

Contenido

[**1ER Clase – 10/06/2021 5**](#_heading=h.30j0zll)

[Temario 5](#_heading=h.1fob9te)

[*Evaluación de algoritmos 5*](#_heading=h.3znysh7)

[Aprendizaje automático Machine Learning 5](#_heading=h.2et92p0)

[*Aprendizaje automático supervisado 5*](#_heading=h.tyjcwt)

[*Aprendizaje automático no supervisado 6*](#_heading=h.3dy6vkm)

[*Aprendizaje por refuerzo (Reinforcement Learning) 6*](#_heading=h.1t3h5sf)

[Modelos predictivos 7](#_heading=h.4d34og8)

[¿Por qué necesitamos métricas de evaluación? 7](#_heading=h.2s8eyo1)

[Evaluación de Algoritmos 8](#_heading=h.17dp8vu)

[*Aprendizaje automático supervisado - Problemas de clasificación 8*](#_heading=h.3rdcrjn)

[Matriz de confusión 8](#_heading=h.26in1rg)

[Exactitud (*Accuracy*) 12](#_heading=h.lnxbz9)

[Exactitud(Accuracy)Clase no balanceada 13](#_heading=h.35nkun2)

[Quiz 14](#_heading=h.1ksv4uv)

[Tasa (RATE) de Verdaderos Positivos (Sensibilidad) TPR 15](#_heading=h.44sinio)

[Tasa de Falso Negativos FNR 15](#_heading=h.2jxsxqh)

[Tasa de Verdaderos Negativos (especificidad) TNR 16](#_heading=h.z337ya)

[Tasa de Falso Positivos FPR 16](#_heading=h.3j2qqm3)

[Alternativas a Exactitud 17](#_heading=h.1y810tw)

[Caso de estudio (Clase desbalanceada) 17](#_heading=h.4i7ojhp)

[Quiz 18](#_heading=h.2xcytpi)

[Precisión (Precision) 19](#_heading=h.1ci93xb)

[Precisión Malignos 19](#_heading=h.j4ehxuman0bk)

[Precisión: Benignos 19](#_heading=h.im6eb7aquvap)

[Caso de estudio (Clase desbalanceada) 19](#_heading=h.htp3gimldlep)

[Precisión Clase no balanceada 20](#_heading=h.xdlmt2zbqsqu)

[Sensibilidad (Recall) 21](#_heading=h.qi9jeir1hl0j)

[Sensibilidad Maligno (TPR) 21](#_heading=h.6xmbanm959y9)

[Sensibilidad Benigno (TNR) 21](#_heading=h.6u3cumn3tb5z)

[Caso de estudio (Clase desbalanceada) 21](#_heading=h.2uvvmdopfd88)

[Sensibilidad Clase no balanceada 22](#_heading=h.7ndz4h356k28)

[Quiz 23](#_heading=h.3whwml4)

[Puntaje F (F-Score) 24](#_heading=h.hxk45f38dcfd)

[“Trade off” Precisión vs. Sensibilidad 24](#_heading=h.2bn6wsx)

[F1 - Score 24](#_heading=h.qsh70q)

[Quiz 25](#_heading=h.3as4poj)

[Umbral (Threshold) 25](#_heading=h.1pxezwc)

[AUC –ROC 26](#_heading=h.49x2ik5)

[AUC (Area Under the Curve) 26](#_heading=h.2ixlvr66cr78)

[ROC (Receiver Operating Characteristic) 26](#_heading=h.2p2csry)

[AUC –ROC - Puntos para recordar: 27](#_heading=h.147n2zr)

[Notebook 27](#_heading=h.3o7alnk)

[**2da Clase – 17/06/2021 28**](#_heading=h.23ckvvd)

[Repaso de la 1er Clase 28](#_heading=h.ihv636)

[*Estandarización de los datos 28*](#_heading=h.32hioqz)

[Matriz de confusión 29](#_heading=h.4zm6mrnkmwx6)

[*AUC – ROC - Como se construye. Cálculo de TPR FPR 29*](#_heading=h.i45bbq8h2nrh)

[*AUC – ROC - Como se construye. Cálculo de TPR FPR (Otro ejemplo) 34*](#_heading=h.1hmsyys)

[*Quiz 36*](#_heading=h.41mghml)

[LogLoss 37](#_heading=h.xl8spqdpb10s)

[*PROBLEMAS con AUC - ROC 37*](#_heading=h.ypoc453w2yiw)

[*Comparo dos modelos 37*](#_heading=h.h7v25vecgcd9)

[*LogLoss. Concepto 38*](#_heading=h.2grqrue)

[*¿Cuándo se usa? 38*](#_heading=h.dg65vsl0spb3)

[*Advertencias de uso 38*](#_heading=h.y6a1x0uj5ydq)

[*¿Que son las probabilidades corregidas? 39*](#_heading=h.rr5mw96hfb6p)

[*Aplicamos Logaritmo 39*](#_heading=h.vx1227)

[*Quiz 42*](#_heading=h.3fwokq0)

[Coeficiente de Gini 43](#_heading=h.1v1yuxt)

[Gráficos de Ganancia (Gain) y Elevación (Lift) 43](#_heading=h.jvtp3aqx7ib3)

[*Gráfico de Ganancia (Gain) 43*](#_heading=h.rc7cr1lr0547)

[*Gráfico de Elevación (Lift) 45*](#_heading=h.uwaibrwpy7po)

[Diferencia entre Gain y ROC 45](#_heading=h.klz33oyyhzd3)

[Gráficos de Kolmogorov-Smirnov (K-S) 47](#_heading=h.4f1mdlm)

[Notebook 49](#_heading=h.2u6wntf)

[**3ta clase – 24/06/2021 50**](#_heading=h.19c6y18)

[AUC - ROC 50](#_heading=h.3tbugp1)

[Benchmarking y Línea de Base 51](#_heading=h.28h4qwu)

[*Benchmarking 51*](#_heading=h.nmf14n)

[*Línea Base 51*](#_heading=h.37m2jsg)

[Clasificador Tasa Cero ***ZeroR y*** Clasificador Aleatorio Ponderado 52](#_heading=h.1mrcu09)

[*Clasificador Tasa Cero 52*](#_heading=h.46r0co2)

[*Clasificador Aleatorio (Porcentaje Ponderado) 52*](#_heading=h.2lwamvv)

[Lanzamiento moneda 53](#_heading=h.111kx3o)

[Clase desbalanceada 53](#_heading=h.3l18frh)

[Clasificación Votación 54](#_heading=h.206ipza)

[Conclusiones 54](#_heading=h.4k668n3)

[Notebook 55](#_heading=h.2zbgiuw)

[Función de Perdida 55](#_heading=h.1egqt2p)

[Función de Perdida vs Métrica 56](#_heading=h.3ygebqi)

[Parámetros e Hiperparámetros 57](#_heading=h.2dlolyb)

[*Diferencia entre Parámetros e Hiperparámetros 57*](#_heading=h.sqyw64)

[Parámetros 57](#_heading=h.3cqmetx)

[Hiperparámetros 57](#_heading=h.1rvwp1q)

[Métodos de optimización Hiperparámetro 58](#_heading=h.4bvk7pj)

[Métricas Regresión 58](#_heading=h.2r0uhxc)

[*Aprendizaje automático supervisado – Problemas de regresión 58*](#_heading=h.1664s55)

[*¿Qué definimos como Error? 59*](#_heading=h.3q5sasy)

[*MAE: Error Medio Absoluto 60*](#_heading=h.25b2l0r)

[*MSE: Error Medio Cuadrado 60*](#_heading=h.kgcv8k)

[*RMSE: Error cuadrático Medio 61*](#_heading=h.34g0dwd)

[¿Por qué puede a veces no ser el mejor? 62](#_heading=h.1jlao46)

[*RMSLE: Error logarítmico cuadrático medio 62*](#_heading=h.43ky6rz)

[*RMSE: Error cuadrático Medio vs RMSLE: Error logarítmico cuadrático medio 63*](#_heading=h.2iq8gzs)

[*Quiz 64*](#_heading=h.xvir7l)

[*MSE Error cuadrático Medio Relativo 65*](#_heading=h.3hv69ve)

[*R –cuadrado R2 66*](#_heading=h.1x0gk37)

[*R2 y R2 ajustado 66*](#_heading=h.4h042r0)

[*Conclusiones 67*](#_heading=h.2w5ecyt)

[*Quiz 68*](#_heading=h.1baon6m)

[Tipos de Regresiones y símiles 69](#_heading=h.3vac5uf)

[Datasets webpages 70](#_heading=h.2afmg28)

[Notebook 70](#_heading=h.pkwqa1)

[**4ta clase – 01/07/2021 71**](#_heading=h.39kk8xu)

[Cross Validation y Overfitting 71](#_heading=h.bmyyy2lvao49)

[Sobreajuste y Subajuste 71](#_heading=h.1opuj5n)

[Sesgo y Varianza Error total 73](#_heading=h.48pi1tg)

[Sobreajuste Subajuste Sesgo Varianza 74](#_heading=h.2nusc19)

[Notebook 76](#_heading=h.1302m92)

[Competencia Kaggle 76](#_heading=h.3mzq4wv)

[Cross Validation 79](#_heading=h.2250f4o)

[*Cross Validation: distintos enfoques 79*](#_heading=h.haapch)

[*K-Fold Cross Validation 79*](#_heading=h.r3ieqobfgn04)

[*Confiuración K-Fold Cross Validation 80*](#_heading=h.8rx39zfk1o6n)

[*Notebook 81*](#_heading=h.319y80a)

[*Stratified K-fold Cross Validation 81*](#_heading=h.ycc9b0iokmax)

[*Cross Validation series de tiempo 81*](#_heading=h.1gf8i83)

[*Notebooks 82*](#_heading=h.40ew0vw)

[**5ta Clase – 08/07/2021 83**](#_heading=h.2fk6b3p)

[Indicadores clave de rendimiento o key performance indicators (KPI) 83](#_heading=h.upglbi)

[*Indicadores KPI 83*](#_heading=h.3ep43zb)

[*Relación entre KPI y métricas 83*](#_heading=h.1tuee74)

[Ejemplo 84](#_heading=h.4du1wux)

[*Diferencia entre Ingresos y Ganancia 86*](#_heading=h.2szc72q)

[*Optimización 87*](#_heading=h.184mhaj)

[*Notebook 88*](#_heading=h.3s49zyc)

[Análisis de churn (KPI) 89](#_heading=h.279ka65)

[*Churn o cancelación 89*](#_heading=h.meukdy)

[*Churn Machine learning 89*](#_heading=h.36ei31r)

[*Churn WorkFlow 90*](#_heading=h.1ljsd9k)

[*Notebook 91*](#_heading=h.45jfvxd)

[Apéndice 91](#_heading=h.2koq656)

[*Parte del contenido de: www.r2d3.us 91*](#_heading=h.zu0gcz)

[APRENDIZAJE AUTOMÁTICO-PARTE I 91](#_heading=h.3jtnz0s)

[MACHINE LEARNING 91](#_heading=h.1yyy98l)

[FORK: 92](#_heading=h.4iylrwe)

[TRADE-OFFS (Contrapartidas) 92](#_heading=h.2y3w247)

[The Best Split (La major division) 92](#_heading=h.1d96cc0)

[Recursión 92](#_heading=h.3x8tuzt)

[Crecimiento de un árbol 93](#_heading=h.2ce457m)

[Hacer predicciones 94](#_heading=h.rjefff)

[Comprobación de la realidad 94](#_heading=h.3bj1y38)

[Recapitulación 94](#_heading=h.1qoc8b1)

[APRENDIZAJE AUTOMÁTICO-PARTE II 95](#_heading=h.4anzqyu)

[Parámetros del modelo 95](#_heading=h.2pta16n)

[Demasiado simple 95](#_heading=h.14ykbeg)

[Demasiado complejo 96](#_heading=h.3oy7u29)

[Ejemplo tangible de la varianza 97](#_heading=h.243i4a2)

[Las fallas son normales 98](#_heading=h.j8sehv)

[El equilibrio entre el sesgo y la variación 99](#_heading=h.338fx5o)

[La relación en términos abstractos 100](#_heading=h.1idq7dh)

[Conclusión final 102](#_heading=h.42ddq1a)

[Recapitulación 102](#_heading=h.2hio093)

# 1ER Clase – 10/06/2021

## Temario

### Evaluación de algoritmos

Hay una parte teórica y otra parte práctica

* Métricas de Clasificación ( precision, recall, F 1 score, ROC, AUC,
* Métricas de Regresión ( MAE)
* Evaluaciones para conjuntos de Rankings ( DCG, NDCG)
* Métricas estadísticas (
* Medida de calidad de vídeo ( SSIM, IoU
* Métricas PLN Perplexity BLEU score)
* Métricas Deep Learning Inception score, Frechet Inception distance)
* Métricas Algoritmos Descriptivos (Silhouette Indice de Dunn)

## Aprendizaje automático Machine Learning

Aprendizaje automático supervisado

* Problemas de clasificación
* Problema de regresión

Aprendizaje automático no supervisado

Aprendizaje por refuerzo Reinforcement Learning

### Aprendizaje automático supervisado

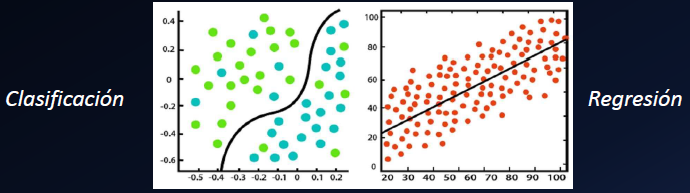
Cuando se poseen datos que han sido clasificados (etiquetas en la terminología del aprendizaje automático) y se desea predecir los resultados para a futuro

**Problemas de clasificación**

Cuando se desea clasificar los resultados en diferentes clases Por ejemplo, si un cliente incumpliría con su préstamo o no es un problema de clasificación que es de gran interés para cualquier banco El resultado puede caer en una de las clases Sí o No

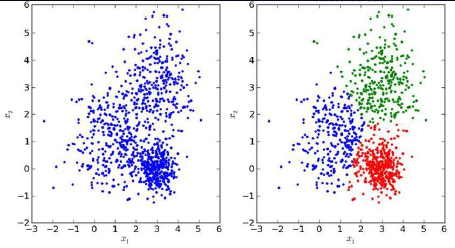
**Problema de regresión**

Cuando uno está interesado en responder “estos problemas caerían bajo el paraguas de regresión Por ejemplo, cuál es la cantidad esperada de incumplimiento de un cliente es un problema de regresión



### Aprendizaje automático no supervisado

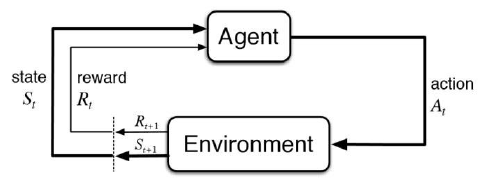
Se desea realizar una segmentación o agrupación Por ejemplo, un banco querría tener una segmentación de sus clientes para comprender su comportamiento Este es un problema de aprendizaje automático no supervisado, ya que no estamos prediciendo ningún resultado aquí.



### Aprendizaje por refuerzo (Reinforcement Learning)

Es un área del aprendizaje automático que se ocupa de cómo los agentes inteligentes deben realizar acciones en un entorno para maximizar la noción de recompensa acumulativa

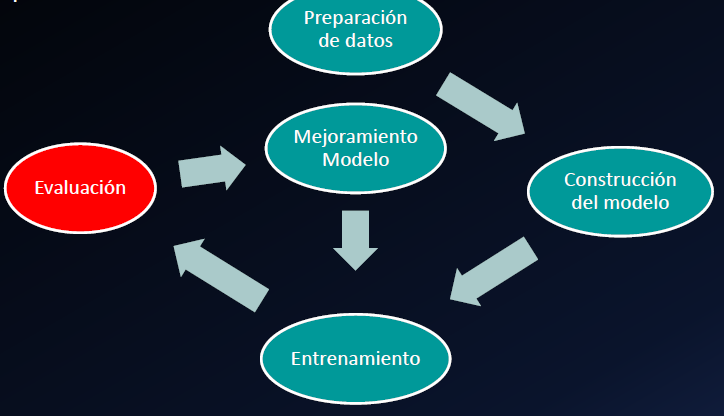
Algunos ejemplos automóviles sin conductor, las aspiradoras con navegación automática y la programación de ascensores



## Modelos predictivos

La idea de crear modelos de aprendizaje automático se basa en un principio de retroalimentación constructiva

Construir, obtener métricas, realiza mejoras y continuar hasta lograr la precisión deseada



## ¿Por qué necesitamos métricas de evaluación?

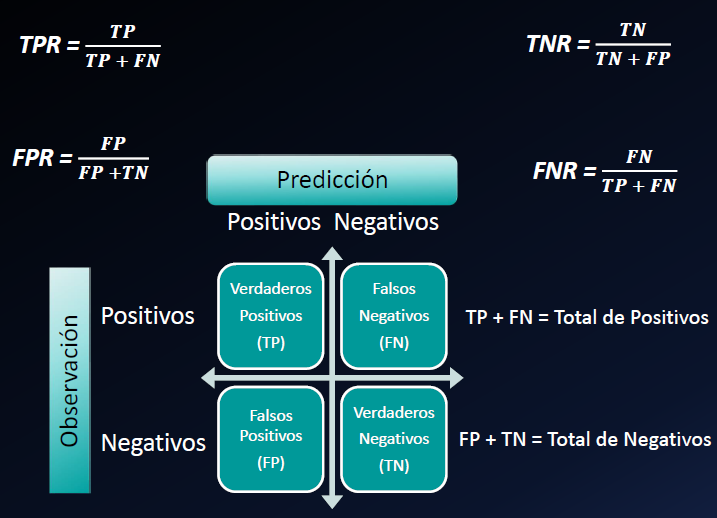
* Las métrica de evaluación explican el rendimiento de un modelo .Un aspecto importante de las métricas de evaluación es su capacidad para discriminar entre los resultados del modelo
* Generalizar sobre los datos que no se usaron para el entrenamiento
* Utilizar diferentes métricas permite mejorar el poder predictivo general de nuestro modelo antes de implementarlo
* Hay que tener cuidado que nuestro modelos no aprendan y no memoricen!!!. Esto se refiere al problema de Overfiting. Es algo que no se si vieron en la otra materia pero para resumir, es cuando el algoritmo ajusta demasiado a los datos de entrenamiento y pierde capacidad de predecir un nuevo registro que nunca vio.

## Evaluación de Algoritmos

### Aprendizaje automático supervisado - Problemas de clasificación

#### Matriz de confusión

Nota: “Positivo” o “Negativo” se refiere a nuestro problema en particular y lo podemos definir arbitrariamente.



En el caso del ejemplo Verdaderos positivos: son: las imágenes que en su medición dieron que SI son tumores malignos y que realmente SI lo son. Verdaderos Negativos: son: las imágenes que en su medición dieron que NO son tumores malignos y que realmente NO lo son. Es la diagonal principal de la matriz.

Falsos Positivos: son: las imágenes que en su medición dieron que SI son tumores malignos pero que realmente NO lo son. Falsos Negativos: son: las imágenes que en su medición dieron que NO son tumores malignos pero que realmente SI lo son. Es la diagonal secundaria de la matriz.

###### Notación

“Positivo” o “Negativo” se refiere a nuestro problema en particular y lo podemos definir arbitrariamente y en general se asigna “1” al positivo y“0” al negativo

Verdaderos Positivos **VP** =True Positives **TP**

Falsos Positivos **FP** =False Positives **FP**

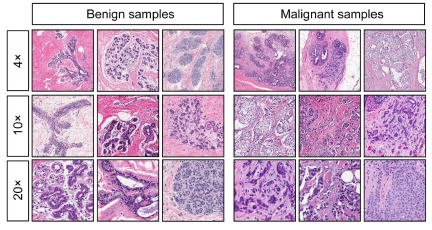
Verdaderos Negativos **VN** =True Negatives **TN**

Falsos Negativos **FN** =False Negatives **FN**

###### Ejemplo Cáncer de Mama Universidad de Wisconsin

https:://archive ics uci edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

Las características se calculan a partir de una imagen digitalizada de una aspiración con aguja fina de una masa mamaria Describen características de los núcleos celulares presentes en la imagen



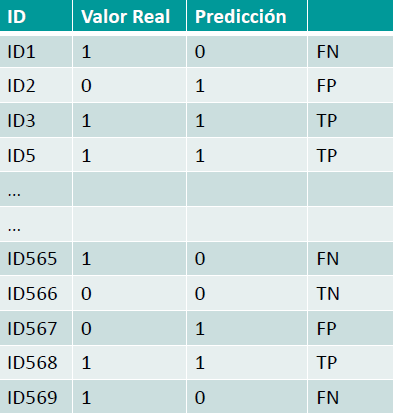
**Información**

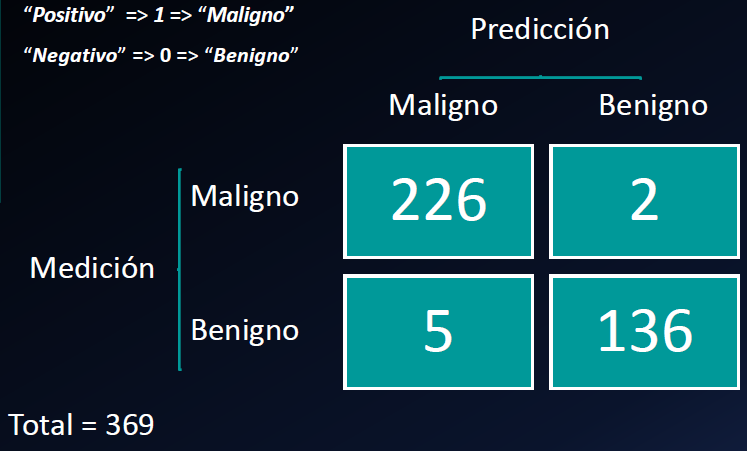
1. ID
2. Diagnosis (M Maligno, B Benigno)Benigno)(tomaremos positivo M y negativo B)

10 Features que se obtienen a partir del análisis de los núcleos

1. Radio (media de las distancias desde el centro a los puntos del perímetro)
2. Textura (desviación estándar de los valores de la escala de grises)
3. Perímetro
4. Área
5. Suavidad (variación local en longitudes de radio)
6. Compacidad ( 2 /área 1 0
7. Concavidad (severidad de las porciones cóncavas del contorno)
8. Puntos cóncavos (número de porciones cóncavas del contorno)
9. Simetría
10. Dimensión fractal ("aproximación de la línea de costa" 1

**Aplico un clasificador por ejemplo un Árbol**





###### Quiz

La matriz de confusión se utiliza para problemas de regresión. ¿Verdadero o Falso?

Suponga que está trabajando en un problema de clasificación donde las etiquetas de destino son 0 (negativo) y

1 (positivo). ¿Cuál de las siguientes opciones describe falsos positivos en este caso?

A) Valores que en realidad son 1 y predijimos 1

B) Valores que en realidad son 0 y predijimos 0

C) Valores que en realidad son 0 y predijimos 1

D) Valores que en realidad son 1 y predijimos 0

El número total de positivos reales se da como?

A) TP + FP

B) TP + FN

C) FP + FN

D) FN + TN

Si hacemos una matriz de confusión N por N, ¿qué representa N aquí?

A) Número de observaciones en los datos

B) Número de características en los datos

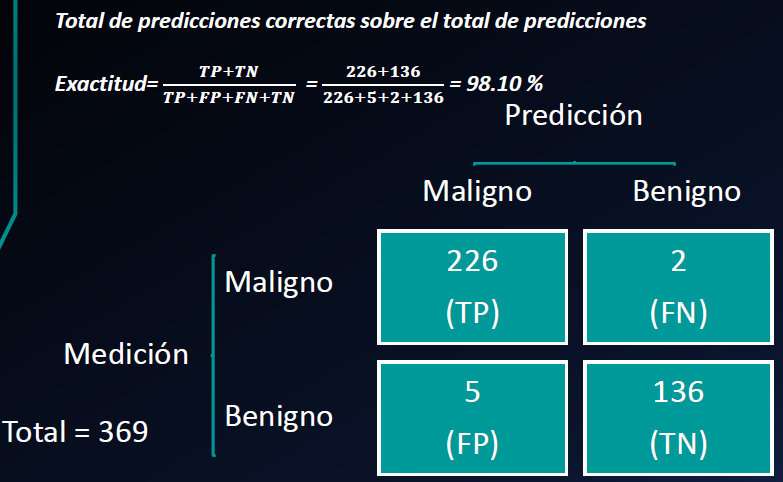
C) Número de clases en la variable dependiente/objetivo

D) Ninguna de las anteriores

RTA: Falso, c), b), c)

##### Exactitud (*Accuracy*)

Total de predicciones correctas sobre el total de predicciones.



