*Algoritmos y Validación de Modelos de Machine Learning*

La evaluación de modelos es un aspecto fundamental y crítico en todo flujo de Data Science. Pero antes de hablar de métricas de performance, resulta importante entender algunos conceptos básicos y esenciales del Machine Learning.

Empecemos😃

**Conceptos básicos de ML**

***Conjunto de datos (dataset)***

Es la materia prima del sistema de predicción. Es el histórico de datos que se usa para entrenar al sistema que detecta los patrones. El conjunto de datos se compone de instancias, y las instancias de factores, características o propiedades.

****

**Instancia, ejemplo o registro (instance, sample, record)**

Una instancia es cada uno de los datos de los que se disponen para hacer un análisis. Si se quiere predecir el comportamiento de los clientes de un servicio de telefonía, cada instancia correspondería a un abonado. Cada instancia a su vez, está compuesta de características que la describen, como la antigüedad del cliente en la compañía, el gasto diario en llamadas, etc. En una hoja de cálculo, las instancias serían las filas; las características, las columnas.

**Característica, atributo, factor, propiedad o campo (feature, attribute, property, field)**

Son los atributos que describen cada una de las instancias del conjunto de datos. En el caso de una cartera de clientes, estaríamos hablando del número de compras de cada cliente, antigüedad, si es seguidor en redes sociales, si se ha dado de alta en la newsletter, qué productos comprados, etc. En una hoja de cálculo, serían las columnas.

**Objetivo (Objective)**

Es el atributo o factor que queremos predecir, el objetivo de la predicción, como puede ser la probabilidad de reingreso de un paciente tras una intervención quirúrgica.

****

**Ingeniería de Factores (Feature Engineering)**

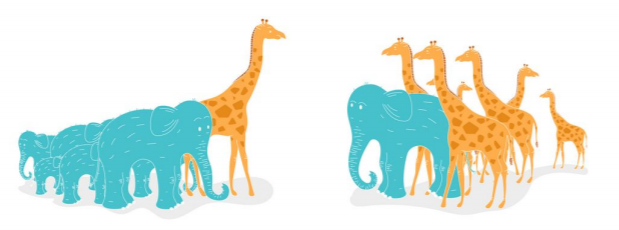
Se trata del proceso previo a la creación del modelo de predicción en el que se hace un análisis, limpieza y estructuración de los campos de los datos. El objetivo, es eliminar los campos que no sirven para hacer la predicción y organizarlos adecuadamente para que el modelo no reciba información que no le es útil y que podría provocar predicciones de poca calidad o confianza. Este proceso es uno de los más importantes y más costosos del proceso de predicción.

**Datos perdidos (Missing data)**

En la gran mayoría de datos con la que trabajemos es muy probable que nos encontremos con valores perdidos, esto es algo muy pero muy normal. El valor faltante puede aparecer de distintas formas, por ejemplo como un signo de interrogación, o N/A, como un 0 o simplemente como una celda en blanco, pero en su mayoría nos lo encontramos representado como NaN que se refiere a “no un número”, pero ¿qué podemos hacer en estos casos? Existen múltiples técnicas para tratar los valores missings, lo veremos más adelante en curso 😃

**Valores Extremos**

¿Qué es un outlier? Es un valor (extremo) que no se corresponde con el patrón general de nuestros datos. Resulta importante destacar, que un valor extremo puede ser bueno, malo o simplemente un error de datos pero en todos esos casos tenemos que realizar un análisis.



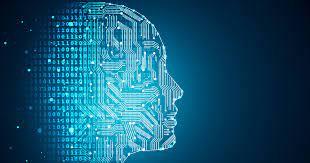
**¿Pero por qué es importante tratar los outlier?** Por que básicamente en términos generales, tratar los outliers suele mejorar los modelos de ML, también muchos modelos avanzados son sensibles a los valores extremos y además, siempre es preferible realizar una buena preparación de datos antes que complejizar los modelos.



**Entrenamiento y Validación**

La pregunta clave que debemos hacernos es la siguiente: **¿Cómo podemos evaluar si nuestro modelo está aprendiendo correctamente o no de nuestros datos?**

Una respuesta posible sería, que para evaluar si nuestro modelo aprendió o no de nuestro datos, observemos su desempeño o performance frente a nuevas instancias es decir, frente a datos que nunca vio.



