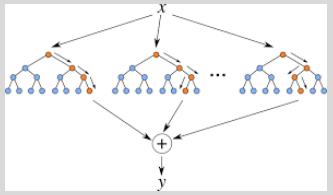


Regresión logística



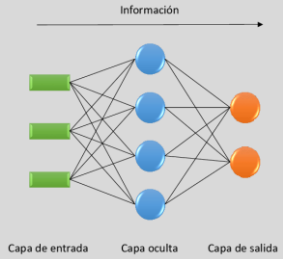
Árboles de clasificación

Bosque aleatorio



Naive Bayes

Red Neuronal



La exactitud y precisión de los modelos depende de siempre del set de datos con los que trabajamos, ellos definen quien es el mejor.

Ojo!!! No existe un modelo ideal, no existe un modelo único

REGRESIÓN LOGÍSTICA

La Regresión Logística consiste en predecir una variable respuesta cualitativa (**Y**) basado en una o varias variables predictoras (**X**). Modela la probabilidad de que la variable respuesta Y pertenezca a una categoría particular en función de las variables explicativas usando una función logística. Usada frecuentemente para modelar situaciones donde la variable respuesta es binaria o dicotómica.

Modelo: $$log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$$

Donde:

- X: Variables predictoras o independientes
- Y: Variable objetivo, target o dependiente
- β: Coeficientes

MÉTRICAS PARA COMPARAR MODELOS

Matriz de confusión

En problemas de clasificación, la Exactitud y el Error son las medidas básicas de desempeño del modelo. En general, permite saber si el modelo es bueno o malo y se calcula de la siguiente manera:

$$\text{Exactitud} = \frac{\text{Clasificados Correctamente}}{\text{Total de Clasificados}}$$

$$\text{Error} = 1 - \text{Exactitud}$$

		Predicción		
		Positivo	Negativo	Total
Verdad	Positivo	Verdadero Positivo (TP)	Falso Negativo (FN)	P
	Negativo	Falso Positivo (FP)	Verdadero Negativo (TN)	N
	Total	P*	N*	

$$\text{Precisión} = \frac{TP}{P*} \quad , \quad \text{Sensibilidad } (\textit{True Positive Rate}) = \frac{TP}{P} \quad , \quad \textit{False Positive Rate} = \frac{FP}{N} = 1 - \text{Especificidad}$$

DATOS DESBALANCEADOS

		Predicción		Total
		Si	No	
Verdad	Si	90	0	90
	No	10	0	10
Total		100	0	100

Exactitud	0.9
Error	0.1
Sensibilidad	1.0
Especificidad	0.0

Los algoritmos de clasificación funcionan bien cuando el tamaño de las clases de la variable objetivo son similares, pero ¿esto se da siempre en la realidad?

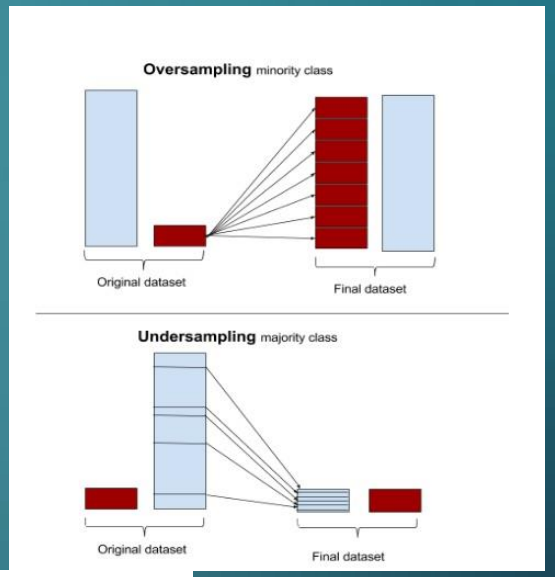
- Cuando analizamos la fuga de clientes, la mitad se va?
- Cuando analizamos la entrega de un crédito, la mitad accede al crédito?
- Cuando analizamos los tumores, la mitad son cancerígenos?