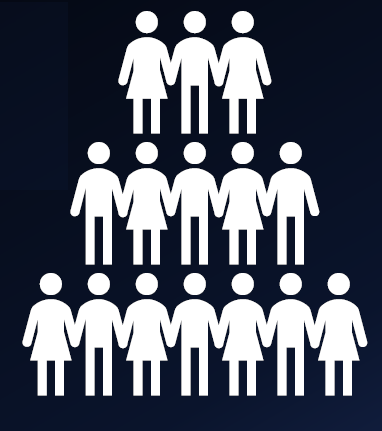
##### Exactitud(Accuracy)Clase no balanceada

###### Nuevo caso de estudio (Clase desbalanceada)

Población de estudio = población total (500)

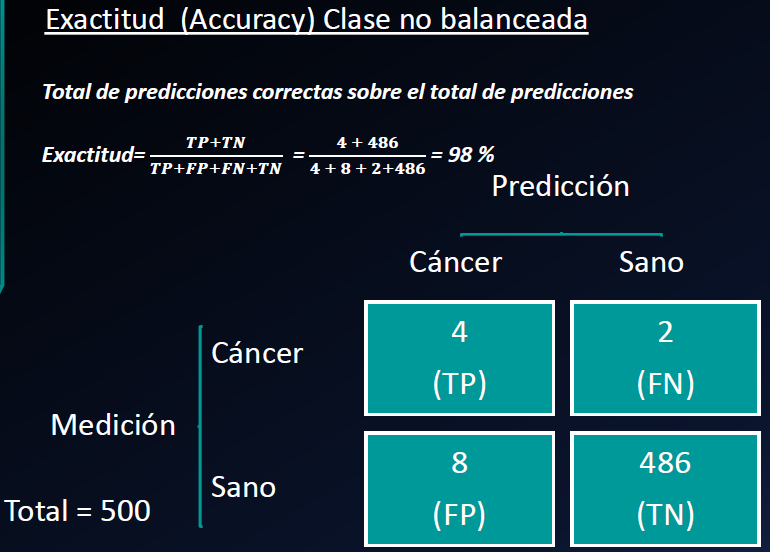
Positivos = Paciente Cáncer 1

Negativo = Paciente Sano 0



Modelo que predice siempre **Negativo**

Total de predicciones correctas sobre el total de predicciones



##### Quiz

¿Qué fórmula de las siguientes opciones es la correcta para calcular la Exactitud ?

A) 

B) 

C) 

D) 

Debe predecir si un paciente tiene cáncer o no. En el conjunto de entrenamiento, la distribución de clases de la variable objetivo es la siguiente:

No --99%

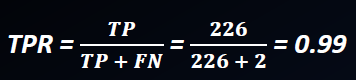
Si .- 1%

En esta situación, la precisión será una mejor opción para evaluar el modelo. ¿Verdadero o Falso?

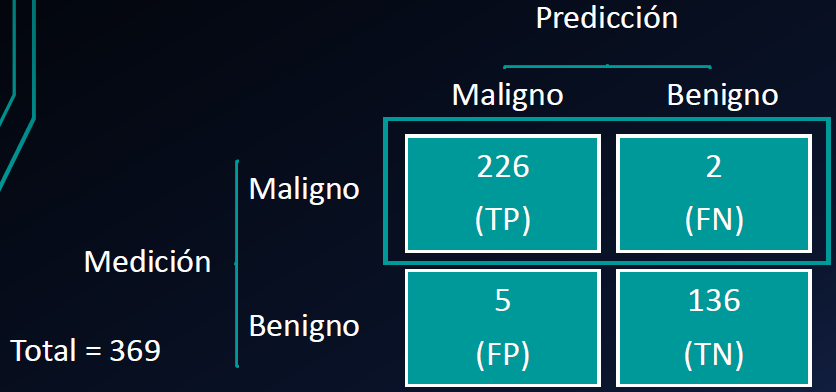
Aclaración: En ese caso debe decir Accuracy o Exactitud y no precisión, tal vez eso es lo que te confunde. Recordá que Accuracy para clase desbalanceada no sirve de mucho.

RTA: b), Falso

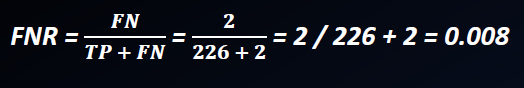
##### Tasa (RATE) de Verdaderos Positivos (Sensibilidad) TPR

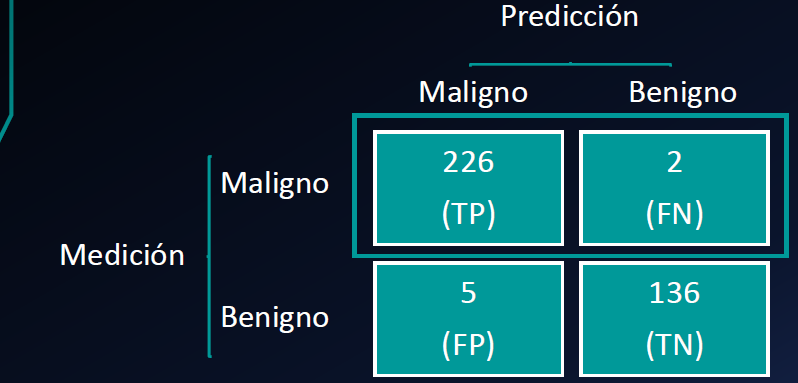


Se refiere al hecho que uno debe mirar con más atención los valores de Precisión (Precision) cuando está frente a un problema donde es muy importante tener pocos falsos positivos. Ejemplo:



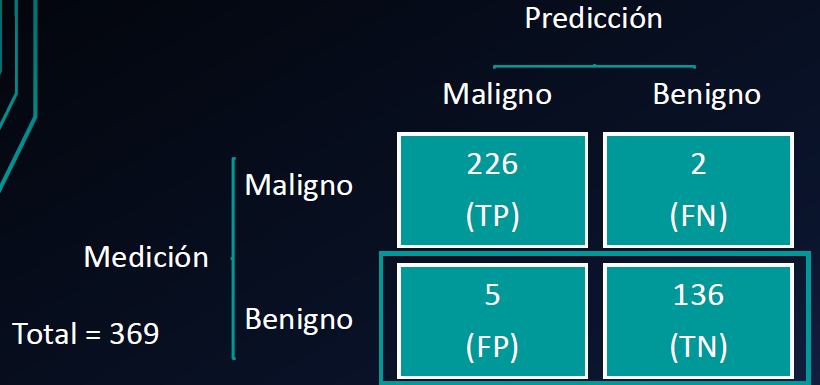
##### Tasa de Falso Negativos FNR



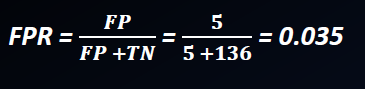


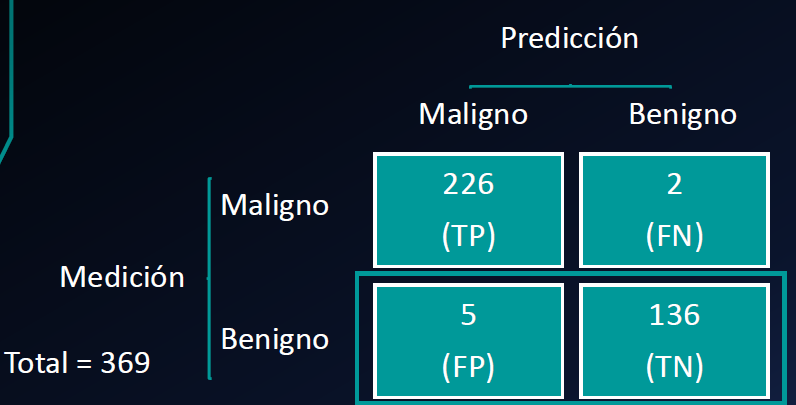
##### Tasa de Verdaderos Negativos (especificidad) TNR



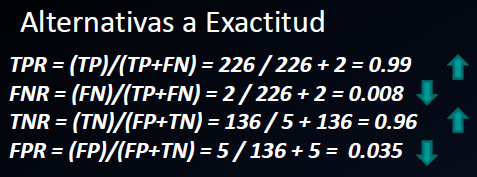


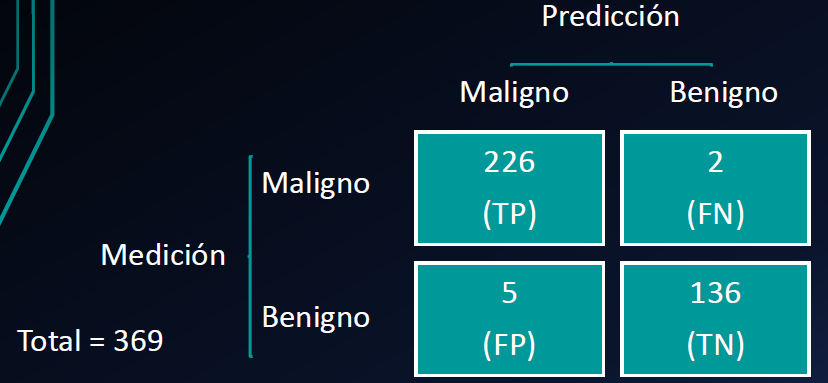
##### Tasa de Falso Positivos FPR





##### Alternativas a Exactitud





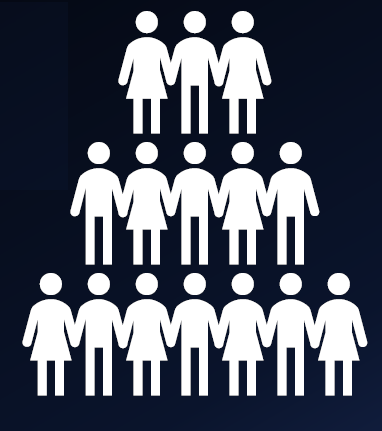
##### Caso de estudio (Clase desbalanceada)

**Modelo que predice siempre Negativo. Exactitud 98.8%!!!**

Población de estudio = población total (500)

Positivos = Paciente Cáncer 1

Negativo = Paciente Sano 0

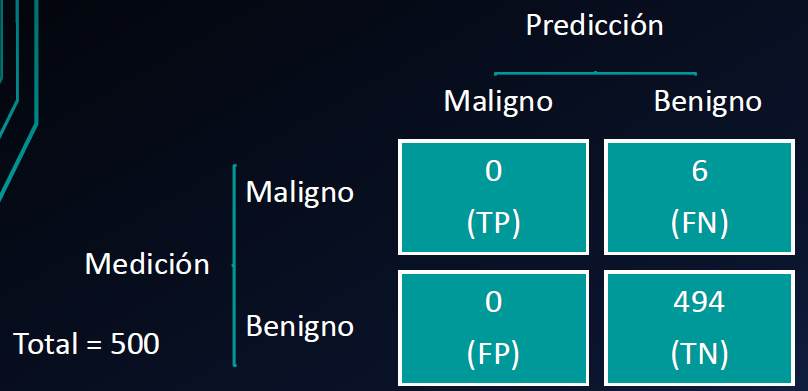


TPR = (TP)/(TP+FN) = 0 / 0 + 6 = 0

FNR = (FN)/(TP+FN) = 6 / 6 + 0 = 1 todos los pacientes clasificados como no malignos!!!

TNR = (TN)/(FP+TN) = 494 / 0 + 494 = 1 identifico bien a los benignos

FPR = (FP)/(FP+TN) = 0 / 0 + 494 = 0 No se equivocó con los benignos



##### Quiz

¿Cuál de las siguientes opciones define correctamente la tasa de verdaderos positivos?

A) Positivos correctamente clasificados de todos los positivos

B) Negativos clasificados correctamente de todos los negativos

¿Cómo decidimos qué métrica usar para evaluar el modelo?

A) Depende del tamaño del conjunto de datos

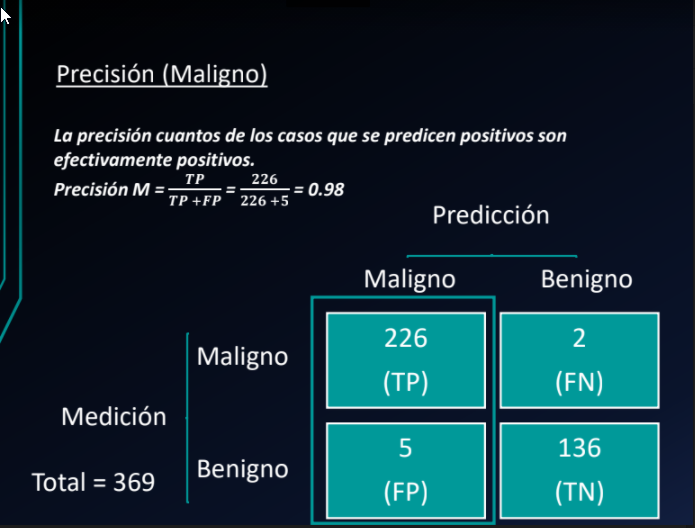
B) Depende del problema comercial

C) Podemos elegir cualquier métrica para evaluar el modelo.

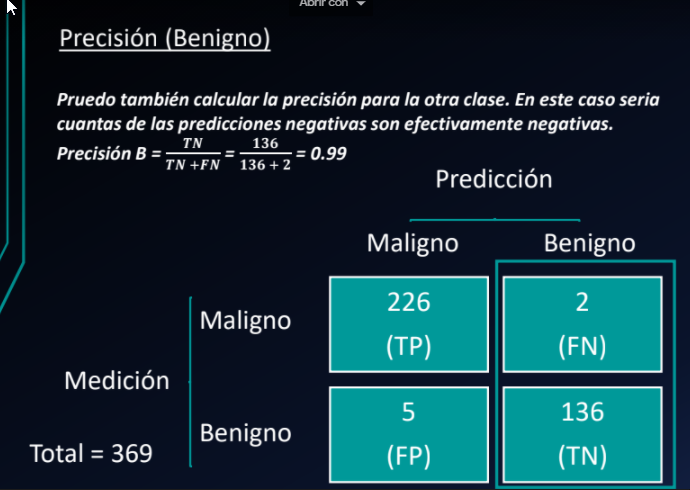
RTA: a); b)

#### Precisión (Precision)

##### Precisión Malignos



##### Precisión: Benignos

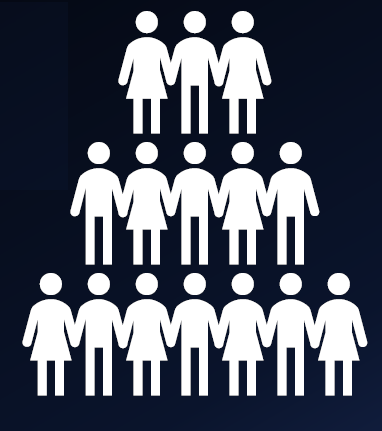


##### Caso de estudio (Clase desbalanceada)

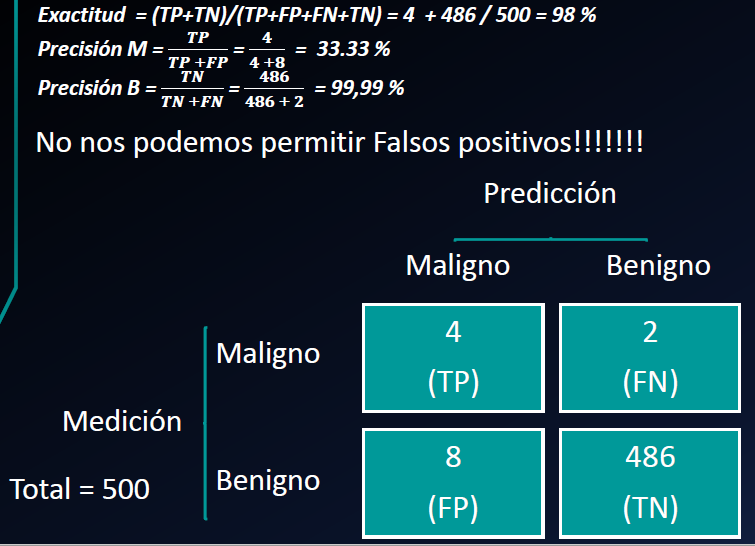
Población de estudio = población total (500)

Positivos = Paciente Cáncer 1

Negativo = Paciente Sano 0



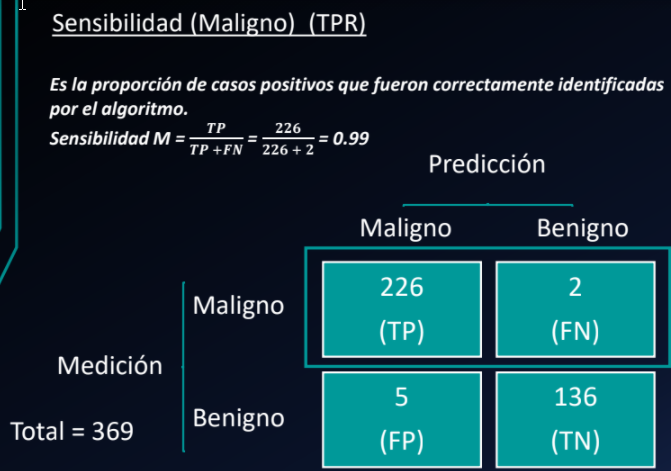
##### Precisión Clase no balanceada



Se refiere al hecho que uno debe mirar con más atención los valores de Precisión (Precision) cuando está frente a un problema donde es muy importante tener pocos falsos positivos. Ejemplo: correos spam.

#### Sensibilidad (Recall)

##### Sensibilidad Maligno (TPR)



##### Sensibilidad Benigno (TNR)

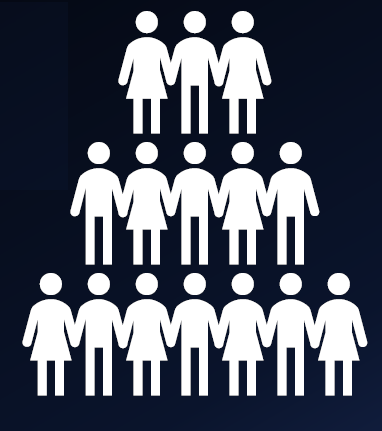


##### Caso de estudio (Clase desbalanceada)

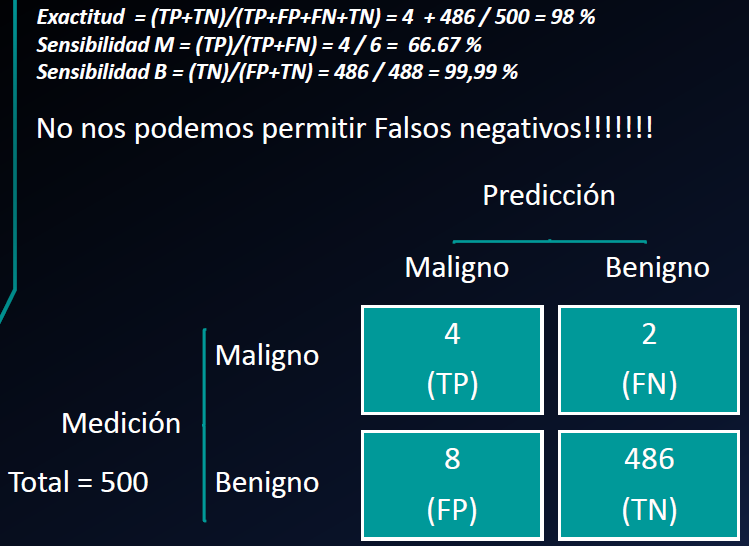
Población de estudio = población total (500)

Positivos = Paciente Cáncer 1

Negativo = Paciente Sano 0



##### Sensibilidad Clase no balanceada



Se refiere al hecho que uno debe mirar con más atención los valores de Sensibilidad (Recall) cuando está frente a un problema donde es muy importante tener pocos falsos negativos. Ejemplo: tumores.

##### Quiz

1. En el caso de Precisión, ¿qué priorizamos en minimizar?

A) Verdadero negativo

B) Falso negativo

C) Verdadero Positivo

D) Falso positivo

1. En el caso de Sensibilidad (Recall), ¿qué priorizamos en minimizar?

A) Verdadero negativo

B) Falso negativo

C) Verdadero Positivo

D) Falso positivo

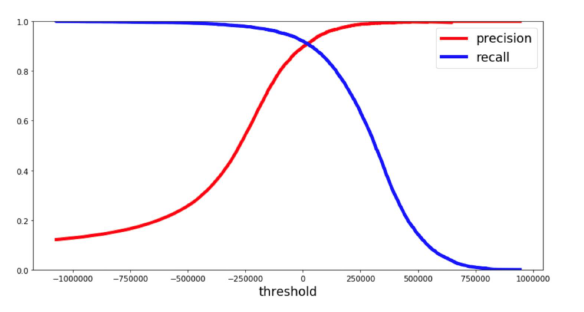
RTA: D), B)

#### Puntaje F (F-Score)

##### “Trade off” Precisión vs. Sensibilidad

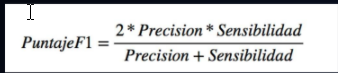
Trade off: Solución de compromiso

Mario: Trade off es un término que se usa muy a menudo y deriva del inglés. Literalmente significa intercambio, pero se usa como término que indica la relación costo beneficio. En particular en esta slide quiere decir que nos va indicando la tasa con la cual identificamos verdaderos positivos con respecto a los falsos positivos. Es decir evalúa cuan bueno es el modelo en todo el rango de poder distinguir entre verdaderos positivos y falsos positivos.



##### F1 - Score

* ¿Qué pasa si para un caso de uso, estamos tratando de obtener la mejor precisión y sensibilidad al mismo tiempo? F1-Score es la media armónica de precisión y valores de recuperación para un problema de clasificación
* La fórmula para F1-Score es la siguiente:



* Ahora por qué están tomando una media armónica y no una media aritmética. Esto se debe a que la media armónica (MA) castiga más los valores extremos. Entendamos esto con un ejemplo
* Tenemos un modelo de clasificación binaria con los siguientes resultados:

**Precisión: 0, Sensibilidad (Recall): 1**

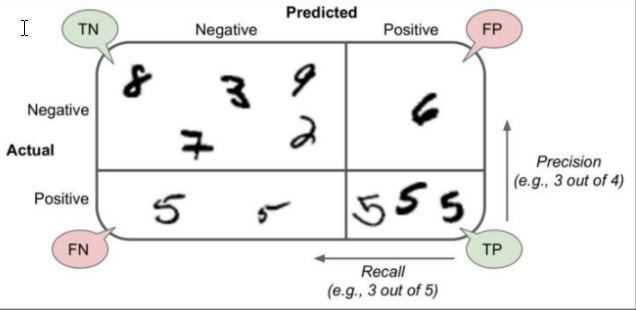
* Si tomamos la media aritmética, obtenemos 0.5. Está claro que el resultado anterior proviene de un clasificador tonto que simplemente ignora la entrada y sólo predice una de las clases como salida. Ahora, si tuviéramos que tomar MA, obtendríamos 0 que es exacto ya que este modelo es inútil para todos los propósitos.
* Esto parece sencillo. Sin embargo, hay situaciones en las que queremos dar un porcentaje más de importancia a la precisión o a la sensibilidad. Alterando un poco la expresión anterior de modo que podamos incluir un parámetro beta ajustable para este propósito, obtenemos:



* F-beta mide la efectividad de un modelo con respecto a un usuario que otorga β veces más importancia al recuerdo que a la precisión.

La idea es equilibrar la precisión y la sensibilidad y la media armónica asegura que uno no puede obtener un F1-score alto si cualquiera de las dos es muy baja (ver ejemplo para entender).

La introducción del parámetro beta simplemente permite ponderar en modo distinto la precisión con respecto a la sensibilidad.



##### Quiz

1. ¿Cuándo es la puntuación máxima de F1?

A) Precisión = 1, Sensibilidad = 0

B) Sensibilidad = 1. Precisión = 0

C) Precisión = Sensibilidad

D) No puede ser determinado

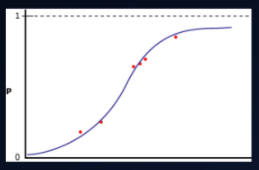
Aquí hubo un problema en la clase lo dije bien, pero luego marqué mal la opción.

La opción correcta es la C no la B. Esto es porque a medida que baja una sube la otra y matemáticamente se deduce que el punto máximo es cuando ambas son iguales. Es por eso que el F1-score es mejor para estimar ambas Precision y Recall juntas.

RTA: C)

#### Umbral (Threshold)

* El modelado predictivo de clasificación normalmente implica la predicción de una etiqueta de clase.
* Sin embargo, muchos algoritmos de aprendizaje automático son capaces de predecir una probabilidad o puntuación de pertenencia a una clase, y esto debe interpretarse antes de que pueda asignarse a una etiqueta de clase nítida.
* Esto se logra utilizando un umbral, como 0.5, donde todos los valores iguales o mayores que el umbral se asignan a una clase y todos los demás valores se asignan a otra clase.