Pero, **¿por qué es necesario usar nuevas instancias y no podemos usar las instancias con las que se entrenó el modelo?** La respuesta es sencilla, por que no se puede ser *“Juez y parte”* al mismo tiempo.

Entonces de este concepto, es que surgen varias características fundamentales dentro del mundo del Machine Learning. Primero de todo, lo que es el “Entrenamiento” y “Validación” para luego hablar del “Sobreajuste” o “Subajuste”.

Empecemos hablando sobre el Entrenamiento 😃

**Aprendizaje o Entrenamiento (learning, training)**

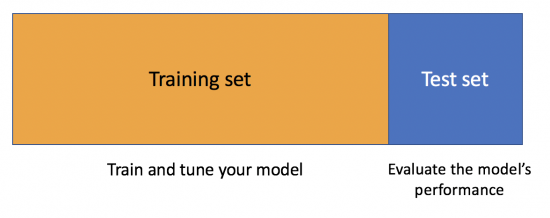
Es el proceso en el que se detectan los patrones de un conjunto de datos, es decir, es el corazón del machine learning. Una vez identificados los patrones, se pueden hacer predicciones con nuevos datos que se incorporen al sistema.

Por ejemplo, los datos históricos de las compras de libros en una web online se pueden usar para analizar el comportamiento de los clientes en sus procesos de compra (títulos visitados, categorías, historial de compras, etc) agruparlos en patrones de comportamiento y hacer recomendaciones de compra, a los clientes nuevos que siguen los patrones ya conocidos o aprendidos.

**Validación**

Es el proceso de evaluar un modelo entrenado sobre un conjunto de datos de prueba. Esto proporciona la capacidad de generalización de un modelo de ML.

Para poder evaluar un modelo correctamente, hay que realizar lo que se conoce como Split de datos es decir, separar nuestro dataset original en *“Datos de Entrenamiento”,* que serán usados justamente para entrenar a nuestro modelo y en *“Datos de Test o de Testing*” que serán aquellos datos que utilizaremos para evaluar la performance de nuestro modelo.

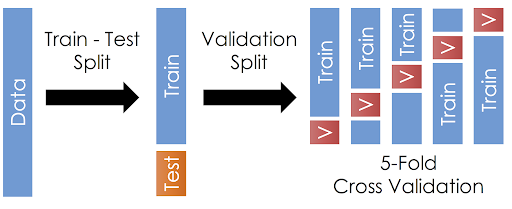


**¿Qué porcentaje se usa para train y test?**

No existe una única respuesta, ni tampoco una verdad absoluta al respecto, pero en términos generales se suele utilizar un 70 % de nuestros datos para el training y un 30 % para el testing.

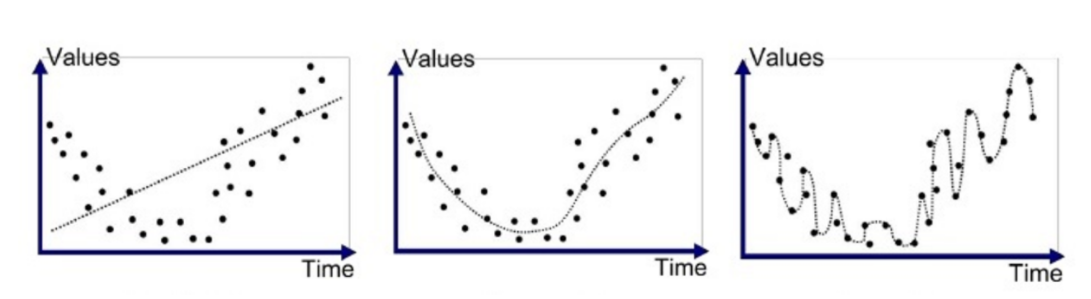
Validación Cruzada

La Validación Cruzada o también conocida como Cross-Validation, es una forma de evaluación muy utilizada en ML, que separa los datos en diferentes particiones y obtiene la media de las diferentes evaluaciones de las distintas particiones. Esta separación que se realiza, ayuda a evaluar los resultados que devuelve el modelo y garantizar la independencia de las particiones que hacemos, con lo cual se evita el sobreajuste.