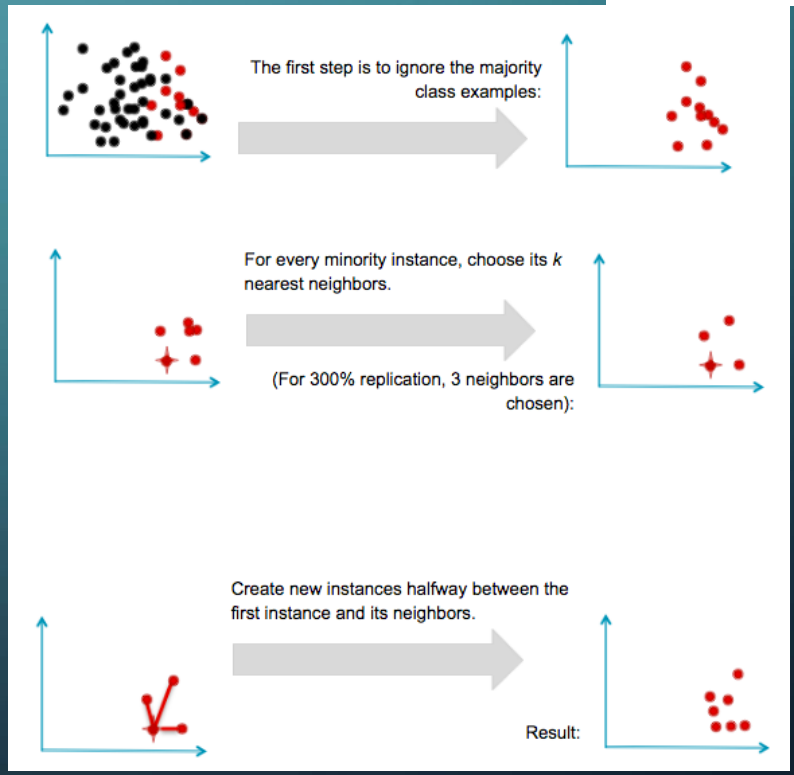
Y así podríamos seguir con la lista, pero lo que ocurre es que necesitamos modelar un evento que ocurre una cantidad de veces muy reducida, por lo cual estaremos ante el problema de la datos desbalanceados o ***"imbalanced data"***

DATOS DESBALANCEADOS

Técnicas de “balanceo”

1. Intenta juntar más datos.
2. No solo tomar a la exactitud y error como medida del modelo.
3. Remuestrear la data para equilibrar o lo que nos brinde mejores resultados (ojo con el muestreo)
4. Generar data sintética (SMOTE)
5. Probar diferentes algoritmos
6. Usar penalizaciones
7. Cambia el enfoque y es bueno ser creativo





K VECINO MÁS CERCANO - KNN

- Basado en la regla de bayes

La razón por la que el algoritmo Naive Bayes se llama Naive no es porque sea simple o ingenuo, es porque el algoritmo hace una suposición muy fuerte acerca de que los datos tienen características independientes entre sí, mientras que en realidad, pueden ser dependientes de alguna manera. En otras palabras, asume que la presencia de una característica en una clase no tiene relación alguna con la presencia de todas las demás características.

Si se cumple este supuesto de independencia, Naive Bayes se desempeña extremadamente bien y, a menudo, es mejor que otros modelos. Naive Bayes también se puede usar con características continuas, pero es más adecuado para variables categóricas. Si todas las características de entrada son categóricas, se recomienda Naive Bayes. Sin embargo, en el caso de las características numéricas, se hace otra suposición fuerte que es que la variable numérica se distribuye normalmente

NAIVE BAYES

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)}$$

donde:

- $P(A_i)$ son las probabilidades a priori.
- $P(B|A_i)$ es la probabilidad de B en la hipótesis A_i.
- $P(A_i|B)$ son las probabilidades a posteriori.