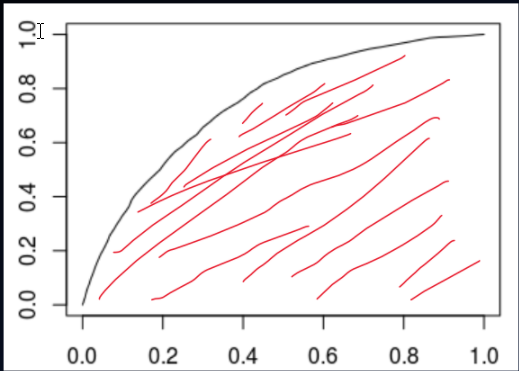
#### AUC –ROC

Hay algún modelo para el cual no seA bueno usar la curva ROC?

En general funciona bien para todos los modelos pero cuando la clase esta muy desbalanceada hay que tener cuidado porque pocos registros de la clase pequeña (bien o mal clasificados) tienen un impacto grande. Por eso para clases desbalanceadas no es lo mejor pero como ya les dije en general el tema de la clase no balanceada se trata de otro modo

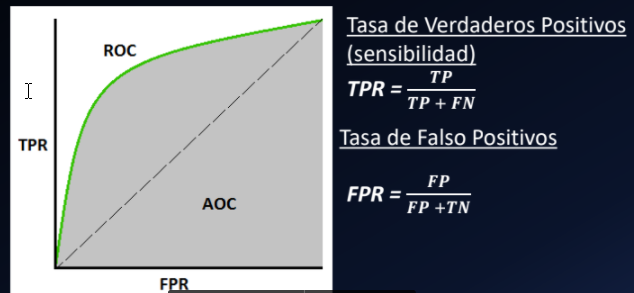
##### AUC (Area Under the Curve)

Es el área bajo la curva. Se calcula con una integral. La métrica más usada en algoritmos de clasificación es la curva ROC



##### ROC (Receiver Operating Characteristic)

* El nombre deriva de teoría de detección de señales y esta relacionado con la posibilidad de distinguir la señal del “ruido”.
* Ahora se lo usa para evaluar clasificadores binarios en Machine Learning
* Nos da el “trade off” de los falsos positivos y los falsos negativos

**

##### AUC –ROC - Puntos para recordar:

* Para un modelo que sólo devuelve la clase como salida, se representará como un solo punto en la gráfica ROC.
* Dichos modelos no se pueden comparar entre sí, ya que el juicio debe tomarse en una única métrica y no en múltiples métricas.
* En el caso del modelo probabilístico se puede obtener un único valor definido como el AUC-ROC. Pero aun así, debemos mirar toda la curva para tomar decisiones concluyentes. También es posible que un modelo funcione mejor en una región y otro funcione mejor en otra
* La curva ROC es casi independiente de la tasa de respuesta. Esto se debe a que tiene los dos ejes que salen de cálculos en columnas de la matriz de confusión. El numerador y el denominador de los ejes X e Y cambiarán en una escala similar en caso de cambio en la tasa de respuesta.

Si tenes un modelo que sólo devuelve la clase no se puede construir una curva ROC porque eso representaría un único punto. Recordá que cada punto de la curva ROC se construye a partir de calcular la FPR y TPR. Si mi modelo me devuelve la clase y no la probabilidad de pertenecer a la clase no puedo aplicar un umbral y por ende solo tendré un punto. Por el contrario, si el modelo me devuelve una probabilidad de pertenecer a una clase, puede usarse cada uno de los valores obtenidos como umbral y para cada uno de ellos calcular un FPR y TPR a partir de los cuales después construiré la curva ROC. Esto me permite evaluar el modelo en todo su rango.

#### Notebook

1er Clase - CANCER\_MATRIZ\_ROC-checkpoint

# 2da Clase – 17/06/2021

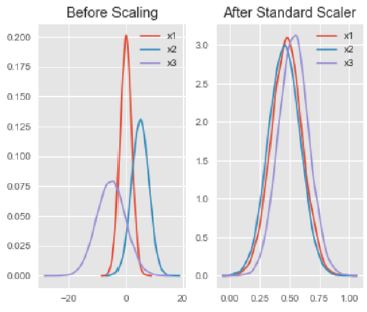
## Repaso de la 1er Clase

### Estandarización de los datos

Referencia del notebook" CANCER\_MATRIZ\_ROC-checkpoint". (está aquí y en el notebook porque en el notebook no puedo pegar el gráfico

Pregunta de Romina: En que escala están los datos?, cuan Mdesbalanceados deben estar los datos para decidir aplicar StandardScaler

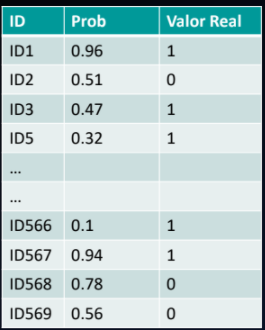
Respues de Mario: z = (x - u) / s se calcula restando el promedio de la población y dividiendo por el desvío estándar de cada variable. Esto lo que te muestra es como los valores varían según el promedio y variación de cada variable y por lo tanto todos quedan en escalas comparables.



## Matriz de confusión

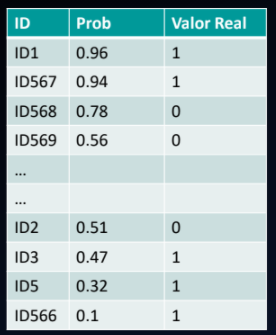


### AUC – ROC - Como se construye. Cálculo de TPR FPR

****

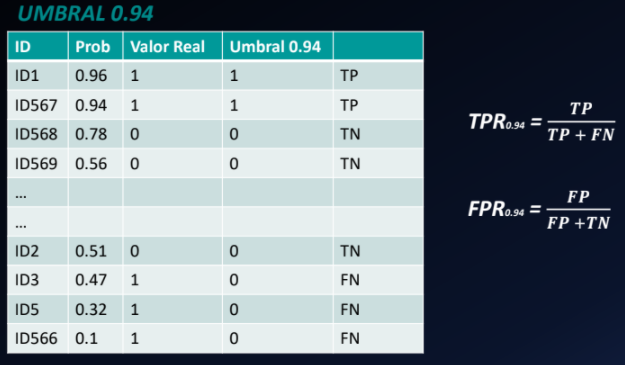
Justamente en este caso tenemos un modelo que en vez de devolverme la clase me devuelve la Probabilidad de pertenecer a una clase. Existen distintos modelos clasificadores que pueden hacer eso y depende del funcionamiento de cada uno. Esto tiene que ver con el modelo utilizado no con la métrica. Si tu modelo no puede devolver probabilidades no se puede construir la curva ROC. Hoy casi todos los modelos están diseñados para devolver probabilidades pero es importante tener en claro esto porque si no es difícil entender cómo se construye una curva ROC a partir de un clasificador.

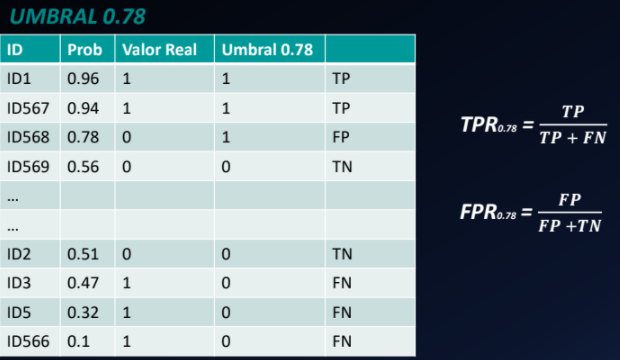
**Primero se ordena por probabilidad**

****

**Luego se calcula la TPR y FPR para cada Umbral**

****

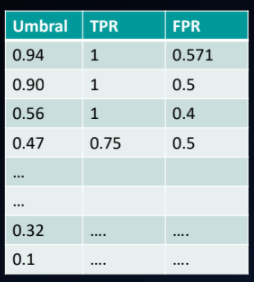
****

****

****

**Obtengo los Valores de TPR y FPR para todas las probabilidades y así construyo la curva**

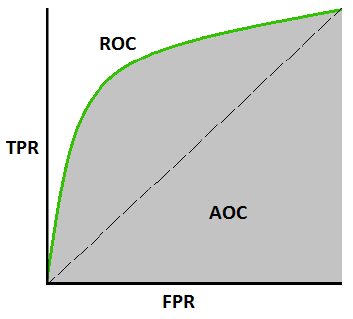
Con todo eso construyo la curva ROC.

****

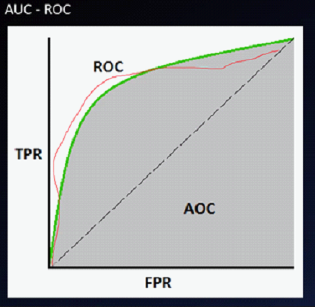
Se muestran los valores de FPR y TPR calculados para cada umbral. Es decir esos valores salen de hacer el cálculo a partir de las cuatro matrices anteriores.

Observar que el número es un número calculado en todo el Dataset y no en las pocas líneas que se visualizan en cada matriz.

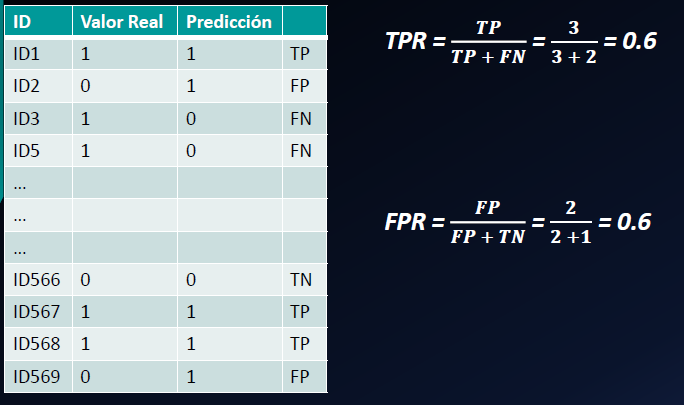
La curva ROC, solo considera el orden de las probabilidades.



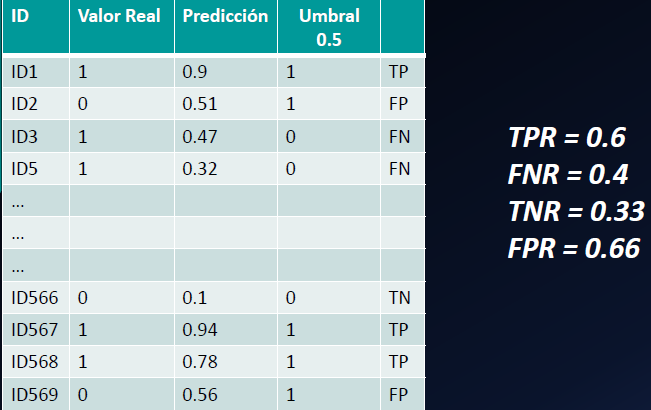
Quizás estas dos curvas tienen la misma área y no sé cuál sería mejor para mi caso.



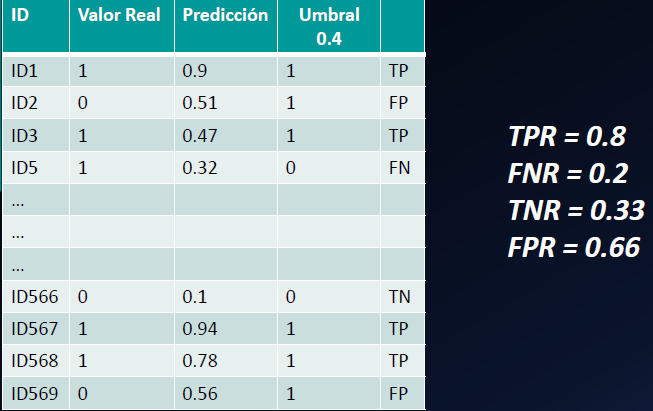
### AUC – ROC - Como se construye. Cálculo de TPR FPR (Otro ejemplo)



**Umbral 0,5**



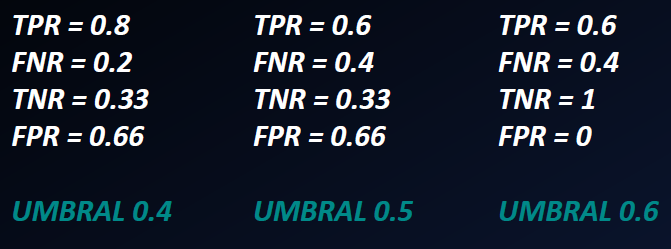
**Umbral 0,4**



**Umbral 0,6**



**Cálculo para los tres umbrales**



### Quiz

1. La curva AUC-ROC se traza entre?

A) Tasa de verdaderos positivos y tasa de verdaderos negativos

B) Tasa de falsos positivos y tasa de falsos negativos

C) Tasa de falsos positivos y tasa de verdaderos positivos

D) Tasa de verdaderos positivos y tasa de falsos negativos

1. AUC ROC no se puede utilizar en caso de distribución sesgada de clases. ¿Verdadero o Falso? (CLASE 2)
2. ¿Cuál de las siguientes afirmaciones es correcta?

A) TPR debe ser mayor que FPR

B) FPR debe ser mayor que TPR

En el caso que los valores de FPR son mayores que TPR significa que el clasificador es peor que el azar. Quiere decir que siempre predice mal. En general es indicativo de algún problema en el algoritmo en que uno identificó mal la clase o en la elección de los datos de entrenamiento y testeo.

RTA: C), Verdadero, A)

## LogLoss

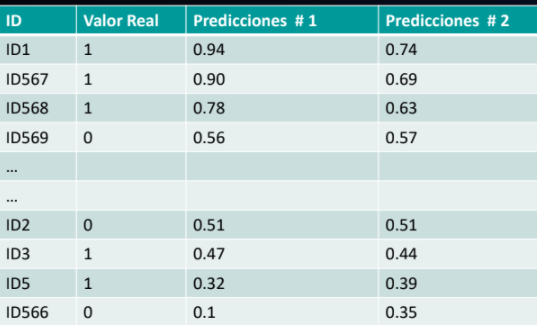
Ahora vamos a ver otra métrica

### PROBLEMAS con AUC - ROC

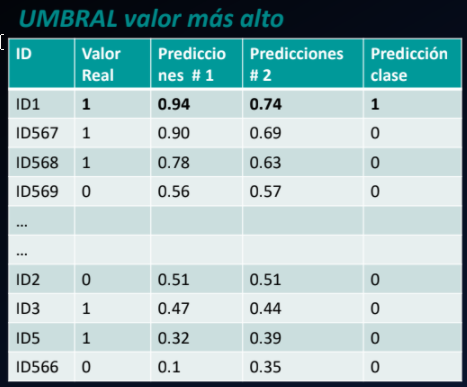
Considera sólo el orden de las probabilidades

No puede ser utilizada para comparar dos modelos

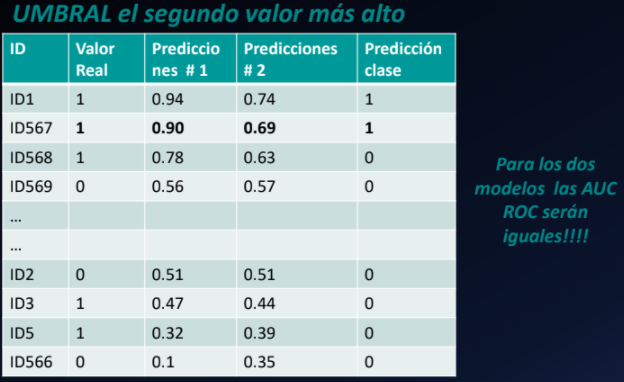
### Comparo dos modelos

****

**Umbral más alto**

****

**Umbral, el segundo valor más alto**

****

Es muy difícil que esto pase, pero podría pasar. Ojo porque esto está midiendo solamente cómo está ordenado. Las dos predicciones dan la misma curva.

### LogLoss. Concepto

* El promedio negativo de los logaritmos de las probabilidades corregidas para cada fila.
* LogLoss es una métrica de evaluación bastante buena para los clasificadores binarios y, a veces, también se utiliza en el caso de regresión logística y redes neuronales.

### ¿Cuándo se usa?

* Cuando la salida de un clasificador son probabilidades de predicción. **LogLoss** tiene en cuenta la incertidumbre de su predicción basada en cuánto varía de la etiqueta real. Esto nos da una visión más matizada del desempeño de nuestro modelo. En general, minimizar la pérdida de registros proporciona una mayor precisión para el clasificador.

Duda Romina: En esta página, a que se refiere con “minimizar la pérdida de registros”? Rta. Mario: “pérdida de registros” es el modo literal de traducir LogLoss, luego me di cuenta que a veces es mejor usar el término inglés para no generar confusión

### Advertencias de uso

* Es susceptible en caso de conjuntos de datos desbalanceados. Es posible que deba introducir pesos de clase para penalizar más los errores de las minorías o puede usar esto después de equilibrar su conjunto de datos
* Dudas Romina: Esto quiere decir que le aplico un peso mayor a los valores de la clase minoritaria, por ejemplo a la clase minoritario le aplico un peso de 0,7 y a la clase mayoritaria le aplico un peso de 0,3? **Rta. Mario**: Esto es correcto. En general no es tan fácil pensar cómo utilizar esos valores porque no hay una regla analítica directa.
* Duda Romina: Si esto es así, porque puedo usar esto después de equilibrar la clases?, si la clase está equilibrada, no necesitaría pesos distintos en cada clase, cierto? **Rta. Mario**: La redacción no es de lo mejor. Quería decir que si se utilizan técnicas para equilibrar la clase se puede usar directamente el LogLoss, una vez aplicado el procedimiento para balancear la clase.

### ¿Que son las probabilidades corregidas?

Si el valor real es igual a 1, entonces no se corrige la predicción. Si el valor real es 0, entonces aplica la predicción corregida: 1- la predicción.

****

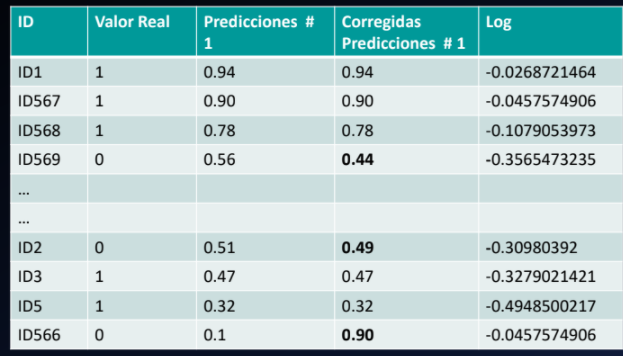
Romina: Las probabilidades son entre 0 y 1. El log cuando el valor llegue a 1 es igual a 0.

Por estos se usa el log. Y los valores se toman cambiados de signo a positivo porque el log entre 0 y 1 es -.

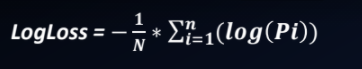
### Aplicamos Logaritmo

Viviana: Tomando el logaritmo puedo minimizar la función y llevar a un valor que pueda medir.

Quiero que sea muy chiquito, por eso hago el logaritmo, lo normalizo y le cambio el signo.



* Como el logaritmo nos da negativo porque las probabilidades van entre 0 y 1 tomamos el promedio negativo.

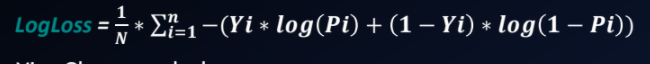
****

Aquí, hace un promedio de como performó para cada registro con respecto a su clase.

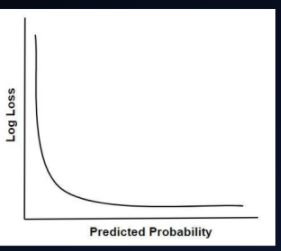
**LogLoss = 0.2144**



* Se puede calcular el LogLoss para cada fila utilizando la ecuación de LogLoss, que mide cuanlejos está cada predicción con respecto a la etiqueta real. Luego calculamos el LogLoss promedio usando todas las filas del conjunto de pruebas. Esta es la formula para realizar lo que dijimos antes en un solo paso.

****

* Yi = Clase verdadera
* Pi = Probabilidad calculada para 1
* 1-Pi = Probabilidad de 0
* i se refiere a todos las filas
* Yi = Clase verdadera
* Cuando la clase es 1 el termino 1 − Yi = 0
* Cuando la clase es 0 el termino Yi ∗ log Pi = 0



Tenemos que buscar el menor log loss. Siemrpe hay que mirar la curva.

Hay que usar los dos métodos y ver cuál de los dos me predice mejor la clase que estoy estudiando.

### Quiz

1. ¿Cuál de las siguientes métricas de evaluación tiene en cuenta la probabilidad de clases?

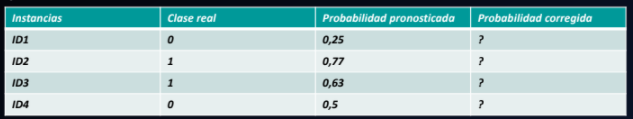
A) Precisión

B) AUC-ROC

C) Log loss

D) F1-Score

A continuación se muestran los valores reales y las probabilidades pronosticadas para la instancia que pertenece a la clase 1:



1. ¿Cuáles serán los valores de probabilidad corregidos para cada instancia?

A) 0,25, 0,77, 0,63, 0,5

B) 0,25, 0,33, 0,37, 0,5

C) -0,25, 0,77, 0,63, -0,5

D) 0,75, 0,77, 0,63, 0,5

1. El valor de la métrica de evaluación de la pérdida de registros puede ser negativo. ¿Verdadero o Falso?

RTA: del 1ero: B y C, del segundo: D)

*B y C son las respuestas correctas para el primero. La otra sería la D. y la última sería que no puede ser negativo, puede ser cero pero no negativo.*

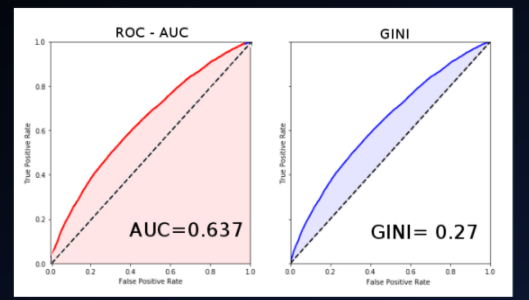
## Coeficiente de Gini

* El coeficiente de Gini se utiliza a veces en problemas de clasificación. El coeficiente de Gini se puede derivar directamente del número AUC ROC. Gini no es más que la relación entre el área entre la curva ROC y la línea diagonal y el área del triángulo anterior.
* La fórmula para el coeficiente de Gini: **Gini = 2 \* AUC - 1**
* Gini por encima del 60% es un buen modelo. Un punto importante a tener en cuenta es que este coeficiente de Gini es diferente del índice de Gini que anunciamos en los árboles de decisión.

Cuanto mejor funciona con respecto al valor cero.

Lili: Area que queda entre 0,5 y la superficie total. Si tenés menos que 0,5, hay que mirarlo del otro lado.

Sirve simplemente porque si tengo 2 números y les sumo 0.5, saber la diferencia entre esos 2 números

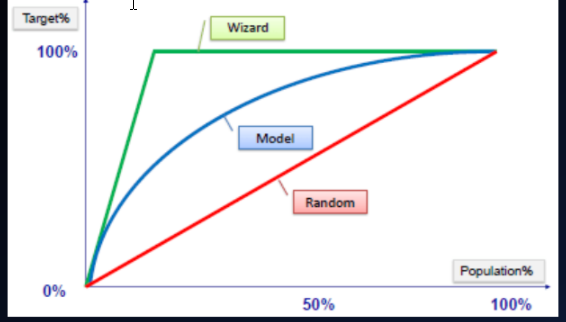
****

Duda: Por qué la fórmula de GINI es (2\*AUC)-1?, ya que al multiplicar AUC por 2, me queda el doble del triángulo y el doble del GINI y luego cunado le resto 1, me quedaría el doble de GINI y no GINI, cierto?

## Gráficos de Ganancia (Gain) y Elevación (Lift)

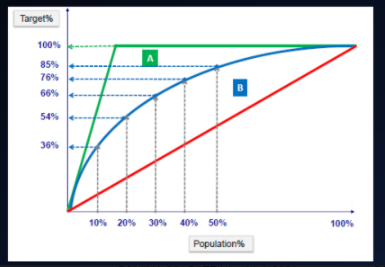
### Gráfico de Ganancia (Gain)

* Normalmente se denomina “Gráfico de ganancias acumuladas”
* Ejemplo:
  + Tenemos 1000 clientes. Si realizamos una campaña publicitaria para todos nuestros clientes, podríamos encontrar que el 30% (300 de 1000) responderá y comprará nuestro nuevo producto.
  + El marketing dirigido a todos nuestros clientes podría ser una estrategia para ejecutar una campaña pero es más efectivo apuntar sólo aquellos clientes con una alta probabilidad de responder positivamente a la campaña para reducir el costo de la campaña y, en segundo lugar para no molestaremos a los clientes con publicidad que no tienen interés en nuestro nuevo producto.
  + Si tenemos datos históricos con las reacciones de los clientes a campañas pasadas, podemos usar los datos para construir un modelo.
  + Los resultados de dicho modelo son típicamente, para cada cliente, la probabilidad de una
  + reacción positiva y negativa del cliente.
  + Podemos clasificar a los clientes de acuerdo con la probabilidad de una reacción positiva a la campaña y ejecutar la campaña solo para un porcentaje de clientes con mayor probabilidad.
* En el eje X tenemos el porcentaje de la base de clientes que queremos apuntar con la campaña. El eje Y nos da la respuesta a cuál es el porcentaje de todos los clientes de respuesta positiva que se han encontrado en la muestra objetivo.