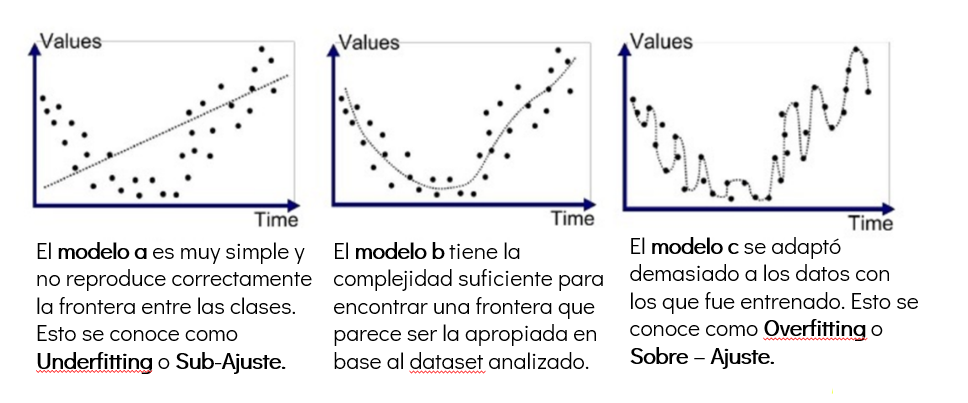


**Overfitting y Underfitting**

Considerando los siguientes 3 escenarios:



**¿Qué modelo parece ser el mejor?**



Entonces:

**¿Qué conceptos tenemos que entender de todo esto que hemos estado desarrollando?**

Las principales causantes de obtener malos resultados en Machine Learning son el Overfitting o el Underfitting de los datos. Dado que cuando entrenamos nuestro modelo intentamos “hacer encajar” -fit en inglés- los datos de entrada entre ellos y con la salida.

Tanto el Over como el Under – Fitting, se relacionan al fallo de nuestro modelo al generalizar -encajar- el conocimiento que pretendíamos que adquieran.

Ahora bien : **¿Qué es el Overfitting?**

Se podría decir que es el problema que surge cuando nuestro modelo se aprende los datos de train perfectamente, por lo que no es capaz de generalizar, y cuando le lleguen nuevos datos obtiene pésimos resultados. Existen diferentes formas de prevenir el *Overfitting* como ser por ejemplo:

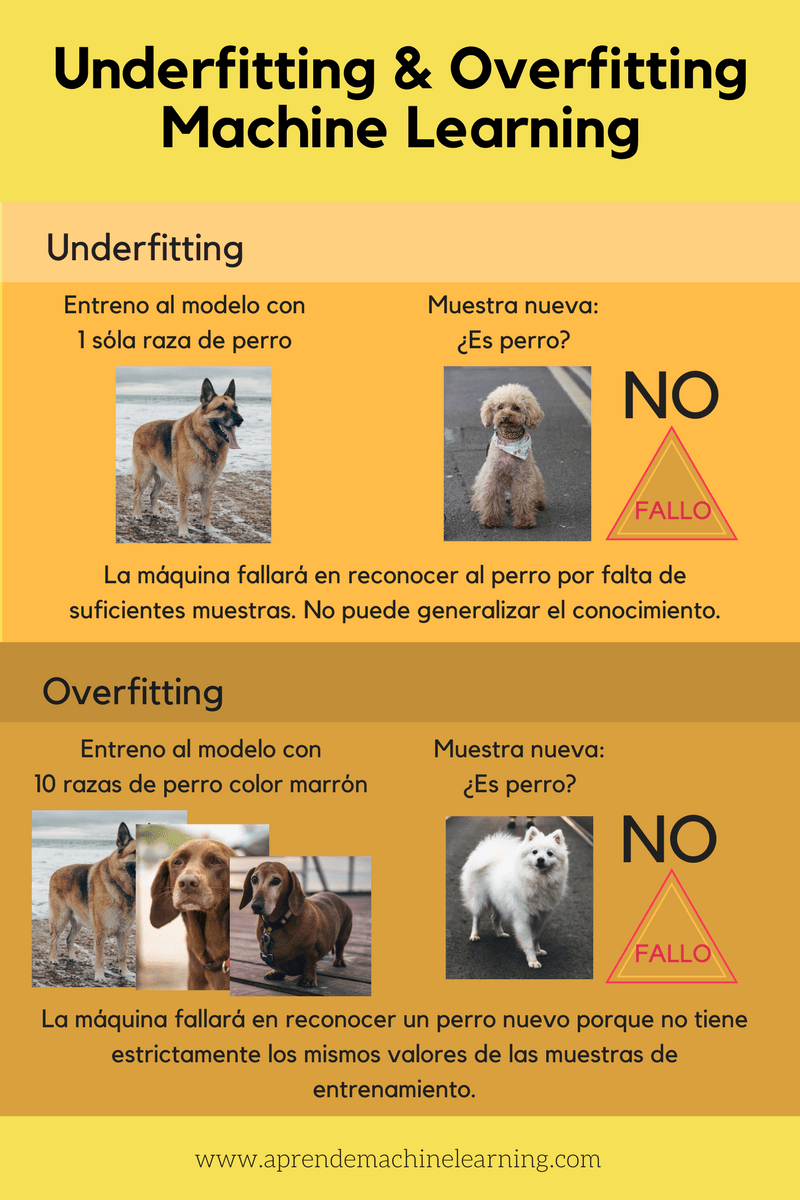
* Dividir nuestros datos en training, validación y testing.
* Obtener un mayor número de datos.
* Ajustar los parámetros de nuestros modelos.
* Utilizar modelos más simples en caso de ser posible.