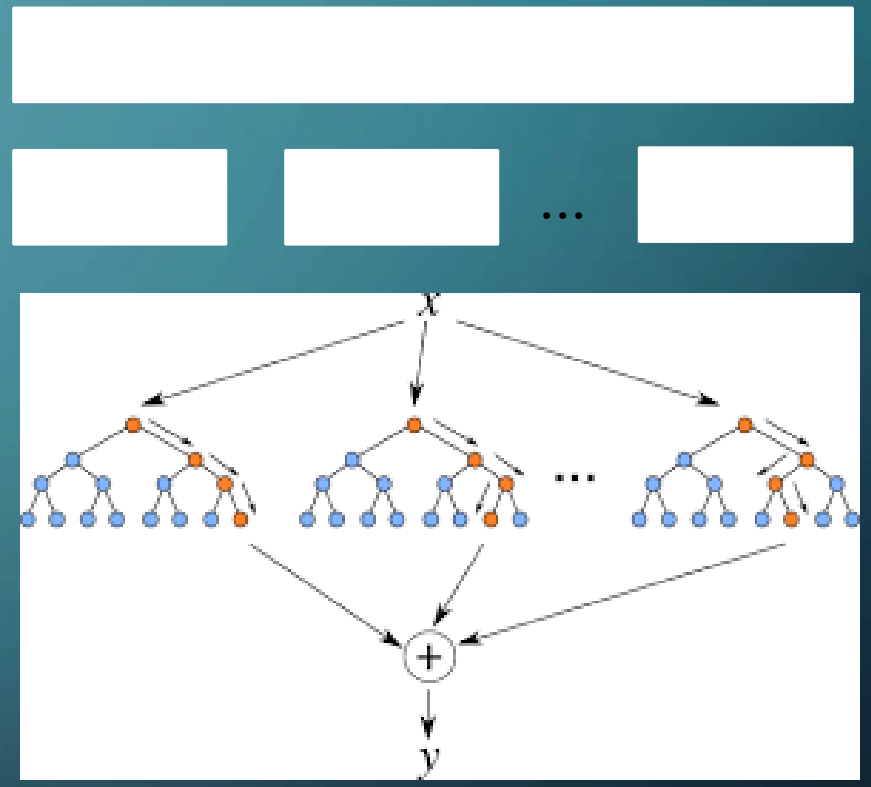
- Basado en la regla de bayes

La razón por la que el algoritmo Naive Bayes se llama Naive no es porque sea simple o ingenuo, es porque el algoritmo hace una suposición muy fuerte acerca de que los datos tienen características independientes entre sí, mientras que en realidad, pueden ser dependientes de alguna manera. En otras palabras, asume que la presencia de una característica en una clase no tiene relación alguna con la presencia de todas las demás características.

Si se cumple este supuesto de independencia, Naive Bayes se desempeña extremadamente bien y, a menudo, es mejor que otros modelos. Naive Bayes también se puede usar con características continuas, pero es más adecuado para variables categóricas. Si todas las características de entrada son categóricas, se recomienda Naive Bayes. Sin embargo, en el caso de las características numéricas, se hace otra suposición fuerte que es que la variable numérica se distribuye normalmente

BOSQUE ALEATORIO

- El algoritmo de un árbol por separado no puede producir modelos exactos, porque la variedad provoca una inestabilidad que se puede observar al crear árboles de decisiones por separado.
- El resultado de un bosque aleatorio es el promedio del resultado de un grupo de árboles de clasificación, donde los árboles son construidos con diferentes muestras del train y seleccionando en cada nodo un subconjunto de las variables predictoras. Con esto se consigue resultados más robustos al cambio de los datos y con el subconjunto de predictores en cada nodo conseguimos que se obtenga más información de todas las variables.
- Funciona tanto para clasificación como regresión.



RED NEURONAL

- Las Redes Neuronales Artificiales son modelos computacionales inspirados en las neuronas biológicas.
- La información de los predictores están en la capa de entrada.
- El target en la capa de salida.
- El modelo va aprendiendo de los datos de entrenamiento regulando los pesos dentro de la capa oculta.
- Entre los parámetros importantes están size (número de unidades ocultas intermedias) y decay (se usa para evitar el sobre ajuste)