Dudas Romina:

* + No entiendo bien la curva verde. RTA Mario: **La curva verde es si tuviéramos un algoritmo que ordena perfectamente a los registros de modo que los que tiene mayor probabilidad se clasifiquen efectivamente entre los mejores. Es como si efectivamente conociera la clase del dataset.**
  + La recta roja, es el azar, cierto?. Rta. Mario: **Si exacto**
  + En el texto quisiste decir 36% que da 108 clientes, en lugar de 28%, cierto? Rta: Mario: **Si, mil gracias!**
  + El eje X es el % de los clientes que más compran, en esto caso serían los 300 clientes, es decir el 30% de total de clientes que es 1000? Rta: **No, el eje X es el porcentaje de los clientes ordenados. Es decir se ordenan por probabilidades y luego se grafican cortando y calculando cual es el porcentaje de los realmente positivos en los deciles.**
* ¿Qué sucede si solo apuntamos al 10% de nuestra base de clientes?
* Si tomamos el 10% de los clientes con la mayor probabilidad de una respuesta positiva, obtendremos el 28% de todas las posibles respuestas positivas. Lo que significa que encontraremos 84 clientes con respuestas positivas de los 100 clientes alcanzados por la campaña (84 es el 28% de los 300 clientes de respuesta positiva en nuestra base de clientes).
* La elección del porcentaje al que se dirigirá la campaña depende de los costos de la campaña y de los beneficios de las respuestas positivas esperadas.

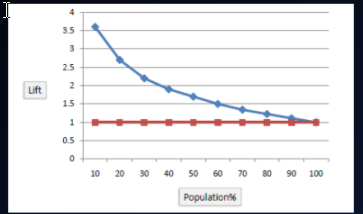


Aquí me dice cuántos pesqué. Aquí me dice el 10% de los mejores compradores, de esos el 36% van a responder de forma positiva, es decir van a comprar. La población es el resultado del algoritmo ordenados de los mejores peores.

### Gráfico de Elevación (Lift)

* Un gráfico de elevación (LIFT) proviene directamente de un gráfico de ganancias (Gain), donde el eje X es el mismo, pero el eje Y es la relación entre el valor de ganancias del modelo y el valor de ganancias de un modelo que elige clientes al azar
* Muestra cuántas veces el modelo es mejor que la elección aleatoria de casos.

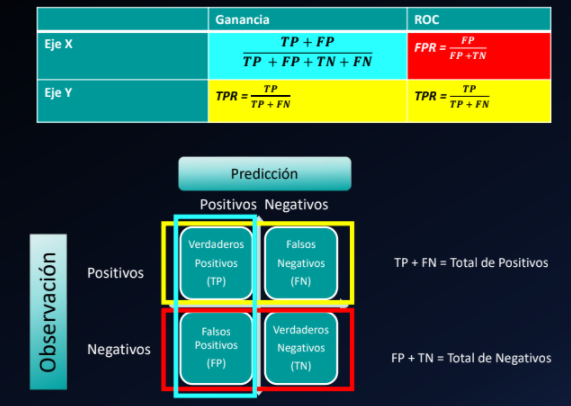
Cuanto mejor que el azar estoy funcionando. El rojo es el azar. El lift es el azul.

**

Dudas: cómo llegamos a lograr este gráfico?

## Diferencia entre Gain y ROC

* Ambas curvas muestran la dependencia de la categoría correctamente predicha en cuestión con el cambio de corte de asignación a esa categoría.
* La diferencia es la escala en el eje X del gráfico, mientras que el eje Y es el mismo para Ganancias y gráfico ROC.

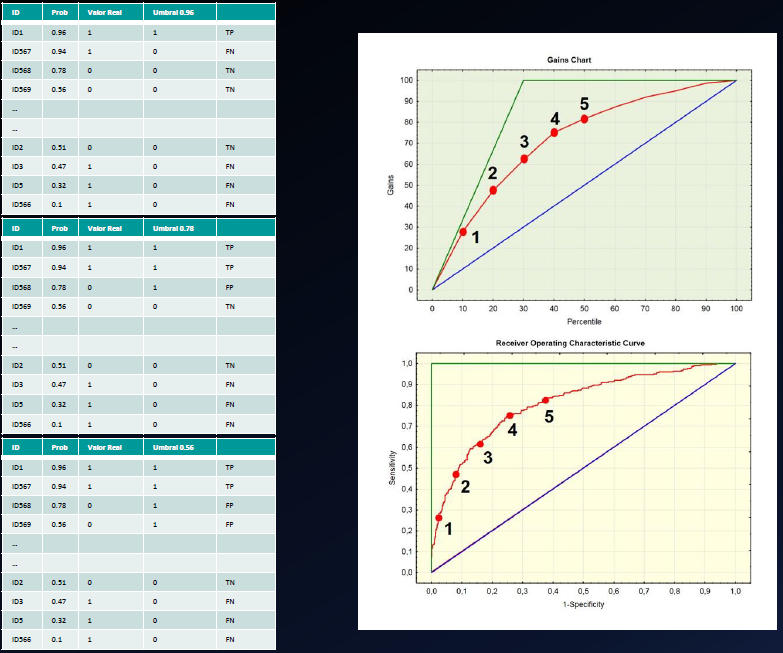


Dudas:

No entiendo que quiere decir esta frase: “Ambas curvas muestran la dependencia de la categoría correctamente predicha en cuestión con el cambio de corte de asignación a esa categoría.”

Que trata de medir en el eje X de la curva de Ganancias?

**Rta Mario**: “corte de asignación” es el Umbral. El eje X en el gráfico de Ganancia indica el porcentaje de los que se clasificaron positivo sobre los totales. Recordá que prácticamente se ordenan según la probabilidad calculada con respecto al total. Ósea se ordenan según la probabilidad calculada y luego para cada decilo se calcula cuanto verdaderos positivos hay. SI haces las cuentas te queda expresado de ese modo.



Son parecidas, pero tienen el eje x diferente. No nos va a interesar un modelo único. Hoy tenemos mucha capacidad de cálculo.

Duda: No entiendo de donde sale la curva Gain?

Rta. Mario: Fijate que en este caso justamente están ordenados por la probabilidad… entonces Gain se calcula tomando el 10 % con mayor probabilidad y viendo cuantos verdaderos positivos hay. Y así para cada punto… es decir se toma el 1%, el 2% el n% de los mejores y se calcula cuantos verdaderos positivos hay. Para curva ROC en vez para cada umbral se Calcula el FPR y TPR

## Gráficos de Kolmogorov-Smirnov (K-S)

Aquí, se intenta buscar el mejor punto entre las dos poblaciones.

El K-S es una medida del grado de separación entre las distribuciones positivas y negativas. El K-S es 100 si el score divide a la población en dos grupos separados donde uno de los grupos contiene todos los positivos y el otro todos los negativos.

Probabilidad Cuentas Acumuladas Inferior Superior No-Target Target No-Target Target K-S



Intenta ver el mejor punto para separar las dos poblaciones. Es parecido al gain en el sentido que en el eje de las x tengo porcentaje de la población, ordeno por las probabilidades y voy a dividir en decilos o algo, de acuerdo a la probabilidad. Me voy a fijar cuántos están dentro de lo que yo quería o no. Me va a dar un porcentaje, por eso voy a ver una curva de cómo va a ir aumentando el % de los que son target y los que no son target. La curva azul es la del target y la naranja es la del no target. Genero las frecuencias acumuladas en porcentaje. El K-S me da el valor máximo de distancia que existe entre ambas curvas.



Romina: Si el modelo funciona muy bien, significa que estoy comprobando algo que ya sabía. Por ejemplo los que trabajan en un banco ya saben que tienen que mirar para saber si un cliente si va a comprar o no.

Los modelos aprenden de los que ven, nunca van a extrapolar.

Cuando tengo un modelo y me llega un set de datos nuevos, lo divido en tres set de datos: test, training y validación. Entonces ajusto el modelo con training y test y luego cuando estoy seguro, pruebo con el set de datos validación para ver cómo funciona el modelo con estos datos que nunca había usado.

Nota de Viviana: Mario va a buscar algún data set para que practiquemos. Las métricas y el modo de aplicar y demás va a depender mucho del dominio donde estemos trabajando.

Conviene asesorarse sobre los temas. Los datasets van a ser parecidos en los dominios. Los expertos ya saben más o menos como funcionan.

Dudas:

* Target seria la cantidad de cuentas/cliente que sería positivo que formen parte de nuestra campaña publicitaria?. **Rta. Mario:** Si, aquí recordá que lo que haces es haces un modelo y target serian sobre los que debería actuar. De nuevo aquí ordeno por probabilidad y calculo el porcentaje de Verdadero positivos y también de verdaderos negativos… en otras palabras es casi igual al grafico de Gain pero calculando el gain para las dos clases.
* No Target seria la cantidad de cuentas/cliente que sería negativo que formen parte de nuestra campaña publicitaria? **Rta. Mario**: Exacto
* Con respecto a esta frase: “El K-S es una medida del grado de separación entre las distribuciones positivas y negativas.” Duda: No veo como me sirve esto para sacar alguna conclusión sobre el algoritmo que estoy evaluando. **Rta. Mario:** Aquí estas comparando como si fuera el gain para las dos clases. Que clasifiques bien a los positivos no necesariamente te habla como estas clasificando los negativos. Si vos tenes un problema donde no solo es importante encontrar a los positivos pero también evitar tomar los negativos esta métrica te ayuda a comprender el poder discriminante de tu modelo.
* Con respecto a esta frase: El K-S es 100 si el score divide a la población en dos grupos separados donde uno de los grupos contiene todos los positivos y el otro todos los negativos. Duda: no entiendo esta frase. **Rta. Mario**: Quiere decir que es perfecto que clasifica a todos los positivos como positivos y a todos los negativos como negativos. Ósea podría haber modelos que es excelente en reconocer a los positivos pero puede suceder que clasifique mal a los negativos.

## Notebook

2da Clase - ROC\_GAIN\_LIFT

# 3ta clase – 24/06/2021

## AUC - ROC

La curva ROC no es fácil de entenderla, lleva tiempo y la vamos a encontrar como métrica en varios lugares.

Este es un sitio en Internet que modela dos poblaciones y muestra cómo te daría la curva ROC cuando vas moviendo las poblaciones y se van acercando.

<http://www.navan.name/roc/>

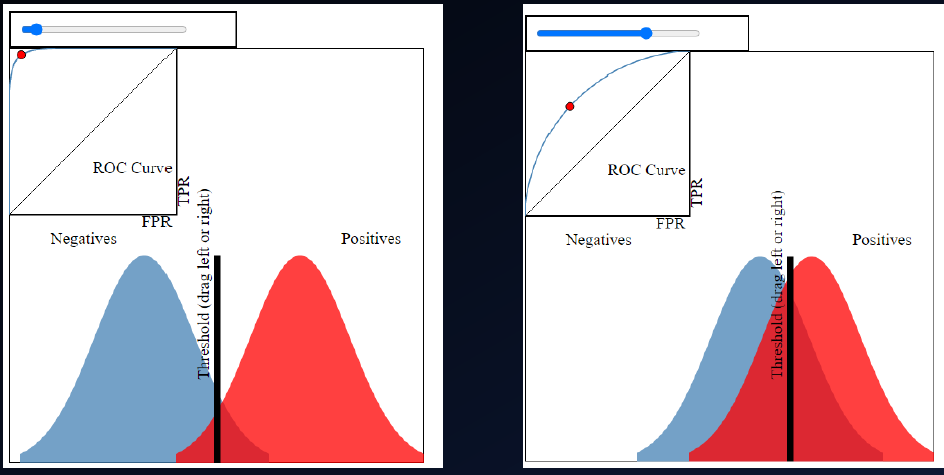
Cuando tenés una curva casi perfecta, significa que las dos poblaciones están muy separadas. Cuanto más Overlaping tenés en las dos poblaciones, la curva se acerca más a la línea de base.

Cuando las dos poblaciones coinciden exactamente, cualquier cosa que elijo, voy a poder elegir los + o -, no tengo forma de distinguirlos.

La línea negra indica cómo me muevo sobre la curva ROC, es decir, de donde salen esos puntitos, la TPR y FPR calculadas para cada umbral, en todo el arco de la población, es la distribución de las probabilidades de cada una de clases

Cando estoy a la cima de la montaña azul, voy a tener una probabilidad más alta de ser negativo.

La curva ROC evalúa el modelo en todo el rango de posibilidades que tiene. Me dice cómo se comporta el modelo, si clasifica mejor los positivos o los negativos y puedo comparar distintos modelos.



## Benchmarking y Línea de Base

Son comparadores.



Cualquier modelo vas a ser tan fuerte como el más débil de sus componentes.

### Benchmarking

* Evaluación y comparación de modelos.
* Desarrollo un nuevo modelo y funciona como los modelos anteriores
* Confirmar que un nuevo método puede identificar correctamente de manera confiable patrones que se sabe otros métodos identifican.
* Puedo comparar mi nuevo modelo con otros que ya sea que funcionan y ver si el mío es mejor o peor que los demás. Puedo estudiar algunas métricas y correr en paralelo modelos que ya están estudiando.
* Permite ver en modo comparativo fortalezas y debilidades relativas a otros modelos
* Para esto se usan datasets específicos. Sirven para probar un modelo.
* Los **Datasets** para **benchmarking** viene en tres gustos:
  + Datos accesibles y bien estudiados del mundo real
  + Datos simulados, o datos que se han generado artificialmente, a menudo para "parecer” datos del mundo real, pero con patrones
  + Datos generados artificialmente (Toy Datasets)que tienen patrones conocidos, pero que no necesariamente representan el mundo real, por ejemplo
* Es decir, pruebo mi modelo junto con otros modelos que ya sea que funcionan con cierto dataset y evalúo las métricas para mi modelo y los otros Para ver si performa mejor o peor

### Línea Base

* Es importante saber cómo poner la línea base. Es para saber si mi modelo está aprendiendo más de, mínimo hemos definido. Por ejemplo en la curva ROC con clases balanceadas, el AUC debe ser mayor a 0,5
* Cuando se evalúa un modelo de aprendizaje automático y se obtiene una de las métricas es necesario saber si es significativa
* En los modelos de clasificación desequilibrados, no es tan fácil determinarlo a priori. Ejemplo:
  + Se construye un clasificador para predecir quién ganará las elecciones basado en todos los distritos de un país
  + El modelo tiene una Accuracy deldel86 %
  + Parece muy bueno pero sabemos que el 84 %de todos los distritos votan al partido A
  + Ahora bien la mayoría de los distritos que votan al partido A son más pequeños y por lo tanto hay muchos más!!!!!
  + Si considerara este caso la clase claramente está desbalanceada
  + De hecho si genero un clasificador que predice que todos los distritos votaran al partido A tendré una Accuracy deldel84 %!!!!!
* Mario dijo que nos va a traer algunas herramientas para ver como taclear a las clases desbalancea das. Las clases desbalanceadas son un problema que vamos a tener todo el tiempo. Hay formas de corregirlas y de tratarlas, pero hay que tener claro cuál es la línea de base

## Clasificador Tasa Cero ***ZeroR y*** Clasificador Aleatorio Ponderado

Con un clasificador aleatorio, si tengo una clase que no está balanceada, para saber que estoy funcionando mejor que el azar, no es tan simple como el 50%. Tengo que hacer algunos cálculos.

### Clasificador Tasa Cero

* Clasificador de tasa cero(o **ZeroR** del inglés Zero Rate) clasifica siempre utilizando la clase más frecuente. En el caso de un modelo binario la mayoría de las veces estará en lo correcto si la clase está desbalanceada.
* Define la línea base, prediciendo siempre la clase más frecuente. Es decir tengo un modelo que predice la clase más frecuente, entonces si le gano a esa modelo quiere decir que mi modelo está funcionando mejor que la línea de base mínima definida. En el ejemplo de las elecciones, la línea base sería el 84%. Si predigo más del 84%, entonces quiere decir que estoy agregando valor.
* La selección siempre del grupo mayoritario es útil como **línea de base** (baseline) y es el que se elige para cualquier modelo de clasificación. Para que un algoritmo de aprendizaje automático demuestre que tiene habilidad en un problema, debe lograr una precisión mejor que este valor cero. (llama valor cero a la línea base)
* Quiso decir: ....demuestre que tiene la habilidad en un problema de clasificar por encima de eso.
* Para problemas muy desbalanceados una Accuracy incluso un poco más alta que **ZeroR** puede ser significativa. Sin embargo, de cualquier manera, el modelo debe ser mejor que **ZeroR** para que se considere útil en un problema
* El Accuray siempre tiene que estar por encima del % de la clase mayoritaria

### Clasificador Aleatorio (Porcentaje Ponderado)

* El accuracy siempre tiene que estar por encima del % de la clase mayoritaria para que sea útil. Pero también puedo hacer unos cálculos para saber cuánto le estoy ganando al azar
* Otra estrategia que podemos usar para calcular la línea de base es evaluar nuestra Accuracy si adivinamos el porcentaje ponderado de cada clase
* Si elijo al azar, tengo la probabilidad de elegir con cierto %. Ahora cuando tengo una clase muy desbalanceada lo que me conviene hacer es un modelo que elija siempre la clase mayoritaria y entonces voy a tener una Accuracy alta. Entonces para que mi modelo sirva, tengo que ganarles al que predice la clase mayoritaria, pero además puede hacer algunos cálculos para ver cuánto le está ganando al azar
* Este valor siempre será menor que el ZeroR y por lo tanto no representa exactamente una línea de base, pero puede utilizarse para explicar y comprender los resultados en términos de cuánto valor agrega nuestro modelo a elegir aleatoriamente
* Clasificador aleatorio: aplica el conocimiento previo de las asignaciones de clase para realizar una asignación de clase aleatoria

## Lanzamiento moneda



* El 50 de los resultados son ceca 0 y el 50 de los resultados son cara 1
* **ZeroR:** adivinar siempre cara nos daría un 50 de Accuracy
* **Clasificador Random:**
* La precisión teórica de adivinar al azar en un problema de clasificación binario es:
* (𝑃(𝑐𝑙𝑎𝑠𝑒 𝑒𝑠 **𝑐𝑎𝑟𝑎**) \* 𝑃(𝑎𝑑𝑖𝑣𝑖𝑛𝑒𝑚𝑜𝑠 𝑒𝑠 **𝑐𝑎𝑟𝑎**)) + (𝑃\*(𝑙𝑎 𝑐𝑙𝑎𝑠𝑒 𝑒𝑠 **𝑐𝑒𝑐𝑎**) \* 𝑃(𝑎𝑑𝑖𝑣𝑖𝑛𝑒𝑚𝑜s 𝑒𝑠 **𝑐𝑒𝑐𝑎**))
* Probabilidades de cara 0,50∙0,50=0,25
* Probabilidades de ceca 0,50∙0,50=0,25
* **Línea de base** = Probabilidad cara + Probabilidad ceca =

## Clase desbalanceada

* Si nuestro modelo preside mejor que el 62,5%, le está ganado al azar, esta sería mi línea de base en términos de ganarle al azar. De todos modos, este modelo no me serviría sino tiene un accuracy mayor al 75%. Son dos conceptos distintos, unos es saber cuánto le está ganado al azar y el otro es la utilidad del modelo. Puedo hacer los cálculos para saber cuánto está por encima del azar, pero sino es mayor el 75% no me sirve, porque hay un moldeo que es mucho más simple que es presidir la clase mayoritaria.
* Cada problema va a tener su propia línea de base.
* Tenemos un ejemplo donde de la clase se divide en 75 /25 los eventos 1 y 0
* Para que esto sea aleatorio, tendríamos que predecir el 75% de las veces la clase 1
* Ahora para que sea aleatorio sopesaríamos la suposición de que predecimos el resultado la mayoría del 75 %de las veces
* (𝑃(𝑐𝑙𝑎𝑠𝑒 𝑒𝑠 **0**) \* 𝑃(𝑎𝑑𝑖𝑣𝑖𝑛𝑒𝑚𝑜𝑠 𝑒𝑠 **0**)) + (𝑃\*(𝑙𝑎 𝑐𝑙𝑎𝑠𝑒 𝑒𝑠 **1** \* 𝑃(𝑎𝑑𝑖𝑣𝑖𝑛𝑒𝑚𝑜s 𝑒𝑠 **1**))
* Probabilidades evento **0**: 0,25\*0,25 = 0,0625
* Probabilidades evento **1**: 0,75\*0,75 = 0,5625
* Línea de base=Probabilidad 0+Proabilidad 1=
* Si adivinamos en modo aleatorio, adivinaríamos correctamente sólo el 62 ,5%de las veces. Cualquier modelo de aprendizaje automático que mejore esta línea de base está agregando valor, pero también debe estar por encima del umbral del 75 % del ZeroR para que sea útil como predictor
* Hay que tener cuidado contra que estamos comparando, porque aunque las métricas nos den bárbaro, si no están por encima de una línea de base que puede estar determinada algorítmicamente o por algunas otras cosas, se nos va a complicar

## Clasificación Votación

* ZeroR para el ejemplo de las elecciones es 84% que es la clase mayoritaria. Y nuestro modelo tiene un accuracy del 86%, por lo cual decimos que es útil.
* ¿El clasificador que tenía era tan malo?
* Está por encima de la línea **ZeroR** (86 %vs 84 %), por lo tanto podemos decir que el modelo es útil
* También podemos averiguar cuál sería nuestra línea de base si adivináramos al azar
* Probabilidades de votar **Partido A**:0,16∙0,16=0,0256
* Probabilidades de votar **Partido B**:0,84∙0,84=0,7056
* **Línea de base**=0,162+0,842=0,73 (Esta es la línea base del modelo aleatorio ponderado, es decir
* si adivináramos al azar)
* Por lo tanto un modelo que elige aleatoriamente en modo ponderado, predecirá bien sólo el **73** %de nuestros distritos
* Nuestra **Accuracy** fue del 86 %.
* Esta es una mejora del 13 %! El modelo, que a priori no parecía tan prometedor, agrega valor
* Importante!!!! Está por encima de la línea ZeroR
* Si elijo la clase mayoritaria, ya es un modelo, no estoy eligiendo al azar. Estoy forzando al modelo a elegir algo específicamente. Sobre todo en las competencias, haciendo unos truquitos alrededor de esto, podes lograr fácilmente ganar y sacar ventaja

## Conclusiones

* El modelo tiene que preformar mejor **ZeroR** para ser útil en la predicción
* **¡¡¡No hay forma de evitar eso!!!!!**
* Se puede Comparar el modelo con la línea de base teórica de predicciones aleatorias ponderadas para evaluar la utilidad de su modelo
* El modelo debe funcionar mejor que ZeroR (84%) siempre y además se puede comparar el modelo con la línea de base aleatoria ponderada (73%) para saber la utilidad del modelo, que en esta caso sería del 13%
* Hay otras líneas de base que pueden resultarle **útiles (uniform, guessing, random guessing yOne Rule, entre otras)**
* Sklearn.dummy.DummyClassifier, ofrece soluciones automatizada para la línea de base con las siguientes estrategias**:“stratified”, “most\_frequent”, “prior ”, “uniform”, “constant”**
* Cree su línea de base antes de construir su modelo y establezca las reglas para las cuales evaluará su modelo final
* Si alguien nos dice, tengo un modelo que funciona bien, además de decir con que métricas lo está midiendo, nos debe decir cuál es la línea de base (cuanto mejor funciona), contra qué lo está comparando. Esto va a ser importante para poder predecir. Por ejemplo si yo en marketing tengo a Pepe Marketing, puede ser que ese sea nuestra línea base, que sabe decir quién va a comprar o no. No siempre sale de un algoritmo, puede salir algunas veces de algo empírico. Sería a que le tengo que ganar para que tenga sentido