**¿Y el Unverfitting?**

Se podría decir que es el problema que surge cuando nuestro modelo no es capaz de identificar patrones. Por lo que tendrá siempre pésimos resultados. Existen también, diferentes formas de prevenir el Underfitting como ser:

* Tratar los datos correctamente, eliminando outliers y variables innecesarias.
* Utilizar modelos más complejos.
* Ajustar los parámetros de nuestros modelos.

*Overfitting y Underfitting en una Imágen*



**Métricas y Evaluación de Modelos**

Para ir cerrando algunos conceptos de esta sesión, resulta importante comenzar a hablar acerca de las diferentes métricas que existen dentro del Machine Learning para evaluar la performance de nuestro modelo.

Dentro de este contexto, es relevante destacar que simplemente realizaremos una primera aproximación a la temática, en el *Módulo 6: Validación de resultados* *del Modelo y Tuneo* este tema se verá y tratará de manera detallada.

**Métricas para Algoritmos de Clasificación – Matriz de Confusión**

En el campo de la Inteligencia Artificial y el Aprendizaje Automático, la Matriz de Confusión es una herramienta que permite visualizar el desempeño de un algoritmo de aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real. En términos prácticos entonces, nos permite ver qué tipos de aciertos y errores está teniendo nuestro modelo a la hora de pasar por el proceso de aprendizaje con los datos.

**Matriz de Confusión**

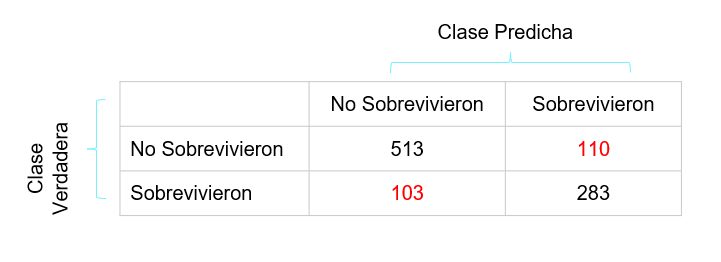


**Interpretación:**

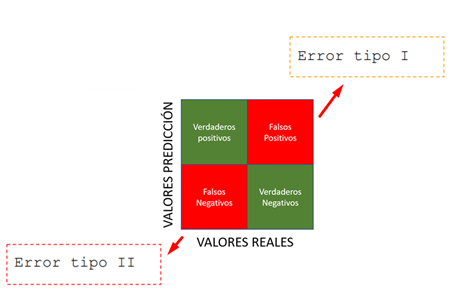
* *Verdadero Positivo (TP)*: Predije que era positivo y efectivamente era positivo.
* *Verdadero Negativo (TN)*: Predije que era falso y efectivamente era falso.
* *Falso Positivo (FP)*: Predije que era positivo pero resultó ser negativo.
* *Falso Negativo (FN)*: Predije que era negativo pero resultó siendo positivo.