## Notebook

Linea\_Base-checkpoint

## Función de Perdida

* Con ese concepto hay confusiones Función de pérdida, función de costo, etc. van a ser cosas que van a estar ahí cuando nos metamos con algoritmos más complicados que no sean los árboles y que tengan distintos pasos intermedios y distintas cuestiones iterativas. Hay que tener en claro de qué estamos hablando. Hay que ponerse de acuerdo con el interlocutor que tengamos de qué estamos hablando
* Recomendación: Siempre hay que asegurarse de que estamos hablando de la misma cosa, ya que el concepto de Función de pérdida en un concepto medio abstracto que se puede aplicar en distintas cosas.
* Las métricas son funciones que queremos utilizar para evaluar la calidad de nuestro modelo. Por ejemplo, para una clasificador, queremos maximizar la Accuracy de nuestras predicciones
* Las métricas sirven para medir la performance/calidad del modelo. A veces las métricas no son fáciles de optimizar, por ejemplo el Acuracy no es fácil de mejorar. Generalmente para mejorarla tienen que ser funciones continuas y derivables, entonces hacés la derivada igual a cero y encontrás un mínimo. Pero ni siquiera esto porque a veces las funciones pueden tener más de un mínimo o más de un máximo y entonces tenes que buscar la función específica para cada algoritmo, esto es lo que pasa dentro del algorítmo. Hay una función que se minimiza y da algo, luego la métrica nos dice cuan bien o cuan mal funciona el algoritmo
* Pero el problema es que nadie sabe realmente cómo optimizar la Accuracy de manera eficiente
* En cambio, se crean funciones de pérdida que son fáciles de optimizar para un determinado modelo
* La función de pérdida en general se debería utilizar solamente cuando se habla de entrenamiento del algoritmo. Es lo que hace el algoritmo cuando aprende.
* Por ejemplo, **LogLoss** se utiliza ampliamente como función de perdida, mientras que la **Accuracy** es la forma en que se evalúa la solución
* Las funciones de pérdida son las que están dentro del algoritmo que hacen que el algoritmo aprenda. Son las que el algoritmo trata de minimizar para decir: estoy aprendiendo.
* Por lo tanto, la función de pérdida es una función que permite optimizar nuestro modelo, y la métrica es cómo queremos que se evalúe la solución
* Pude ser que encuentren alguien que hable de función de pérdida, cuando nos encontremos con Redes Neuronales, que tiene que ver con algo que vamos a ver más adelante que tiene que ver con los hiperparámetros

## Función de Perdida vs Métrica

* La función de pérdida se utiliza para entrenar un modelo
* Es lo que tengo que minimizar, lo que está dentro del modelo que va iterando, cada vez que viene una fila, mete los valores y dice: con esta función de pérdida y los nuevos valores la quiero hacer cada vez más chica
* La métrica se utiliza para evaluar su modelo
* Una vez que el algoritmo entrenó, la métrica nos dice como éste funciona
* Por lo tanto se utiliza una función de pérdida durante el proceso de aprendizaje y una métrica después del proceso de aprendizaje
* Ejemplo:
  + Suponiendo que entrena tres modelos diferentes, cada uno de los cuales usa algoritmos y funciones de pérdida diferentes para resolver la misma tarea de clasificación de imágenes. La elección del mejor modelo en función del error de pérdida no siempre funcionaría, ya que no son directamente comparable. Por lo tanto, las métricas se utilizan para evaluar sus modelos entrenados
* En general, cuando la métrica de error de pérdida disminuye, sus puntajes mejoran
* Con el logaritmo se ve más proporcional a los valores reales (no se si esta nota es de aquí)
* Cuando la función de pérdida disminuye, la métrica tiende a aumentar. Ambas están relacionadas, pero son dos cosas distintas
* Por tanto, los dos están vinculados y comparten el mismo objetivo
* No siempre tenemos suerte y el modelo lo puede optimizar nuestra métrica
* En el caso que se pueda la función de pérdida es la misma que la métrica
* Por ejemplo: Logloss que es una métrica que vimos, también puede funcionar como función de pérdida
* A veces queremos optimizar métricas que son realmente difíciles o incluso imposibles de optimizar directamente. En este caso, generalmente configuramos el modelo para optimizar una función de pérdida que es diferente a la métrica objetivo, pero después de entrenar un modelo, usamos heurísticas para negar la discrepancia y ajustar el modelo para que se ajuste mejor a la métrica
* Competencias Kagel: Genere un modelo que sea el mejor, medido con X métrica. Para meterle al algoritmo algo para que mejore dicha métrica a veces no es fácil y entonces hay que buscar una función de pedida.
* Por eso a veces aparece el concepto de función de pérdida mezclado con métrica. Por eso siempre hay que confirmar si lo que estamos viendo para un algoritmo determinado es una métrica o una función de pérdida
* Heurística en matemética: la base de la heurística está en la experiencia de resolver problemas y en ver cómo otros lo hacen
* Heurística en informática. Para la informática, la heurística consiste en encontrar o construir algoritmos con buena velocidad para ser ejecutados

## Parámetros e Hiperparámetros

Son conceptos que pueden llegar a mezclarse.

### Diferencia entre Parámetros e Hiperparámetros

#### Parámetros

* Son las propiedades de los datos de entrenamiento que el modelo de Aprendizaje Automático aprende durante el entrenamiento.
* Por ejemplo:
  + Ponderaciones o coeficientes de variables independientes en el modelo de regresión lineal.
  + Pesos o coeficientes de variables independientes SVM.
  + Puntos de división en el árbol de decisiones.
* Son las propiedades que ponemos en el modelo para que éste aprenda. Los parámetros del modelo, son los que el algoritmo va a ir aprendiendo. Estos, se van a ir midiendo con la función de pérdida

#### Hiperparámetros

* Son comunes para modelos similares y no se pueden aprender durante el entrenamiento, pero se establecen de antemano y permiten optimizar el rendimiento del modelo.
* Por ejemplo:
  + Kernel en SVM
  + Valor de K en el KNN
  + Profundidad del árbol de decisión
* Se fijan inicialmente para poder correr el moldeo, luego una vez que el algoritmo aprendió, se van ajustando. Con esto voy mejorando el algoritmo
* ¿Cómo elijo los hiperparámetros?: Muchas veces hay que hiperparametrizas, esto significa que tengo que generar otro tipo de algoritmo para encontrar los mejores hiperparámetros
* La optimización de Hiperparámetros es muy importante para que el algoritmo aprenda
* A veces se pueden obtener las hiperparametrizaciones de otros modelos o tomarlas como referencia de otros sitios.
* Lleva más tiempo encontrar los hiperparámetros ópticos, que entrenar el modelo



##### Métodos de optimización Hiperparámetro

* Grid Search
* Random Search
* Bayesian Optimisation
* Una métrica no tiene que vez solamente de los que vimos. Se pueden usar métricas inventadas, Por ejemplo: función de costo. Hago un algoritmo donde a cada cliente que predigo que se iba a ir de una cartera Premium le doy un valor X; sino no predije le doy un valor -Y. entonces genero un métrica para calcular la función de ganancia. Este mejoramiento del algoritmo tiene que ver con la hiperparametrización

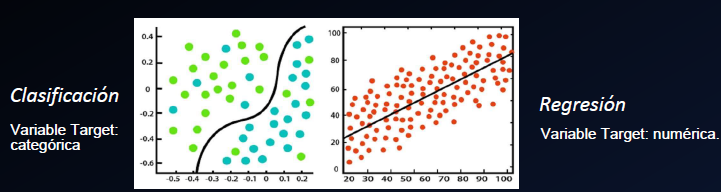
## Métricas Regresión

Aquí vamos a ver como mide una regresión, para ver si es buena o mala.

En algunos casos, algunos algoritmos de clasificación, los puedo pensar como una regresión. Y aquí las cosas se van mezclando.

### Aprendizaje automático supervisado – Problemas de regresión

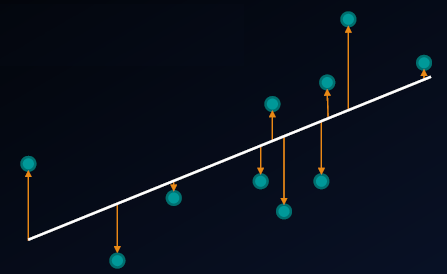
* Cuando uno está interesado en responder “cuanto”, estos problemas caerían bajo el paraguas de regresión. Por ejemplo, cuál es la cantidad esperada de incumplimiento de un cliente es un problema de regresión

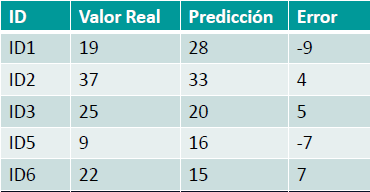


* Métricas que se utilizan en regresión (independientemente del tipo de regresión que se esté analizando
  + Error medio cuadrado ( MSE ).
  + Error cuadrático medio ( RMSERMSE).
  + Error logarítmico cuadrático medio ( RMSLERMSLE).
  + R2.
  + R2 ajustado

### ¿Qué definimos como Error?

Distancia entre la observación y la predicción

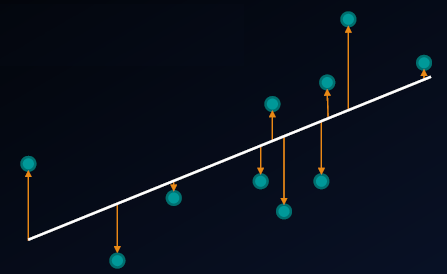




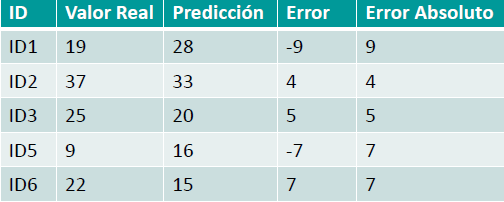
### MAE: Error Medio Absoluto

Tomo el promedio de los valores absolutos





Donde 𝑌𝑖 es el valor real e  es el valor predicho



Esta función no es fácil de derivar. Es una función de porquería

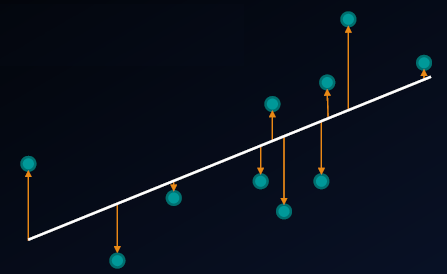
MAE = 5,6

### MSE: Error Medio Cuadrado

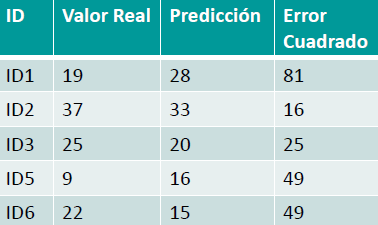
Tomo el promedio de los errores al cuadrado

La idea del R2 ajustado es ponerle un término que tenga en cuenta tanto las variables que estamos utilizando como las cantidades de filas





Donde 𝑌𝑖 es el valor real e  es el valor predicho





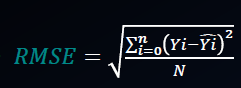
El problema aquí, es que el Error Cuadrado Medio, no está en las mismas unidades que las variables que estaba midiendo. En ese caso, dicho error está en m2, cuando lo que estoy midiendo es en m. De todos modos, puedo minimizar el error, luego veo como lo informo ya que no está en las mismas unidades que las variables que estoy midiendo.

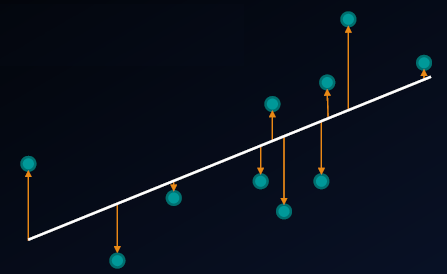
𝑀𝑆𝐸=34,4 m2

Hay otras métricas que funcionan un poco mejor. Igualmente puedo calcularlas a todas, ver que me están indiciando e ir acercándome.

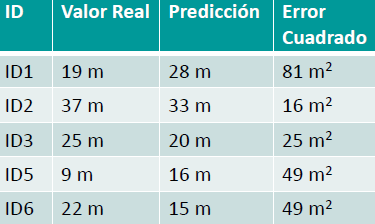
### RMSE: Error cuadrático Medio

Tomo el promedio de los errores al cuadrado





Donde 𝑌𝑖 es el valor real e  es el valor predicho



𝑀𝑆𝐸=34,4 m2

𝑅𝑀𝑆𝐸=5,86 m

Entonces, lo que se hace es tomar la raíz cuadrado del error cuadrático medio. Este error está en el rango del error medio absoluto.

El problema que la raíz cuadra es un poco más amigables que error medio absoluto, pero no es tan amigable. Por eso se sigue trabajando con el error medio cuadrado, porque es más fácil de trabajar

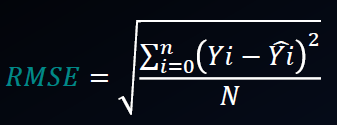
#### ¿Por qué puede a vecesno ser el mejor?

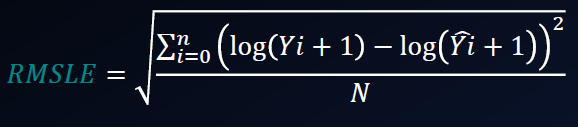
Mide cuanto estoy lejos del valor real.





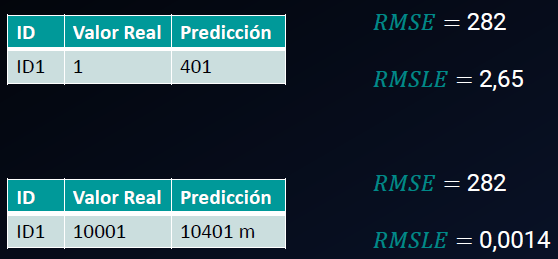
### RMSLE: Error logarítmico cuadrático medio





Reduce los valores muy grandes por las propiedades del logaritmo

### RMSE: Error cuadrático Medio vs RMSLE: Error logarítmico cuadrático medio



Aplicando logaritmo, hago que el error sea proporcional a los valores reales.

* Ahora que sabemos que es el **RMSE** y el **MSE**, sabemos que si esto valores decrecen nuestro modelo está más ajustado.
* Pero Esto por si solo no es suficiente para saber la performance del modelo.
* Tenemos que tener una línea de base. En realidad todos los algoritmos necesitan línea de base
* Cuando tratamos las métricas de clasificación la línea de base era casi intuitiva y vimos que si nuestro modelo tenia métricas que se alejan del **ZeroR** y azar es un mejor modelo
* Ahora ¿cuál es nuestra línea de base para comparar en el caso de Regresión?

### Quiz

Suponga que tiene que construir un modelo para predecir las ventas de un supermercado

¿Qué métricas de evaluación podemos utilizar para este problema?

A) Accuracy

B) Error medio cuadrado

C)Precision

D) RMSLE

El error siempre es positivo. ¿Verdadero o falso?

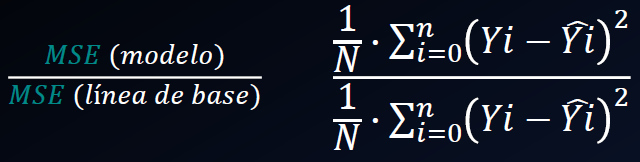
A) Verdadero

B) Falso

RTA: B) y D) y Falso

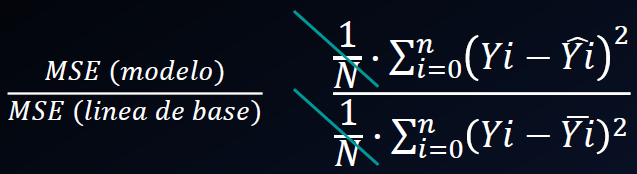
### MSE Error cuadrático Medio Relativo

Como vimos, la línea de base es algo a lo que le tenemos que ganar.



 es la predicción.  es el promedio, no debería tener subíndice i. Esto es, todos los puntos para que funcionen un poco mejor tienen estar distanciados del promedio, un poco menos. Esta es una de las líneas de base, que se puede usar en regresión.

**MSE** (línea de Base) reemplazamos  por el promedio de 





En el caso peor **MSE** (modelo) = **MSE** (línea de Base)

El valor es 1

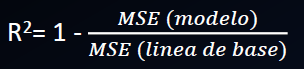
En realidad nosotros queremos que un valor que varié entre 0 y 1 y que en el peor escenario sea 0 no 1

Por esto es que a este error se le aplica algún cambio matemático y se define el R cuadrado

### R –cuadrado R2

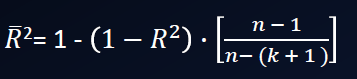
El R2 sale de acá, de tratar de buscar una línea de base a la cual tenés que comparar tu modelo. La transformación de agregarle 1-, es para que el modelo varíe entre 0 y 1.

Con esto estamos comparando todos los valores contra el promedio, el promedio es algo muy malo, de lo peor.

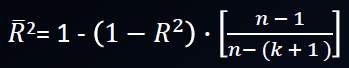


Cuando MSE (modelo) disminuye el valor de R2se acerca a 1

En el caso que tengamos muchas variables el valor de R2 aumenta o es igual nunca disminuye independientemente de cómo las variables impacten en el modelo



### R2 y R2 ajustado



* Regresión lineal simple: tiene una variable independiente y otra dependiente. También tenemos regresiones lineales múltiples, multi-variadas. En las regresiones lineales múltiples tenemos 1 sola variable target y muchas variables que se mueven. En el ejemplo de las casas, queremos predecir el Precio de la casas y entonces utilizamos otras variables que tenemos que combinar de algún modo.
* Sirve para ver el peso de las variables que voy agregando al modelo. Esto es cuando por ejemplo tengo una regresión lineal múltiple
* La idea del R2 ajustado es ponerle un término que tenga en cuenta tanto las variables que estamos utilizando como las cantidades de filas
* Tengo un Dataset con n filas y k variable, entonces ajustamos el R2 por la cantidad de variables. Porque si yo quiero comparar dos modelos, unos que tiene 10 variables y otro que tiene 1000 variables, necesito saber si esas 990 variables que agregué, le están agregando algo de valor al modelo
* Si a medida que voy aumentando las variables R2 aumenta, es decir ((1-R2)\* la fracción) aumenta, entonces 1-un nro. que está aumentando, va a ser un nro. cada vez más chiquito, pero como esto está amentando de algún modo, se van a compensar.
* Donde n es el número de filas y 𝒌 el número de columnas. Es decir n es la cantidad de muestras y 𝒌 la cantidad de variables o features
* Es decir que si el R2 está variando en función de la cantidad de variables que estoy agregando, entonces el R2 ajustado más o menos va a quedar constante
* Por lo tanto si 𝑘aumenta el denominador disminuye, la fracción aumenta y por lo tanto si 𝑹𝟐aumenta  ajustado aumenta.
* Pero si voy agregando un montón de variables y el R2 no varía, entonces la cantidad que le voy a resta a 1 va a ser cada vez más grande.
* Sin Embargo si 𝒌 aumenta y 𝑹𝟐 no aumenta el término que se termina restando es más grande y por lo tanto a un R2 ajustado disminuye
* A veces puedo seguir aumentando la cantidad de variables de mi modelo pero esto no mejora mi sistema. Aquí estaría midiendo cosas que son inútiles. Aquí se debería aplicar un proceso que se llama regularización son dos métodos, que tienen ver con decidir con cuantas variables me voy a quedar por ejemplo cuando quiero hacer una regresión, define cuales son las mínimas variables que necesito para hacerlo
* A mayor 𝒌si no aumenta 𝑹𝟐,  ajustado disminuye
* El problema de agregar cada vez más variables, puede pasar que el modelo esté aprendiendo de memoria en tu entrenamiento y vas a estar ajustando cada vez más a esos datos que estás midiendo (overfiting). Por eso, hay un compromiso entre que el modelo aprenda de los datos con los que está entrenando, pero que tenga la capacidad que cuando se le carguen datos nuevos, poder clasificarlos.
* En ML no se puede extrapolar, los modelos aprenden de lo que ven. Por eso se usa en el contexto de Big Data porque cuando tenés muchos datos, te independizar de algunas cosas de súper molesta pero súper útiles de la estadística como un montón de supuestos, ya que al cubrir un cantidad muy grande de datos, podes estar casi seguro que estas aprendiendo todo el universo que hay para ver Que probablemente cuando venga algo nuevo, el modelo ya lo habrá visto antes.
* El problema es cuando viene algo que el modelo no vio antes, ¿qué hace? ¿cómo reacciona? si el modelo se ajustó mucho, probablemente el modelo me lo va a clasificar mal. Por eso tengo que tener algún modo de evaluar como regular la cantidad de variables
* El problema de aplicar la estadística clásica a grande cantidad de datos es que siempre vas a encontrar una diferencia estadísticamente significativa. Ahora, esa diferencia significativa, tiene sentido?. ejemplo: si yo mido la altura de los habitantes de un país y luego de otro y encuentro que la altura media en el 1ro es 1,74m y en el 2do 1,75m, pero tengo tantos datos que eso lo que me va a permitir es separarlo lo suficiente como para decir que son significativos.
* Entonces la pregunta es: ¿Son efectivamente unos más petisos que otros?, ¿esto tiene algún valor? Por eso hay que tratar que los algoritmos sean lo suficientemente capaces de pescar diferencias pero que no aprendan de memoria.

### Conclusiones

Recomendación general / CONCLUSIÓN

* 𝑅2y 𝑅2ajustado son mejores para explicar el modelo a otras personas porque puede explicar el número como un porcentaje de la variabilidad de salida. (Con respecto a una línea de base)
* Para comparar modelos es mejor utilizar MSE, RMSE o MAE. (Pero siempre voy a tener que calcular el R2 para saber si estoy por encima de la línes de base)
* RMSE es la primera opción ( Kaggle ), sin embargo, se puede usar MSE si el valor no es demasiado grande y MAE si no desea penalizar grandes errores de predicción.
* 𝑅2ajustado es la única métrica que considera el problema de sobreajuste (Overfitting)

### Quiz

¿Cuál de las siguientes afirmaciones sobre r-cuadrado es verdadera?

A) Agregando nuevas variables o features , R R2siempre aumenta

B) Agregando nuevas variables o features , R R2puede aumentar o disminuir

C) Agregando nuevas variables o features , R R2aumenta o permanece igual, nunca disminuye

D) Agregando nuevas variables o features , R R2siempre disminuye

¿Qué Métrica de evaluación que tiene en cuenta la cantidad de variables del conjunto de datos?

A) RMSE

B) R2

C) R2 ajustado

D) Tanto R

R2 como R2ajustado

RTA: C) y C)

## Tipos de Regresiones y símiles

Todo esto es una materia de un cuatrimestre entero en la Maestría de Exactas

* Regresión Lineal Simple
* Regresión lineal Múltiple
  + Multivariadas: cuando querés predecir más de una variable. Para el ejemplo de las casa, querés predecir el precio y el valor de re venta de la casa.
* Regresión Logística
  + Tiene una forma sigmoide y se usa como clasificador.
* Regularización:
  + LASSO
  + RIDGE
  + Tienen un desarrollo matemático que incluyen un término que penaliza, cuanto están aprendiendo. Esto permite aplicando una estrategia LASSO o RIDGE, cuales variables tienen más peso y con cantas te podes quedar para que el modelo siga explicando, sin perder mucha capacidad de explicación. Para el peso de las variables, se calculan los factores por los cuales tenemos que multiplicar las variables para que tengan pesos relativos en el cómputo de la regresión
* Lineal Generalizado (GLM)
  + Como hacer un modelo lineal, pero lo que vas poniendo no necesariamente se mueve linealmente sino que son funciones que pueden ser no lineales
* Modelo Generalizado Aditivo (GAM )
* Generalizado Aditivo Localización, Escala y Forma (GAMLSS)
* Todo esto termina en Redes Neuronales. Hay un paso intermedio y las RN son un modelo donde se pueden utilizar variaciones lineales entre las variables que están utilizando y también relaciones complemente distintas que no son lineales. son muy potentes para lagunas cosas y son poco amigables para explicar. Es muy difícil de entender desde un punto de vista operativo, que es lo que se están midiendo

Un amigo de Mario, hizo un algoritmo para predecir los valores de la bolsa, y los clientes querían saber porque y él les dijo que no podía decirles el porqué, “funcionan pero no puedo decir porque”. A veces son sistemas como una caja negra, que para algunas cosas funcionan muy bien por ejemplo para reconocimiento de imágenes. Y en otros casos, se va por modelos que tengan una capacidad explicativas mayor, para Mario XGBUST es un algoritmo que saliendo de las redes neuronales, están explicando un montón de cosas y puede funcionar como clasificador y como regresión. Se le puede hacer algún tipo de preguntar y puede decir cómo está funcionando, Podes preguntar por el ejemplo: cuál es el peso relativo de las distintas variables en el modelo predictivo que estás usando. No puede saber que aprendió exactamente, pero puedo saber que hay algunas variables que tienen un peso más importantes y esto desde el punto de vista del dominio de trabajo donde estoy me sirven

## Datasets webpages

<https://www.kaggle.com/datasets>

En la clase: Desde el minuto 1:48:21, habla de Kaggle.

Se pueden bajar los códigos.

Desde el minuto 1:48:21, habla de Kaggle. El mostro el código en Kaggle de Breast Cancer EDA and Prediction. Hay de todos los temas.

https://lionbridge.ai/datasets/ultimate ultimate-datasetdataset-aggregatoraggregator-forfor-machine -learning/?ref=hackernoon.com

https://hackernoon.com/top top-10 -regressionregression-datasetsdatasets-forfor-machine -learninglearning-projectsprojects-ce4i3wuu

<https://archive.ics.uci.edu/ml/datasets.php>

<https://hackernoon.com/tagged/datasets?ref=hackernoon.com>

<https://hackernoon.com/tagged/data?ref=hackernoon.com>

<https://datasetsearch.research.google.com/>

## Notebook

3er Clase - Regresion\_I-checkpoint.ipynb

3er Clase - Regresion\_II-checkpoint.

* En la próxima clase veremos el concepto de cross validation que tiene que ver con métricas, pero no es una métrica en sí, pero es un modo de validar y tiene que ver con overfiting.
* Las últimas clases van a ser para algoritmos no supervisados: es más artísticos. No hay que predecir una clase ni un valor. Sino que tenemos por ejemplo estos individuos con ciertas cualidades, agrupármelos del mejor modo posible, busca patrones que dicen quienes se parecen a quien.
* Cuando estamos en un espacio multidimensional, a veces las cosas se complican por cómo están dispuestos los grupos porque no se empiezan a separar de un modo homogéneo, se separan medio entrecruzados y hay algunas cosas un poco más interesantes como para poder bajar y rescindir espacios multidimensionales a cosas más interpretables por nosotros. Ejemplo: un especio multidimensional son los textos, porque cada palabras va a ser una variable.
* Hay procedimiento que te permiten de bajarlo de 300 a dos dimensiones y reagrupar estos textos y empezar a ver patrones.
* Las métricas de patrones son más complejas porque no hay métricas tan precisas. Al menos que tengas un conocimiento no podes saber si se están detectando todos los grupos que existen.

# 4ta clase – 01/07/2021

Antes de comenzar la clase dijo:

A un modelo una vez que saqué las métricas, le puedo agregar variable por ejemplo cuanto gano si es positivo y cuanto pierdo si es negativo y ahí le agrego una variable más para medir mi algoritmo.

El R2 cuadrado tiene intrínseco una línea base Ir a buscar Dataset simples y probar cosas.

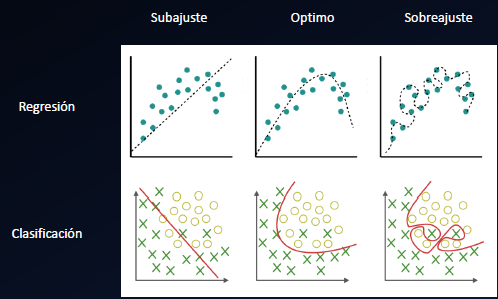
## Cross Validation y Overfitting

A nadie le interesa predecir lo que ya sabes



## Sobreajuste y Subajuste

* Siempre es necesario validar la estabilidad de los modelos de aprendizaje automático.
* No se puede ajustar el modelo a los datos de entrenamiento y esperar que funcione para los datos reales que nunca antes había visto.
* Es necesario asegurarse que el propio modelo posee la mayoría de los patrones útiles, y que no capta demasiado el ruido
* Sobreajuste: es overfiting, es una función que me permite modelar cada uno de los puntos que tengo en el set de datos
* Subajuste: Esto es así y no me importa mucho la división.



Pregunta de Martin: ¿Cómo sigo ajustando mi algoritmo con datos nuevos?

Podrías ser que el cliente le pide a su proveedor del algoritmo que se lo actualice. Depende del negocio, si ingresan muchos datos nuevos todo el tiempo o si el flujo de datos no es mucho.

## Sesgo y Varianza Error total

Si denotamos la variable que estamos tratando de predecir como Y nuestras covariables como X, podemos suponer que existe una relación del tipo:

Y = f X + ε donde ε es el término de error

El error puede ser descompuesto en:

ε = Sesgo2 + Varianza + Error irreductible (ruido)

El **Sesgo** es la diferencia entre la predicción promedio de nuestro modelo y el valor correcto que estamos tratando de predecir. El modelo con alto sesgo presta muy poca atención a los datos de entrenamiento y simplifica demasiado el modelo. Sesgo: A los que se parecen los vas a predecir muy parecidos, pero estas muy lejos de la verdad.

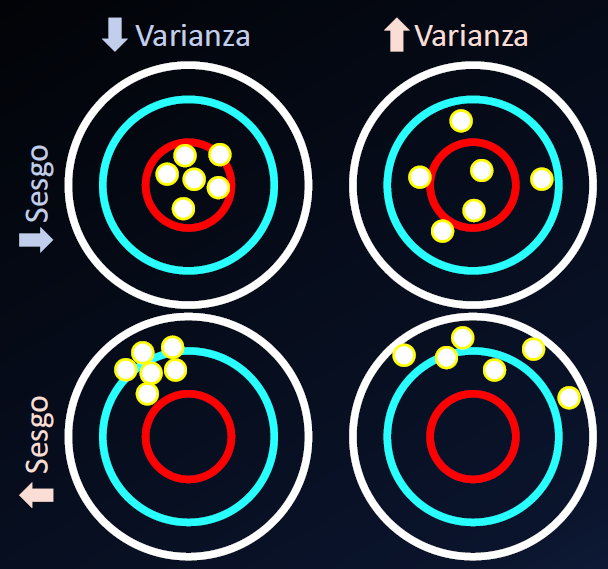
La **Varianza** es la variabilidad de la predicción del modelo para un punto y nos dice la variabilidad de nuestros datos. El modelo con alta varianza presta mucha atención a los datos de entrenamiento y no generaliza sobre los datos que no ha visto antes. Sobreajuste.

El **error irreductible**, es el ruido que no puede ser reducido fundamentalmente por ningún modelo. Si tenés un instrumento que mide, es el error de medición del instrumento. No importa cuánto vos haga, te va a aproximar al límite del error del modelo.

Dado el modelo “verdadero” y los datos infinitos para calibrarlo, deberíamos poder reducir los términos de sesgo y varianza a 0

El sesgo es un peso desproporcionado a favor o en contra de una cosa, persona o grupo en comparación con otra, generalmente de una manera que se considera injusta. Los sesgos se pueden aprender observando contextos culturales.

En teoría de probabilidad, la varianza o variancia de una variable aleatoria es una medida de dispersión definida como la esperanza del cuadrado de la desviación de dicha variable respecto a su media.



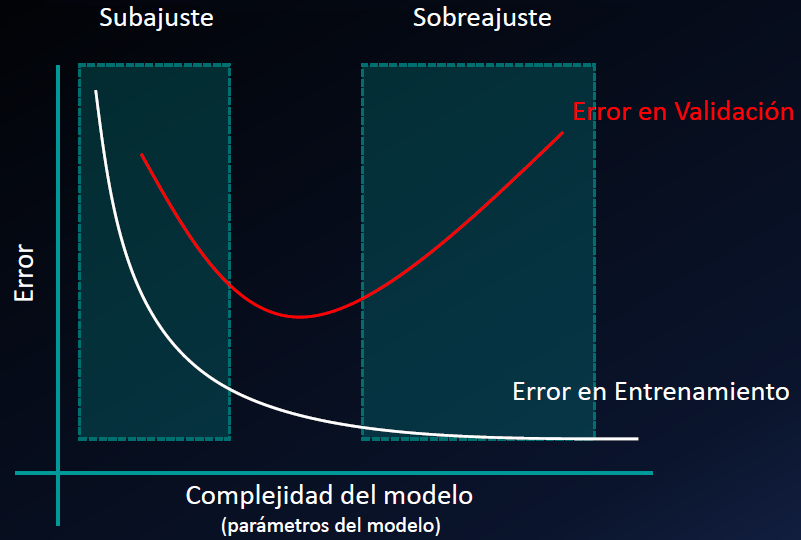
Sesgo: Cuan ajustado estoy al valor verdadero con respecto a la predicción promedio del modelo

Si aumento los datos de entrenamiento, entonces se reducen los datos de testeo y esto aumenta el Sesgo

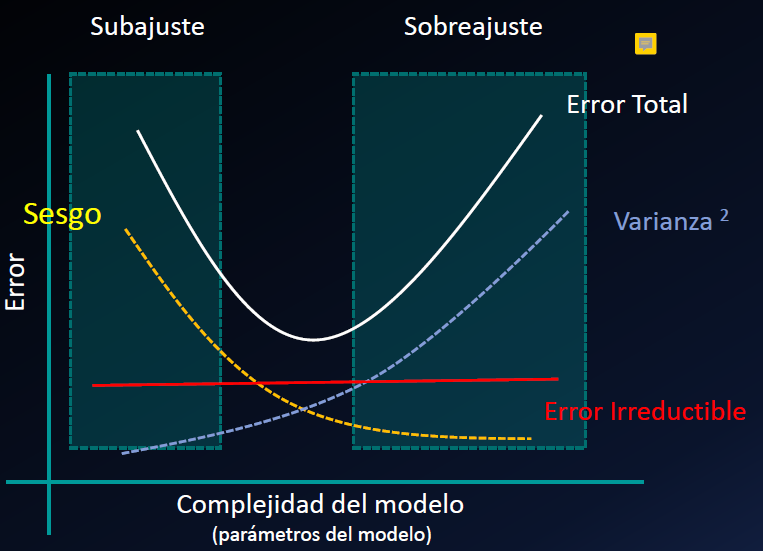
Varianza: Cuanta dispersión voy a tener en clasificarlos

## Sobreajuste Subajuste Sesgo Varianza

* Cuanto más complejo es el modelo, mejora la performance del modelo. Pero si sobre ajusto aumenta el error.
* A menos que mi muestra es representativa de mi población, si esto no es así cuando tengas nuevas datos probablemente el moldeo no va a servir. Por esto está bueno dejar un set de datos para validación que los vamos a usar una vez que hayamos hecho todos los ajustes que queramos y ahí vamos a probar el algoritmo con los datos de validación para ver si los ajustes que hicimos están bien

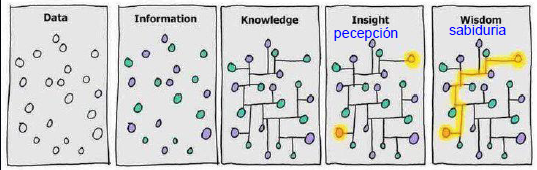


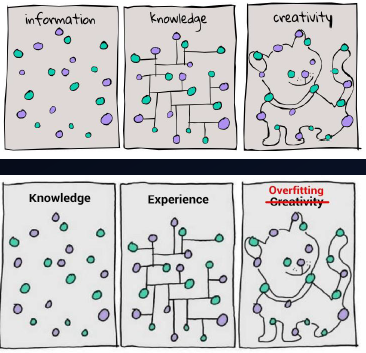
* Error: cuan distante estoy de lo que tengo que predecir.
* Depende del modelo y su complejidad, por ejemplo si uso un función más compleja, por ejemplo un polinomio de mayor grado error
* Si la complejidad de nuestro modelo supera este punto óptimo, estamos sobreajustando nuestro modelo; mientras que si nuestra complejidad no alcanza el punto óptimo, estamos infraajustando el modelo. En la práctica, no existe una forma analítica de encontrar este punto. En su lugar, debemos utilizar una medida precisa del error de predicción y explorar diferentes niveles de complejidad del modelo para luego elegir el nivel de complejidad que minimice el error global. Una de las claves de este proceso es la selección de una medida de error precisa, ya que a menudo se utilizan medidas muy inexactas que pueden ser engañosas. El tema de las medidas de precisión se discute aquí, pero en general las medidas basadas en el remuestreo, como la validación cruzada, deben preferirse a las medidas teóricas como el criterio de información de Aikake (AIC).
* Como se cuándo parar, para no llegar a overfiting?: Hay algunas técnicas que nos permiten saber, donde estoy parado.
* Comprender el sesgo y la varianza es fundamental para entender el comportamiento de los modelos de predicción, pero en general lo que realmente importa es el error global, no la descomposición específica. El punto óptimo para cualquier modelo es el nivel de complejidad en el que el aumento del sesgo es equivalente a la reducción de la varianza.



* Modelo demasiado complicado: Digamos que tenemos muchas variables de entrada, que se relaciona con una alta varianza. El modelo sobreajustará los datos de entrenamiento, teniendo una baja precisión sobre datos nunca vistos debido a su particularización.
* Modelo demasiado sencillo: Por otro lado, puede que el modelo no esté capturando toda la información que hay en los datos debido a su simpleza. Esto está relacionado con un alto sesgo.
* Datos de entrada insuficientes: Los datos crean formas en un espacio n-dimensional (donde n es todas las variables de entrada + objetivo). Si no hay suficientes puntos, esta forma no se desarrolla bien.
* Esto comentarios son para las páginas 5, 6 y 7. Para comprender el desarrollo de las funciones ver: <http://scott.fortmann-roe.com/docs/BiasVariance.html>







No entiendo estos dibujos

Tengo que tener alguna métrica que me permite ver que estoy generalizando

## Notebook

4ta Clase - Sesgo\_Varianza

## Competencia Kaggle

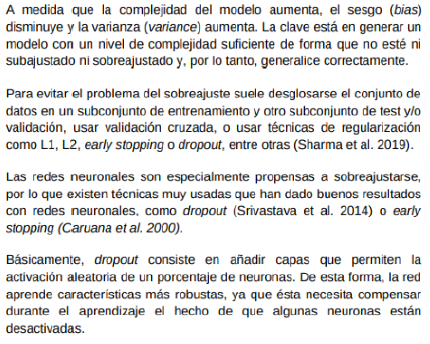
* La competencia se basa en dos tablas de clasificación: pública y privada
* Estas dos tablas de clasificación se crean dividiendo aleatoriamente el conjunto de datos de prueba
* La tabla de clasificación pública se basa en un subconjunto del 30% de las observaciones en el conjunto de datos de prueba y la tabla de clasificación privada se basa en el 70% restante de los datos
* La clasificación privada permanece en secreto hasta el final de la competición
* Mientras la competencia está activa uno envía sus resultados y les devuelven el score basado en la tabla de clasificación pública
* Antes del final de la clasificación de la competencia se debe seleccionar sus dos presentaciones favoritas Finalmente, estas dos presentaciones se clasificaran darán utilizando la clasificación privada
* El propósito de esta división es evitar que gane el sobreajustarse a la clasificación pública y por lo tanto esto motiva a que los modelos generalicen bien.
* Se pueden sacar ideas de aquí.
* A vece limitan los recursos de procesamiento, para normalizar la competencia.
* Son competencia, alguien tiene datos, se queda con una cierta cantidad para test. Y entrega los datos de traning y los competidores hacen un algoritmo que entregan a Kagel esto lo comparan con un Dataset de prueba. Cuando terminan comparar el algoritmo que hiciste con el data set completo. Es de gente que decide donar los notebook a todos

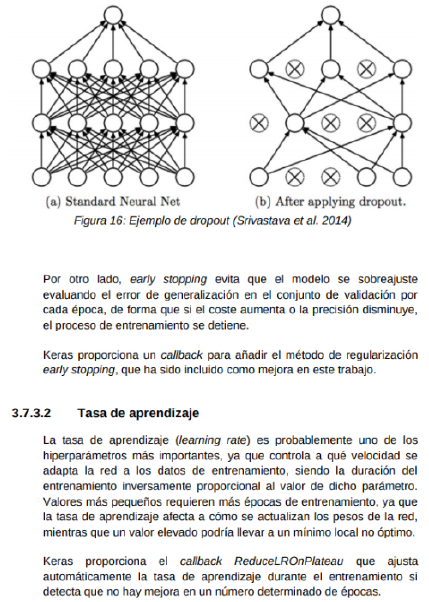


Soluciones Sobreajuste Subajuste Sesgo Varianza

* Usar modelos poco complejos
* Partir la muestra: Entrenamiento / Testeo
* Usar validación cruzada (Cross-Validation)
* Bootstrap y Bagging
* Usar técnicas de Penalización o **Regularización**, como:
  + Regresión Ridge. A las predicciones le agrega un término tal que la predicción cuando va mejorando, ese término empieza a crecer La idea es que me dice que cuando baja el error, este término empieza a aumentar.
  + Regresión LASSO
* **Early Stopping**. Más mecánico, para aquí

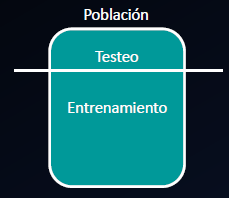
Las dos notas que siguen son para explicar un poco los algoritmos de Regularización y Early Stopping:





## Cross Validation

* Dividir la población es distintas subpoblaciones y luego mezclarlas y volverlas a dividir y volver a probar y volver a hacer varias veces. Si aumento los datos del testeo, me va a aumentar el sesgo, y si uso mucho datos para entrenamiento y poco para test
* La validación cruzada es un concepto simple que ya hemos visto y consiste en dejar una muestra fuera del proceso de entrenamiento del modelo para luego usarla para medir performance de nuestro modelo



* Este es el caso más simple. Pero como ya vimos tiene el problema que utilizamos sólo una cantidad de datos para entrenar/validar el modelo.
* El problema reside en que si aumento los datos de entrenamiento estoy reduciendo los datos de testeo y viceversa y por lo tanto en el primer caso tendré un aumento del sesgo .
* En el segundo caso efectivamente reduciremos el sesgo debido a la selección de la muestra hasta cierto punto, pero proporciona una pequeña muestra para entrenar el modelo

### Cross Validation: distintos enfoques

* K-Fold Cross Validation
* Leave One Out Cross Validation
* Stratified K-fold Cross Validation

Dejar solo un valor para predecir. Como el ejemplo del notebook de sesgo que en los gráficos predice solo un valor. No se usa mucho.

### K-Fold Cross Validation

* Dividimos a toda la población en 5 muestras iguales. Ahora entrenamos modelos en 5 muestras (recuadro gris) y validamos en 1 muestra (recuadro verde). Luego, en la segunda iteración, entrenamos el modelo con una muestra diferente. En 5 iteraciones (k), básicamente hemos construido un modelo utilizando cada muestra tanto en el entrenamiento como en el testeo



* Esta es una forma de reducir el sesgo de selección y reducir la varianza en el poder de predicción.
* Los resultados de K-fold CV se resumen reportando la media de las puntuaciones de los scores del modelo. También es una buena práctica incluir una medida de la varianza de los scores obtenidos, como la desviación estándar o el error estándar.
* Al final voy a tener 5 valores de cada métrica. Si la población e suficientemente grande, puedo ver si las métricas que uso son estables, eso me dice que mi modelo es bastante estable. Se debe ver las métricas, el promedio de las mismas, el desvío standard, etc. Se debe ver que no hay mucha variación de la métrica para casa modelo. Esto lo hago en cada moldeo que quiero probar.

* ¿Cómo ayuda esto a encontrar el mejor modelo (sin overfitting)?
* La validación cruzada de k-fold se usa ampliamente para verificar si un modelo está overfitted o no.
* Si las métricas en cada uno de los k modelados están cerca entre sí y la media de la
* métrica es la más alta estás bien encaminado
* En una competencia de Kaggle, puede confiar más en CV en la puntuación pública de Kaggle.

### Confiuración K-Fold Cross Validation

* Recomendación: empezar con algo básico y simple y ver cómo funciona
* El valor de k debe elegirse cuidadosamente para su muestra de datos.
* Representativo: el valor de k se elige de manera que cada grupo de pruebas / tren de muestras de datos sea lo suficientemente grande como para ser estadísticamente representativo del conjunto de datos más amplio.
* k = 10: El valor de k se fija en 10, un valor que se ha encontrado mediante
* experimentación que generalmente da como resultado una estimación de la habilidad del modelo con un sesgo bajo y una varianza modesta.
* k = n : el valor de k se fija en n, donde n es el tamaño del conjunto de datos y de este modo se les da a cada muestra la oportunidad de ser utilizada en el conjunto de datos que no se utilizan. Este enfoque se denomina Leave One Out Cross Validation.



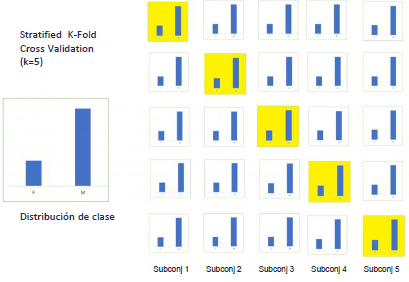
* La elección de k suele ser 5 o 10, pero no existe una regla formal. A medida que k aumenta, la diferencia de tamaño entre el conjunto de entrenamiento y los subconjuntos de remuestreo se reduce. A medida que esta diferencia disminuye, el sesgo de la técnica se vuelve más pequeño
* k = 10 es usado muy comúnmente y se recomienda si tiene dificultades para elegir un valor para su conjunto de datos.
* Un enfoque consiste en explorar el efecto de diferentes valores de k en la estimación del rendimiento del modelo y compararlo con una condición de prueba ideal. Esto puede ayudar a elegir un valor apropiado para k.
* Una vez que se elige un valor k, se puede usar para evaluar un conjunto de algoritmos diferentes en el conjunto de datos y la distribución de resultados se puede comparar con una evaluación de los mismos algoritmos usando una condición de prueba ideal para ver si están altamente correlacionados o no. Si está correlacionado, confirma que la configuración elegida es una aproximación robusta para la condición de prueba ideal.

### Notebook

4ta Clase - Cross Validation

### Stratified K-fold Cross Validation

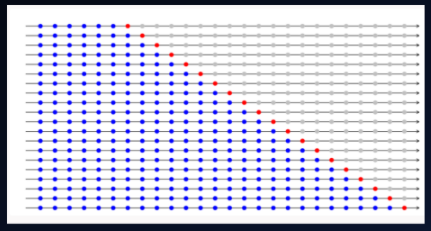
* Para balancear un clase se pueden generar más datos de la clase minoritaria o podemos sacar datos de la clase mayoritaria y así lo podemos balancear, estar es una forma.
* Intenta resolver algunos problemas del enfoque K-Fold.
* En K fold, primero mezclamos aleatoriamente los datos y luego los dividimos en grupos (folds). En algunos casos existe la posibilidad de que obtengamos pliegues altamente desequilibrados que pueden causar que nuestro modelo esté sesgado hacia una clase en particular.
* Por ejemplo, obtengamos de alguna manera un grupo que contenga la mayoría de las muestras de la clase positiva y solo unas pocas muestras de la clase negativa. Esto sin duda afectará a nuestro entrenamiento y para evitarlo hacemos los folds estratificados mediante el proceso de estratificación.
* Estratificación Es el proceso de reorganizar los datos de manera que cada uno de los grupos sea un buen representante de todo el conjunto de datos de diferentes clases.



* Mezclar los datos logrando que la clase quede balanceada de la misma forma como en la población original.

### Cross Validation series de tiempo

* Esto es una materia en misma
* Validación cruzada para series de tiempo
* Dividir un conjunto de datos de series de tiempo de forma aleatoria no funciona porque los subconjuntos de tiempo de los datos estarán desordenada. Para un problema de pronóstico de series de tiempo, realizamos la validación cruzada de la siguiente manera.
* Los subconjuntos para la validación cruzada de series de tiempo se crean en forma de encadenamiento hacia adelante
* Suponga que tenemos una serie de tiempo para la demanda anual del consumidor de un producto durante un período de n años.
* Los subconjuntos se crearían de este modo:



* Hay problemas de auto relación, los datos que vienen antes, van a modificar los datos que vienen después
* Hay transformaciones, de Furier, descomponen sus componentes en una curva en lugar de medir todos los puntos, medís las propiedades de la curva
* Serie de tiempo hay que ver todos los conceptos de nuevo. Los resúmenes del banco de los clientes en el tiempo es una serie de tiempo.
* En cross validation: vas tomando un período X de tiempo que a saber tiene cierta estabilidad. Tomas todos hasta esta fecha y vas prediciendo la próxima fecha.
* Seleccionamos progresivamente un nuevo conjunto de entrenamiento y testeo.
* Comenzamos con un conjunto de training que tiene un número mínimo de observaciones necesarias para ajustar el modelo.
* Progresivamente, cambiamos nuestro training y testing prueba con cada subconjunto.
* En la mayoría de los casos, los pronósticos de 1 paso pueden no ser muy importantes.
* En tales casos, el origen del pronóstico se puede cambiar para permitir el uso de errores de varios pasos.

### Notebooks

* 4ta Clase - Differentes CV
* 4ta Clase - Serie de tiempos

# 5ta Clase – 08/07/2021

## Indicadores clave de rendimiento o key performance indicators (KPI)

Las métricas son para medir cuan bueno es el modelo. Como traducimos ese modelo a la vida real no siempre es fácil. Los indicadores de rendimiento son para la medir utilidad del modelo.