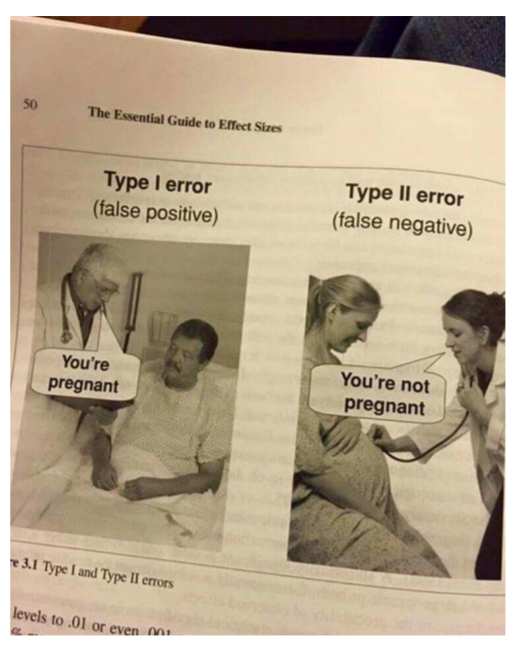
Resulta importante mencionar, que los Verdaderos Positivos como Negativos son aciertos, los Falsos Negativos como Positivos son errores.

**Matriz de Confusión – Ejemplo Titanic:**



**Matriz de Confusión:**





**La Matriz de Confusión y sus Métricas**

**La Exactitud ( en inglés Accuracy )** : Se refiere a lo cerca que está el resultado de una medición del valor verdadero. En términos estadísticos, la exactitud está relacionada con el sesgo de una estimación. Se representa por la proporción entre los positivos reales predichos por el algoritmo y todos los casos positivos.

En forma práctica la Exactitud es el % total de elementos clasificados correctamente.

(VP+VN)/(VP+FP+FN+VN) \* 100

**La Precisión (Precision)**: Se refiere a la dispersión del conjunto de valores obtenidos a partir de mediciones repetidas de una magnitud. Cuanto menor es la dispersión mayor la precisión. Se representa por la proporción entre el número de predicciones correctas (tanto positivas como negativas) y el total de predicciones. En forma práctica, es el porcentaje de casos positivos detectados y nos sirve para medir la calidad del modelo de ML en tareas de clasificación.

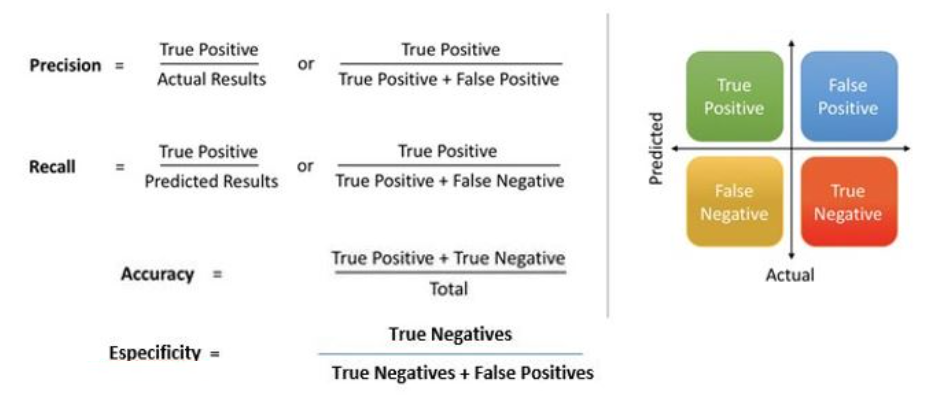
Se calcula como: **VP/(VP+FP)**

**La Sensibilidad (“Recall” o “Sensitivity”):** También se conoce como Tasa de Verdaderos Positivos (True Positive Rate) ó TP. Es la proporción de casos positivos que fueron correctamente identificadas por el algoritmo.

Se calcula: VP/(VP+FN) o lo que seria igual en términos de salud: Verdaderos positivos / Total Enfermos (o bien: es la capacidad de poder detectar correctamente la enfermedad entre los enfermos).

**La Especificidad (“Especificity”):** También conocida como la Tasa de Verdaderos Negativos, (“true negative rate”) o TN. Se trata de los casos negativos que el algoritmo ha clasificado correctamente. Expresa cuan bien puede el modelo detectar esa clase.