### Indicadores KPI

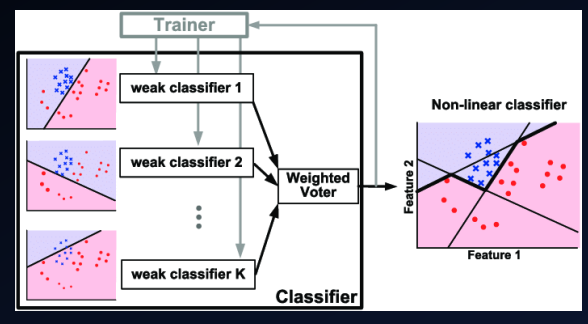
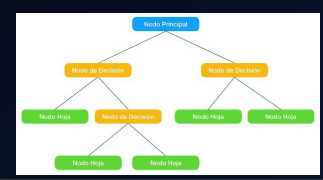
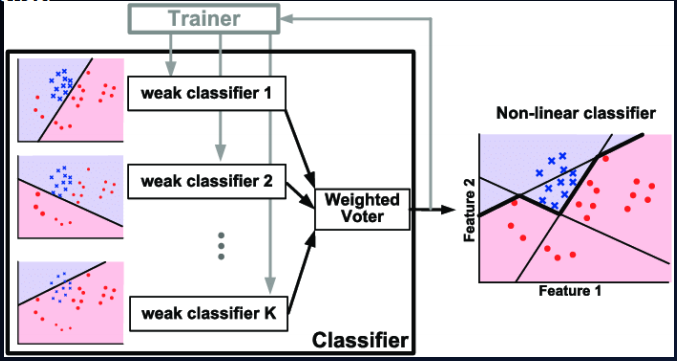
*Hay que tratar de traducir, entender como mejorar algo del negocio. Ej. Performance, incremento en publicidades de marketing, lograr el mayor retorno de inversión (ROI)*

*Ver si las métricas que tenemos representan lo que quiero medir. (Las métricas vistas hasta ahora hablan de cuan bueno es el modelo en sí, ahora debemos analizar la alternativa que sea más rentable)*

### Relación entre KPI y métricas

* El aumento del rendimiento de un modelo no siempre está asociado con un crecimiento o mejoría empresarial.
* **El seguimiento y la correlación de las métricas del modelo con los KPI** son fundamentales para acortar la brecha entre el análisis del rendimiento de los modelos que se implementan y el crecimiento empresarial.
* En este sentido el trabajo coordinado debería redundar en mejorar las chances de lograr una alineación con los objetivos técnicos de análisis de datos y los objetivos comerciales de la empresa.
* Por lo tanto es esencial interpretar las métricas y mejoras que se realizan a los modelos de aprendizaje automático en el contexto de los distintos KPI, esto ayuda a cuantificar qué factores afectan el crecimiento del negocio y por lo tanto orientan a los expertos en datos cómo ajustar el modelo para que efectivamente pueda ser útil en coordinar y dirigir un crecimiento empresarial
* La performance de un modelo proporciona una medida de que tan bien funciona en predecir eventos.
* Como vimos existen una gran variedad de Métricas y mecanismos que nos permiten evaluar la capacidad del modelo para realizar una predicción.
* De este modo también podemos detectar desvíos, sesgos, e inconsistencia de datos entre otros.
* A la detección de los errores le sigue la mitigación de estos errores para garantizar que el modelo implementado esté realizando predicciones precisas y resulte lo más resistente posible a las fluctuaciones de datos.
* Si bien en el ámbito académico la mejora en el rendimiento del modelo es la medida de suceso del diseño e implementación de un modelo en el caso del mundo empresarial la situación es bien distinta.
* En este sentido es importante recordar que el mundo empresarial no se preocupa por una alta **Accuracy** o **AUC-ROC, mayor explicabilidad** de los modelos, pero si se preocupa por los indicadores clave de **rendimiento (KPI)**.
* **Los KPI son medidas cuantificables de factores que afectan el rendimiento de una empresa.**
* Existen varios KPI que se desprenden de conceptos como crecimiento de los ingresos (revenue growth), rentabilidad, planificación estratégica en ventas y marketing, etc.
* Estos indicadores dan una visión detallada de qué tan bien se está desempeñando un área en particular o toda la organización y debería permitir generar herramientas analíticas para la toma de decisiones.
* Las empresas te pagan por la performance, por cuan bueno es el modelo.

#### Ejemplo

* El lado comercial del Machine Learning requiere que el modelo integre el doble objetivo de lograr un rendimiento aceptable y cumplir con metas empresariales.
* Escenario muy simple: una nueva empresa emergente ofrecer análisis predictivos basados en Machine Learning.
* En particular tiene que generar clasificadores binarios y para ello usan un modelo de clasificación basado en árboles de decisiones con una técnica de Ensamble (**AdaBoost**).
* La técnica de Ensamble (**AdaBoost**): Son muchos árboles de decisión que tienen poder predictivo muy bajo, luego de ejecutarlos por separado, los combina y obtiene información más completa. Cada árbol hace una clasificación y luego veo para cada punto, como fue clasificado por cada árbol (algoritmo). La salida se podría combinar con “Randon Forest”.
* *Varios árboles de decisión con técnica de ensamble. Dataset con clase binaria generan un modelo (AdaBoost) con mayor accuracy .*
* Cuanto mayor sea la Accuracy de su predicción, mayores serán los ingresos. (Más adelante contradice esto???)
* Haciendo una predicción aleatoria (asumiendo una clase balanceada) uno puede obtener un 50% de predicción correcta.
* Por lo tanto, sólo se les paga por una mayor Accuracy por encima de un cierto umbral.
* Entonces con un umbral de 50% si tiene un modelo con 75% de Accuracy se les paga un ingreso proporcional al 75% - 50% = 25%.
* Obviamente se desea obtener la Accuracy más alta para maximizar la ganancia
* Cuanto mejor performance tenga, tendré más ganancia. Una vez que mi modelo funciona bien, para mejor la performance puedo tratar de optimizar los hiperparámetros. Un hiperparámetro de “Adaboost” es por ejemplo la cantidad de arboles de decisión que voy a usar.
* Para ello se pueden ajustar los hiperparámetros del algoritmo.
* Para AdaBoost se pueden ajustar distintos hiperparámetros, principalmente relacionados con el estimador base que usa (arboles de decisión).
* Para un **árbol de decisión**, estos pueden ser el número mínimo de muestras por hoja (tamaño mínimo de nodo, que regula la creación de nuevas divisiones. Un nodo no se dividirá si el número de puntos de datos que contiene está por debajo del tamaño mínimo del nodo), la profundidad máxima del árbol, el criterio de división como el índice de Gini, etc. <http://www.r2d3.us/visual-intro-to-machine-learning-part-1/> (Ver contenido en sección **Apéndice**)
* *Para lograr un mayor accuracy puedo tocar un hiperparámetro (son los parámetros que se ajustan por afuera). Ej., cuantos árboles de decisión voy a poner.*
* *A mayor número de árboles (nodos)*🡪 *mayor tiempo de proceso*
* *Analizar: cuánta más ganancia me da si subo más árboles …*
* *En este punto hay que definir si me conviene agregar más árboles en función a la performance.*
* Para hacerlo más fácil elegimos el hiperparámetro más intuitivo: el número de estimadores de árboles utilizados.
* En esencia, las técnicas de conjunto como **Boosting (esto sería Adaboost)** funcionan manteniendo el estimador base relativamente simple y con baja precisión (un poco por encima del 50% está bien). Estos modelos logran un poder de generalización robusto al emplear una gran cantidad de estimadores de base simples en paralelo y promediar sus predicciones
* Por lo tanto, a mayor número de estimadores base (arboles de decisión) se obtendrá un mayor poder de generalizar los resultados a datos no conocidos y por lo tanto aumentar la Accuracy.
* Anteriormente, establecimos el escenario simple de que los ingresos son directamente proporcionales al nivel de Accuracy.
* La estrategia más simple para maximizar la Accuracy del modelo y, por lo tanto, maximizar los ingresos de la empresa, es mantener los estimadores base individuales realmente simples, eligiendo la profundidad máxima de los árboles algo así como 2 o 3, y empleando un gran número de ellos.

### Diferencia entre Ingresos y Ganancia

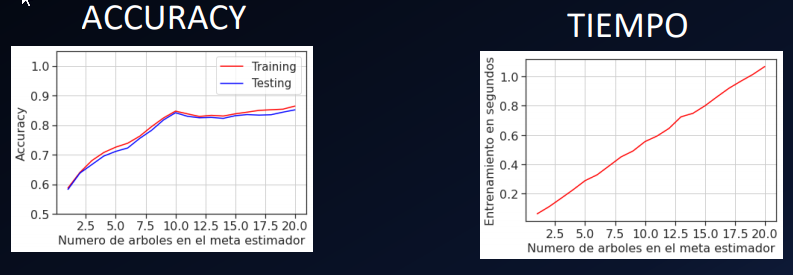
* Los ingresos no siempre son equivalentes a las ganancias y en el ámbito comercial el objetivo es maximizar las ganancias.
* Siempre se desea obtener ingresos considerables con un bajo costo operativo.
* Hasta este momento estaba claro cuál es la relación entre la mejora de las métricas del modelo y el ingreso obtenido (Mayor Accuracy == Mayor Inrgesos), pero, ¿**Cuál es la relación entre el costo operativo y el modelo?**
* No siempre es fácil entenderlo (por eso buscamos un ejemplo simple) y en nuestro caso podemos suponer que el costo es proporcional al tiempo total de cálculo para el ajuste y la predicción del modelo.
* Esto es un escenario bastante realista para una empresa pequeña que tiene que alquilar algún tipo de servicio en la nube para ejecutar sus algoritmos, por ejemplo AWS EC2, Google Cloud, etc. etc., que se factura en función del tiempo total de procesamiento.
* **En este sentido cuanto mayor es el número de estimadores base (árboles de decisión), mayor será la carga computacional del algoritmo y mayor el tiempo de cómputo para el ajuste y la predicción del modelo.**
* Esto nos permite identificar la relación entre un hiperparámetro de modelo: número de estimadores base y dos métricas comerciales: ingresos y costos.
* ¿Y qué es la ganancia? 🡪 **Ganancia = Ingresos – costo**
* Entonces estamos en un escenario distinto del que planteamos hasta ahora (pero tal vez uno mucho más real!)
* Es decir aquí no sólo importa el trade off Sesgo/Varianza, la búsqueda de hiperparámetros para determinar la mejor performance, cual es la métrica correcta:¿Accuracy / Precisión / Recall? F1-Score ? ¿Curva ROC y AUC? Etc.
* Todo esto sigue siendo de importancia crítica, pero….
* Desde una perspectiva empresarial probablemente lo único que importe es que ganancia neta que genera nuestro modelo.



* Cuando mejoro el Accuracy, mejoro la performance del modelo, pero tengo que ver cuanto más me van a costar estas mejoras

### Optimización

* Este ejemplo es muy simple, pero muestra algo práctico donde la elección de un modelo está fuertemente asociada a una métrica comercial.
* Por lo tanto en este caso hay que buscar algún modo de encontrar la configuración óptima de los parámetros del modelo que maximicen la métrica empresarial, en este caso la ganancia.
* El valor de la función objetivo se calculará teniendo en cuenta tanto la Accuracy del algoritmo en el conjunto de testing y el costo que será proporcional al tiempo que lleva el entrenamiento y predicción del modelo.
* El tiempo tiene que ver con el costo que tendré con google o Amazon para hacer el procesamiento de mi modelo.



Training es el accurary de los datos de training (acc\_train=accuracy\_score(y\_train,pred\_train) #accuracy en training)

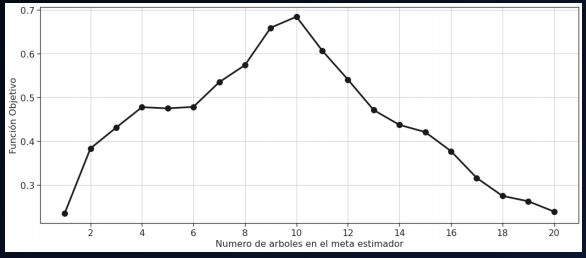
Test es el accurary de los datos de test (acc\_test=accuracy\_score(y\_test,pred\_test) #accuracy en testing)

* *No solo me voy a quedar con el mayor Accuracy sino con el que más rentabilidad me deja (costo en función del tiempo)*
* **Accuracy** comienza en un valor bajo para un pequeño número de estimadores, pero se satura después de que ese número alcanza un cierto nivel. Por otro lado, la carga computacional sigue aumentando.
* Por lo tanto, no tiene sentido seguir aumentando el número de estimadores a medida que la tasa de rendimiento (en términos de mejorar la Accuracy) alcanza un máximo en un cierto nivel y luego desciende.
* *Llega un punto que por más que agregue más árboles, no aumenta el accruracy pero si el tiempo, por lo cual la decisión se da ahí, eliminando los que no me generan más accuracy pero si mayor tiempo de procesamiento y hace que baje la performance (e incremente el costo)*
* *IMPORTANTE: ESTA DECISIÓN SIEMPRE DEPENDE DEL PROBLEMA QUE ANALIZO*
* Creamos una función objetivo (TOOOOOOOODO un TEMA!!!!!), que integra tanto la Accuracy como el costo de cálculo en una único valor, combinado la Accuracy del conjunto de testing y el tiempo de cálculo en una función lineal con factores que permite ponderarlos en modos distintos.
* Función objetivo = [α\*(Accuracy − Umbral Acc) – β\*tiempo de computo]

(Aquí sino divido por los máximos(es decir si solo tengo 1 árbol), entonces estoy restando cosas distintas: Accuracy menos tiempo de cómputos) Rta: esta función se normaliza acomodando el alfa y el beta, los vas moviendo para normalizar el la función, esta es un ejemplo pedagógico que armó Mario para explicarnos el tema. Ver en Notebbok: “5ta Clase - Funcion\_Ganancia”, sección: “Construyo una función que puedo minimizar usando **SciPy”**



* Ver Notebook: 5ta Clase - Funcion\_Ganancia
* Alfa y beta, nos permiten ponderar de forma distinta a los parámetros
* Este gráfico es una combinación de los dos gráficos anteriores



* Eje X: Cantidad de árboles, eje Y: *Función objetivo*
* La función objetivo es compleja de generar y depende del caso que se analice.

### Notebook

5ta Clase - Funcion\_Ganancia

## Análisis de churn (KPI)

El análisis de Chrurn se hace para predecir los clientes que se darán de baja en un dominio dado, por ejemplo una cuenta bancaria, una prepaga de salud, etc.

### Churn o cancelación

* Retener clientes es uno de los principales indicadores clave de rendimiento para las empresas.
* La competencia es difícil, especialmente en los mercados donde los clientes pueden elegir entre una gran cantidad de proveedores.
* Una mala experiencia y el cliente puede simplemente pasar al competidor, lo que resulta en una pérdida de clientes.
* El churn o perdida/cancelación de clientes es el porcentaje de clientes que dejaron de usar el producto o servicio de una determinada empresa durante un período de tiempo determinado.
* Una de las formas de calcular una tasa de churn es dividir la cantidad de clientes perdidos durante un intervalo de tiempo determinado por la cantidad de clientes activos al comienzo del período. Por ejemplo, si consiguió 1000 clientes y perdió 50 el mes pasado, su tasa de abandono mensual es del 5 por ciento.
* Predecir la pérdida de clientes es un problema empresarial extremadamente importante, especialmente en industrias donde el costo de adquisición de clientes es alto (tecnología, telecomunicaciones, finanzas, etc.).
* La capacidad de predecir que un cliente posee un alto riesgo de abandonar, mientras aún hay tiempo para hacer algo al respecto, representa una enorme fuente de ingresos potencial adicional para las empresas.

### Churn Machine learning

* El objetivo principal del modelo predictivo de churn de clientes es retener a los clientes con mayor riesgo de abandono al interactuar proactivamente con ellos
* Por ejemplo: ofrezca un voucher de regalo o cualquier precio promocional y fidelizarlos durante uno o dos años más para extender su valor de por vida a la empresa
* Hay dos conceptos importantes:

1. Queremos un modelo predictivo para predecir el churn por adelantado (digamos con un mes de anticipación, tres meses de anticipación o incluso seis meses de anticipación; todo depende del caso de uso)

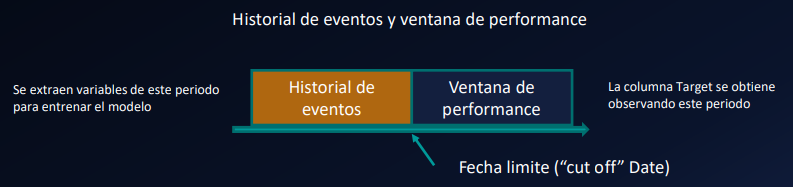
Esto significa que debe tener mucho cuidado con la fecha límite, es decir, no se debería utilizar ninguna información después de la fecha límite en el modelo de aprendizaje automático. El período anterior a la fecha límite se conoce como Evento

1. Normalmente, para la predicción de churn de clientes, generar la variable target requiere cierto trabajo y generalmente no está disponible en la forma que le gustaría

Por ejemplo, si desea predecir si el cliente abandonará en el próximo trimestre, se iterará a través de todos los clientes activos a partir de la fecha de cierre del evento y comprobará si dejaron la empresa en el próximo trimestre o no (1 para sí, 0 para no)

El trimestre en este caso se llama Ventana de tiempo a.

* Cuanto antes se puede predecir, más cosas se pueden hacer para retener el cliente.
* Ejemplo de distribución de clientes: 1% de los pacientes, gastan el 30% del presupuesto. Esto aplica en cualquier ámbito???



### Churn WorkFlow

* ¿Cómo se utiliza este modelo de aprendizaje automático en la empresa?
* El modelo se entrena en el historial de churn de clientes. Cada mes, la base de clientes activa se transfiere al modelo predictivo de aprendizaje automático para devolver la probabilidad de abandono de cada cliente
* La lista se ordenará según el valor de probabilidad más alto al más bajo y los equipos de retención de clientes comenzarán a interactuar con el cliente para detener la deserción
* Los clientes que tienen una probabilidad muy baja de abandono son clientes satisfechos. Y sobre ellos no se toman medidas

Pregunta Romina: ¿Se podría hacer un ejercicio con algoritmos no supervisados?. Rta: Si.

Luego se podrían comparar los resultados con el proceso de Churn. Esto aplica si son muchos datos, sino no se pueden aplicar algoritmos no supervisado.

El historial de Churn de los clientes es cuando le doy valor a mi variable target.

Ejemplo:

De enero a julio defino las variables que me pueden servir, que van cambiando en el tiempo, por ejemplo el saldo, puede ser el mínimo o máximo o promedio.

Defino la matriz con:

Fila de observaciones y columnas variables (las miro de enero a julio)

Luego la columna para la clase: de julio a septiembre veo que cliente se fue y quien no se fue. En este período no miro las otras variables.

Luego del período en el cual defino la clase, puedo correr el algoritmo

En general no puedo predecir con mucho tiempo de anticipación, en general si alguien decide irse lo hace de un mes para el otro. Si predigo con mucho tiempo de anticipación y la persona no se quiere ir, entonces estoy metiendo ruido. Mientras voy corriendo el algoritmo, voy encontrando la ventana temporal adecuada.

### Notebook

5ta Clase - Churn

## Apéndice

### Parte del contenido de: [www.r2d3.us](http://www.r2d3.us)

http://www.r2d3.us/visual-intro-to-machine-learning-part-1/

http://www.r2d3.us/visual-intro-to-machine-learning-part-2/

*(Conviene mirar el ejemplo (de las casas) que se tiene en el sitio.)*

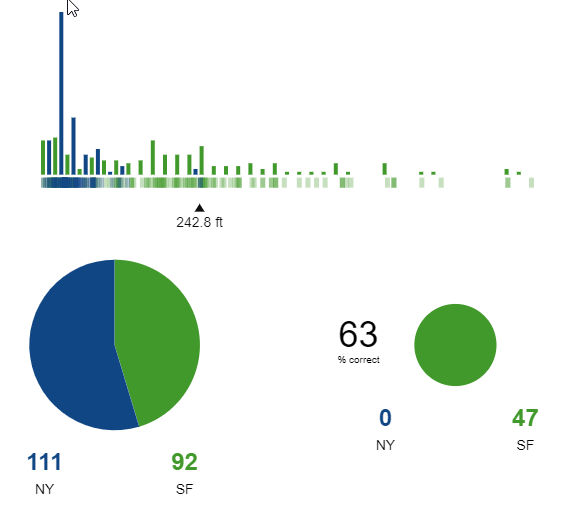
#### APRENDIZAJE AUTOMÁTICO-PARTE I

##### MACHINE LEARNING

*La búsqueda de patrones en los datos es lo que hace el aprendizaje automático. Los métodos de aprendizaje automático utilizan el aprendizaje estadístico para identificar límites.*

*Un ejemplo de método de aprendizaje automático es el árbol de decisión. Los árboles de decisión analizan una variable cada vez y son un método de aprendizaje automático razonablemente accesible (aunque rudimentario).*

***Encontrar mejores límites 🡪*** *es buena práctica y ayuda a mejorar la intuición, plasmar en un histograma todas las variables para ver su frecuencia. (recordar análisis de las casas y sus variables)*



##### FORK:

*Un* ***árbol de decisión*** *utiliza declaraciones* ***if-then*** *para definir patrones en los datos.*

*Por ejemplo, si la elevación de una casa está por encima de un número, entonces es probable que la casa esté en San Francisco.*

*En el aprendizaje automático, estas afirmaciones se denominan bifurcaciones y dividen los datos en dos ramas en función de algún valor.*

*Ese valor entre las ramas se llama punto de división. Las viviendas situadas a la izquierda de ese punto se clasifican de una manera, mientras que las situadas a la derecha se clasifican de otra. Un punto de división es la versión del árbol de decisión de un límite.*

##### TRADE-OFFS (Contrapartidas)

*La elección de un punto de división tiene contrapartidas. Ejemplo, habíamos definido por ej que si una casa tenia mas de 73 m, era de NY. Al determinar como nuestra división inicial (~73 m) clasifica incorrectamente algunas viviendas de San Francisco (con más de 73m) como de Nueva York. --> estas viviendas de San Francisco clasificadas erróneamente.* ***Son los llamados falsos negativos.***

*Sin embargo, un punto de división destinado a capturar todas las viviendas de San Francisco incluirá también muchas viviendas de Nueva York. Estos son los llamados* ***falsos positivos****.*

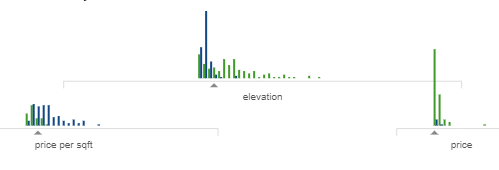
##### The Best Split (La major division)

*En la mejor división, los resultados de cada rama deben ser lo más homogéneos (o puros) posible. Hay varios métodos matemáticos entre los que se puede elegir para calcular la mejor división. Como vemos, incluso la mejor división en una sola característica no separa completamente las viviendas de San Francisco de las de Nueva York.*

##### Recursión

*Para añadir otro punto de división, el algoritmo repite el proceso anterior en los subconjuntos de datos. Esta repetición se denomina recursión, y es un concepto que aparece con frecuencia en los modelos de entrenamiento. Los histogramas de la izquierda muestran la distribución de cada subconjunto, repetida para cada variable. La mejor división variará en función de la rama del árbol que se esté mirando.*

*(Para las viviendas de menor altura, el precio por pie cuadrado es, con 1061 dólares por pie cuadrado, la mejor variable para la siguiente sentencia if-then. En el caso de las viviendas de mayor altura, el precio, de 514.500 dólares, es la mejor variable para la siguiente sentencia if-then).*



##### Crecimiento de un árbol

*Las bifurcaciones adicionales añaden nueva información que puede aumentar la precisión de la predicción de un árbol. Dividiendo una capa más, la precisión del árbol mejora hasta el 84%.*

*Añadiendo varias capas más, se llega al 96%.*

*Incluso se pueden seguir añadiendo ramas hasta que las predicciones del árbol sean 100% precisas, de modo que al final de cada rama, las viviendas estén puramente en San Francisco o puramente en Nueva York.*

*Estas ramas finales del árbol se denominan nodos hoja. Nuestros modelos de árbol de decisión clasificarán las viviendas de cada nodo de hoja según la clase de viviendas que sea mayoritaria.*



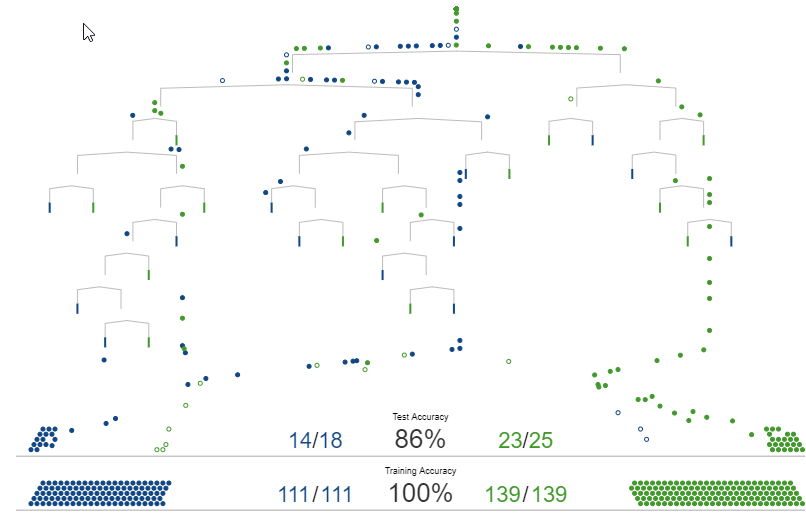
##### Hacer predicciones

*El modelo de árbol de decisión recién entrenado determina si una vivienda está en San Francisco o en Nueva York haciendo pasar cada punto de datos por las ramas.*

*Aquí puede ver los datos que se utilizaron para entrenar el árbol fluyendo a través del mismo.*

*Estos datos se denominan datos de entrenamiento porque se utilizaron para entrenar el modelo.*

*Como hemos hecho crecer el árbol hasta que era 100% preciso, este árbol asigna perfectamente cada punto de datos de entrenamiento a la ciudad en la que se encuentra.*



##### Comprobación de la realidad

*Por supuesto, lo que más importa es cómo se comporta el árbol con los datos no vistos anteriormente.*

*Para comprobar el rendimiento del árbol con datos nuevos, tenemos que aplicarlo a puntos de datos que nunca ha visto antes. Estos datos no utilizados anteriormente se denominan datos de prueba.*

*Lo ideal sería que el árbol tuviera un rendimiento similar tanto con datos conocidos como con datos desconocidos.*

*Estos errores se deben al sobreajuste. Nuestro modelo ha aprendido a tratar como importantes todos los detalles de los datos de entrenamiento, incluso los que resultaron ser irrelevantes.*

*El sobreajuste forma parte de un concepto fundamental en el aprendizaje automático que explicaremos en nuestro próximo post.*

##### Recapitulación

*El aprendizaje automático identifica patrones utilizando el aprendizaje estadístico y los ordenadores desenterrando límites en los conjuntos de datos. Se puede utilizar para hacer predicciones.*

*Un método para hacer predicciones se denomina árbol de decisión, que utiliza una serie de sentencias if-then para identificar límites y definir patrones en los datos.*

*La* ***sobreadaptación*** *se produce cuando algunos límites se basan en distinciones que no marcan la diferencia. Puede ver si un modelo se* ***sobreajusta*** *haciendo que los datos de prueba fluyan a través del modelo.*

*A continuación*

*Ver sobreadaptación y cómo se relaciona con un compromiso fundamental en el aprendizaje automático. Consulte la Parte II: Ajuste del modelo y la compensación entre sesgo y varianza.*

#### APRENDIZAJE AUTOMÁTICO-PARTE II

***Ajuste de modelos y el equilibrio entre el sesgo y la varianza***

*El objetivo del modelado es aproximarse a situaciones de la vida real identificando y codificando patrones en los datos. Los modelos cometen errores si esos patrones son demasiado simples o demasiado complejos.*

*En la primera parte, creamos un modelo que distingue las viviendas de San Francisco de las de Nueva York. Ahora hablaremos de la* ***sintonía y del equilibrio entre sesgo y varianza****.*

##### Parámetros del modelo

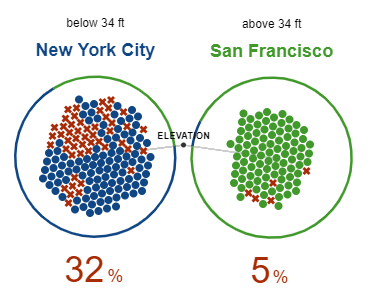
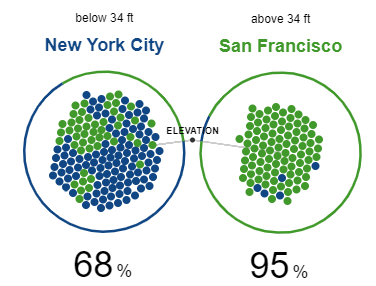
*Los modelos pueden ajustarse para modificar la forma en que se ajustan a los datos. Estos* ***"ajustes"*** *se denominan* ***parámetros****. Un ejemplo de parámetro del árbol de decisión es el tamaño mínimo del nodo, que regula la creación de nuevas divisiones. Un nodo no se dividirá si el número de puntos de datos que contiene es inferior al tamaño mínimo del nodo.*

*El árbol de la primera parte tenía un tamaño mínimo de nodo de uno. Era muy complejo, tenía muchas divisiones y se ajustaba demasiado a los datos. Para ver por qué, revisemos cómo se entrenó el árbol de decisión.*

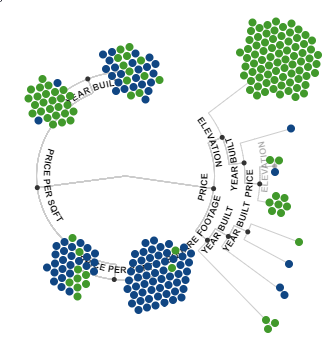
##### Demasiado simple

*La versión más sencilla de un árbol de decisión se denomina tronco. Compuesto por una única división, se componen de una única regla, como "Toda casa cuya elevación sea superior a 34 pies está en San Francisco, y todas las demás están en Nueva York".*

*Se adoptan una visión binaria del mundo e ignoran la complejidad y los matices de los datos de entrenamiento. Esta interpretación en blanco y negro del mundo es propensa a los errores debidos al sesgo.*

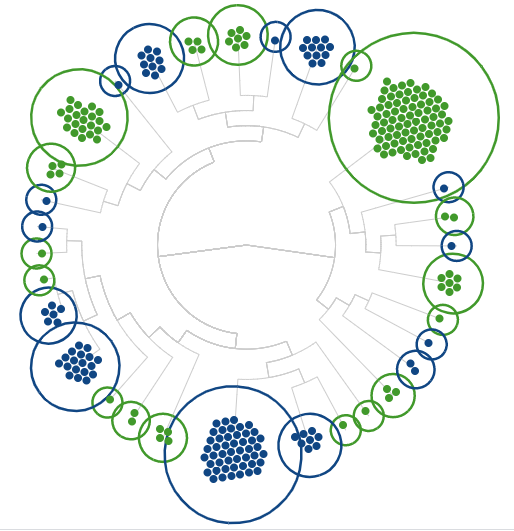
*Un modelo con demasiado sesgo ignora sistemáticamente los detalles relevantes y se equivoca de forma sistemática. El muñón de la derecha clasifica incorrectamente todas las viviendas de menor altitud de SF.* 

*Para disminuir el error debido al sesgo, se pueden añadir divisiones adicionales al árbol.*



##### Demasiado complejo

*Las divisiones adicionales permiten que el árbol tenga en cuenta una mayor complejidad. Puede añadir divisiones hasta que los nodos de las hojas de un árbol sólo contengan viviendas en San Francisco o Nueva York.*



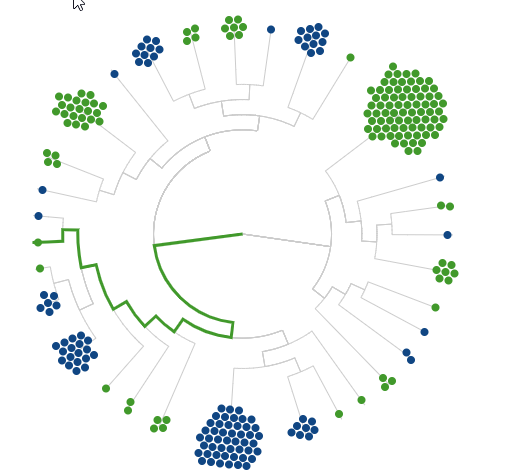
*La pregunta es: ¿cómo se comporta con los datos de prueba?*

*Como observamos en la primera parte, este árbol completo no es tan preciso en los datos de prueba. Por ejemplo, con este árbol la tasa de error es del 12%.*

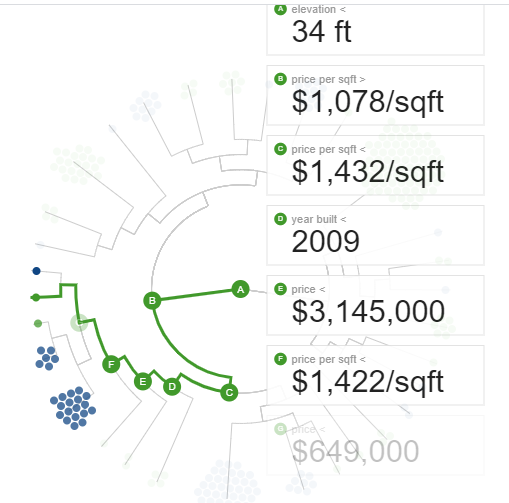
*Este árbol excesivamente complejo sufre errores debidos a la varianza.*

*Los modelos de alta varianza cometen errores al ajustarse en exceso a la idiosincrasia de los datos de entrenamiento. Tienden a equivocarse de forma incoherente.*

*Para ver exactamente lo que ocurre, volvamos al conjunto de entrenamiento y sigamos la creación de un único nodo hoja.*

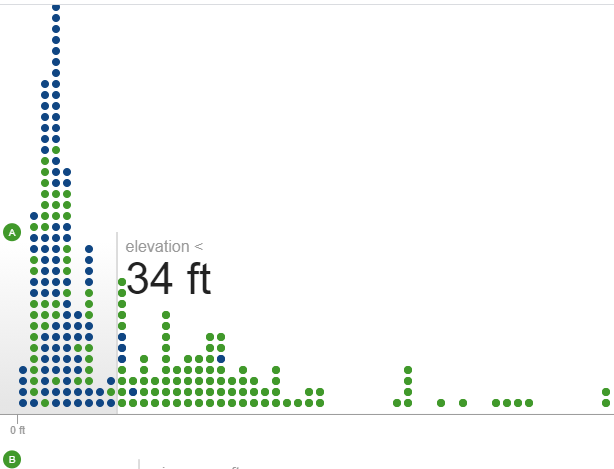


##### Ejemplo tangible de la varianza

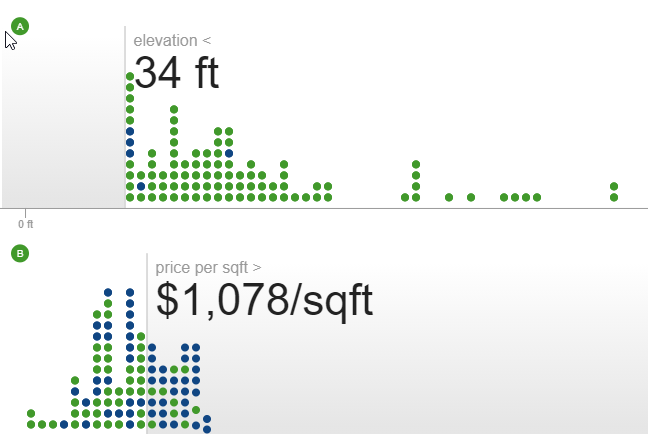


*Este nodo hoja es el resultado de ocho bifurcaciones distintas. Cada bifurcación divide el conjunto de datos en subconjuntos más pequeños, hasta que el nodo hoja contiene un único hogar de San Francisco.*

*Para ver cómo la división repetida conduce a errores de varianza, mostraremos la serie de bifurcaciones como gráficos de puntos de distribución.*



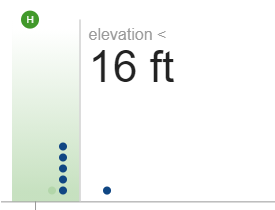
*En cada bifurcación, el punto de división se establece de forma que las ramas resultantes sean lo más homogéneas posible.*



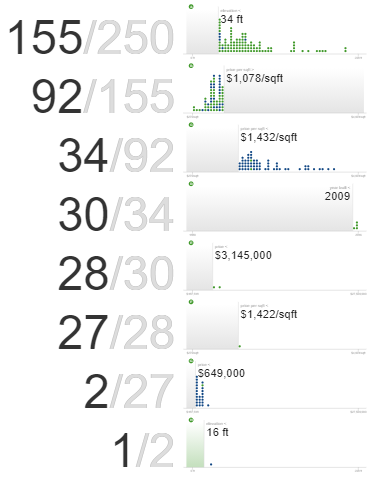
*Cada punto de bifurcación se selecciona con avidez, en lugar de tener en cuenta lo que podría ser mejor más adelante. Las bifurcaciones más tempranas en el árbol pueden tener un efecto en cascada más profundo en el árbol.*

*...En un caso extremo (como éste), la división final se basa en sólo dos puntos de datos bastante arbitrarios.*

*La arbitrariedad de la división se refleja en la precisión del modelo. Si se introducen los datos de prueba en el árbol, cinco viviendas cumplen las normas en la rama. El modelo cree que estas casas deberían estar en San Francisco.*

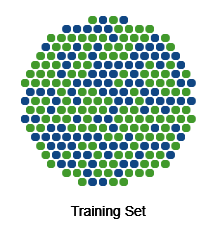
 *...pero las cinco están en Nueva York.*

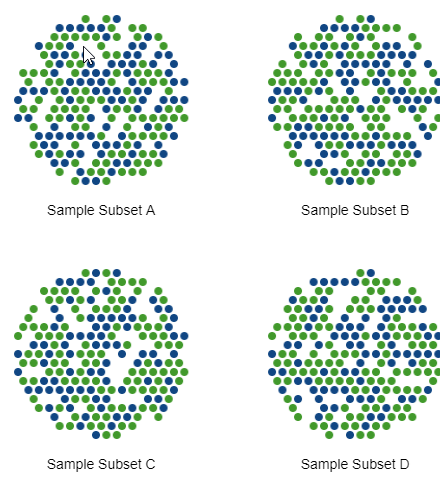
*Las bifurcaciones finales se hicieron utilizando muy pocos datos, por lo que no es de extrañar que las generalizaciones que hacen sean incorrectas. Los patrones extraídos de dos hogares tienen más probabilidades de ser casualidades que algo real.*



##### Las fallas son normales

*Este no es un incidente aislado. Por ejemplo, podríamos hacer crecer árboles adicionales a partir de subconjuntos aleatorios de los datos de entrenamiento.*

 *Del conjunto de entrenamiento de 250 casas... vamos a dibujar cuatro muestras aleatorias de 200 viviendas cada una y a cultivar árboles basados en ellas.*

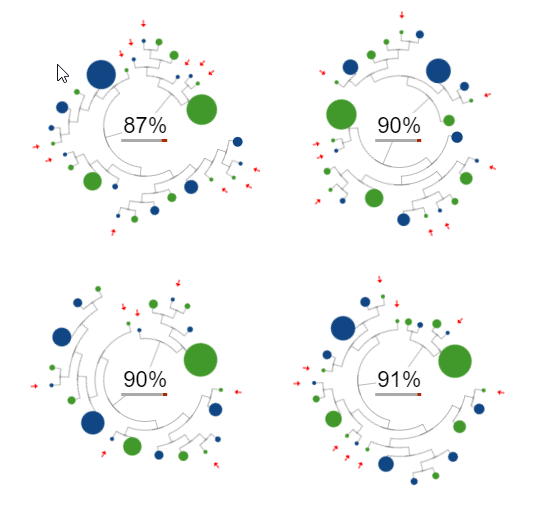


*Los árboles resultantes tienen todos un aspecto razonablemente diferente y también tienen nodos de hoja de una sola vivienda.*

*Estas viviendas aparentemente esotéricas que pueden dar lugar a nodos de hoja de un solo punto de datos son en realidad una parte normal de cualquier conjunto de datos. Son un resultado del método de ajuste del modelo.*

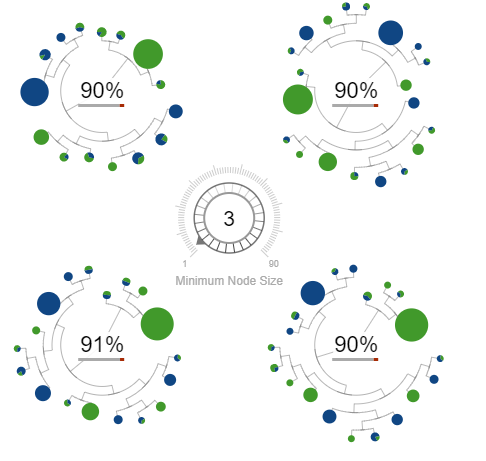
*Cuando el parámetro de tamaño de nodo mínimo es uno, el árbol crece hasta que cada rama tiene un nodo hoja homogéneo.*

*Para un conjunto de datos determinado, el crecimiento del árbol en un conjunto diferente de hogares cambia lo que las ramas sobreajustan, pero el sobreajuste sigue ocurriendo. Los modelos que se sobreajustan son inestables y sensibles a los pequeños cambios en los datos de entrenamiento y, por tanto, de alta varianza.*

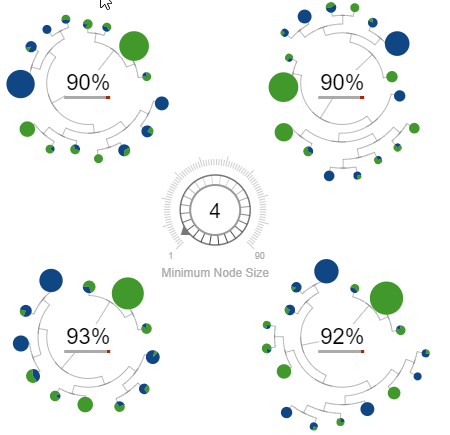


##### El equilibrio entre el sesgo y la variación

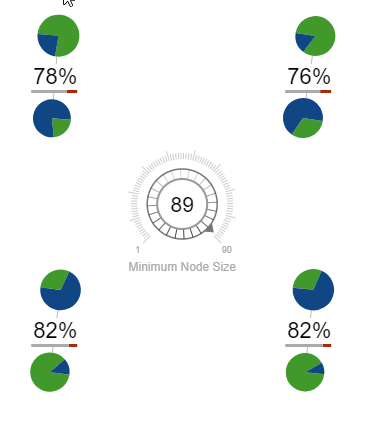
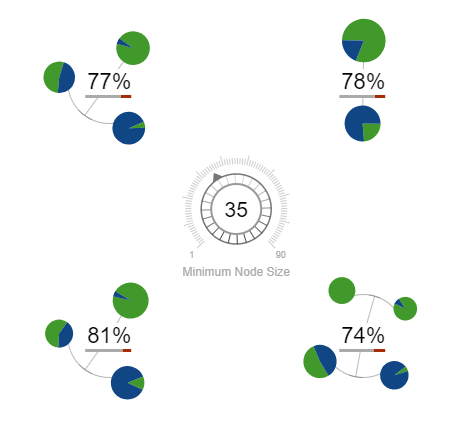
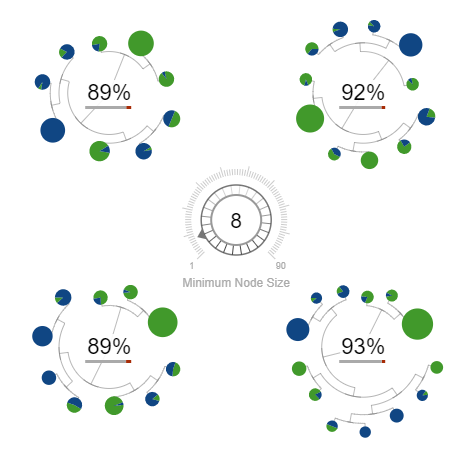
*Una forma de abordar los errores derivados del sobreajuste es imponer límites al crecimiento del árbol cambiando el umbral de tamaño de nodo mínimo.*



*A medida que el umbral de tamaño de nodo mínimo aumenta, hay menos divisiones. Los árboles se vuelven menos tupidos.*



*La precisión de cada árbol mejora al disminuir los errores debidos a la varianza.*



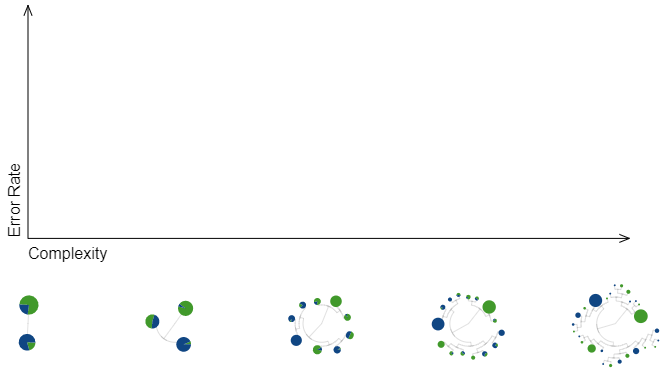
*A medida que el umbral de tamaño de nodo mínimo sigue aumentando, la precisión empieza a deteriorarse por el error debido al sesgo.*

*Hasta llegar a un tope.*

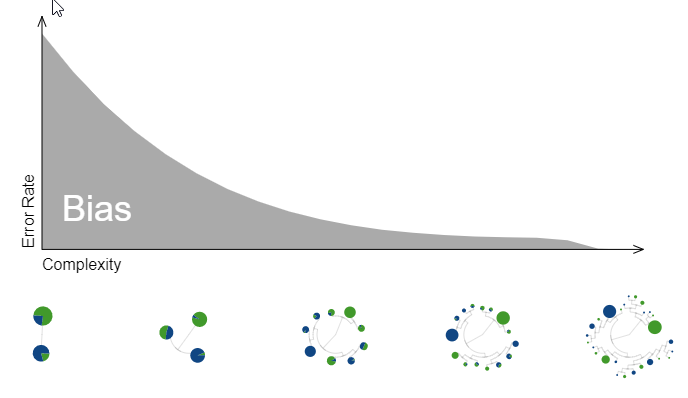
*Un modelo demasiado simplista es tan problemático como uno demasiado escrupuloso. Los errores debidos al sesgo y los debidos a la varianza son distintos. Entender el equilibrio entre el sesgo y la varianza (y cómo los diferentes tipos de modelos permiten equilibrar ambos) es fundamental para modelar bien.*

##### La relación en términos abstractos

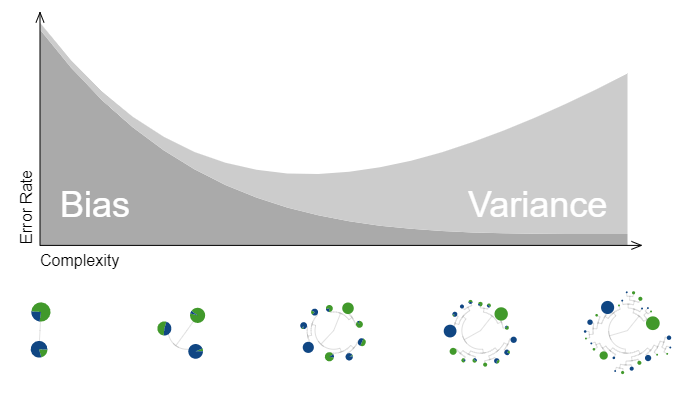
*Una forma habitual de visualizar la relación entre la complejidad y la precisión del modelo es en un gráfico.*



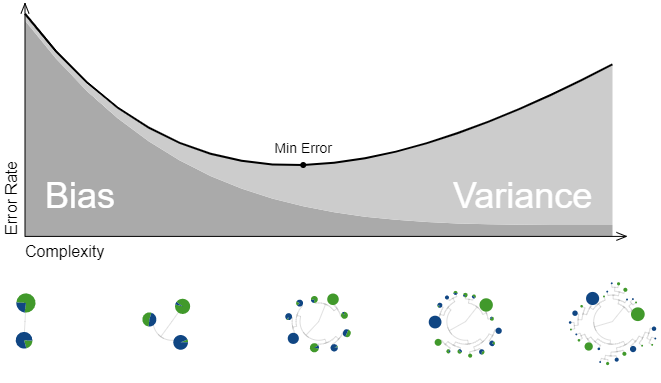
*La relación que se establece entre un parámetro como el tamaño mínimo de los nodos y el error del modelo ilustra de forma más explícita el compromiso entre el sesgo y la varianza.*

*Cuando un modelo es menos complejo, ignora la información relevante, y el error debido al sesgo es alto. A medida que el modelo se hace más complejo, el error debido al sesgo disminuye.* 

*Por otro lado, cuando un modelo es menos complejo, el error debido a la varianza es bajo. El error debido a la varianza aumenta a medida que aumenta la complejidad*



*El error global del modelo es una función del error debido al sesgo más el error debido a la varianza. El modelo ideal minimiza el error de cada uno de ellos.*



*Se puede demostrar matemáticamente que el error debido al sesgo y el error debido a la varianza son distintos.*

##### Conclusión final

*Incluso en su profundidad óptima, los árboles de decisión simples no son los modelos que mejor funcionan. Aunque los árboles son muy fáciles de entender, el mundo es más complejo que un puñado de declaraciones "si-entonces".*

*No obstante, los árboles de decisión pueden utilizarse en conjunto para obtener resultados muy sólidos. Discutiremos estos* ***métodos de ensamble*** *en la Parte III. (no está el link…buscarlo)*

##### Recapitulación

*Los modelos se aproximan a situaciones de la vida real utilizando datos limitados.*

*Al hacerlo, pueden surgir errores debido a suposiciones demasiado simples (sesgo) o demasiado complejas (varianza).*

*La construcción de modelos consiste en asegurarse de que existe un equilibrio entre ambos.*