Se calcula: VN/(VN+FP), o lo que seria igual en términos de salud: Verdaderos Negativos / Total Sanos (o bien: la capacidad de poder identificar los casos de los pacientes sanos entre todos los sanos)

****

También existe la métrica conocida como F1 – Score, esta es otra métrica muy empleada porque nos resume la Precisión (Precisión) y Sensibilidad (Recall) en una sola métrica.

Se calcula:

2 \* (Recall \* Precision) / (Recall + Precision)

Algunas apreciaciones que se desprenden de la métrica de **F1 – Score** son las siguientes:

* Alta precisión y alto recall: el modelo maneja perfectamente esa clase.
* Alta precisión y bajo recall: el modelo no detecta la clase muy bien, pero cuando lo hace es altamente confiable.
* Baja precisión y alto recall: El modelo detecta bien la clase, pero también incluye muestras de la otra clase.
* Baja precisión y bajo recall: El modelo no logra clasificar la clase correctamente.

**Link de Interés – Métricas de Clasificación:**

* <https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/>
* [https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/#Precision\_Precision](https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/)
* <https://sitiobigdata.com/2019/01/19/machine-learning-metrica-clasificacion-parte-3/>
* https://fayrix.com/machine-learning-metrics\_es

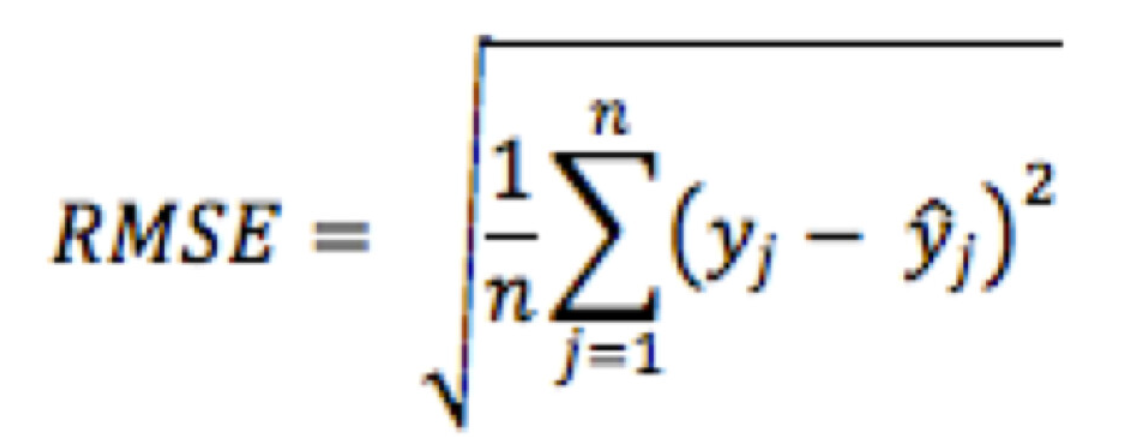
Por último hablamos sobre algunas métricas para evaluación de Modelos de Regresión, recordemos que con un modelo de regresión, predecimos o estimamos el valor numérico de una cantidad desconocida, de acuerdo con unas características dadas. La diferencia entre la predicción y el valor real es el Error, este es una variable aleatoria, que puede depender de las características dadas.

En la actualidad hay muchas formas para estimar el rendimiento y evaluar el ajuste del modelo, algunas de ellas son:

* El Error Cuadrático Medio (RMSE, por sus siglas en inglés, Root Mean Squared Error).
* Error Absoluto Medio (MAE, Mean Absolute Error).
* R-Cuadrado.

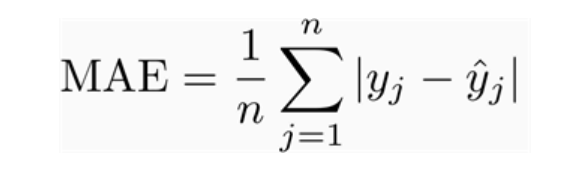
Existen varias métricas más como ser por ejemplo, el R cuadrado ajustado (R²), MSPE – Error de porcentaje cuadrático medio, entre otras.

**Error cuadrático medio (RMSE):** La métrica más comúnmente utilizada para las tareas de regresión, es el error cuadrático medio y representa a la raíz cuadrada de la distancia cuadrada promedio entre el valor real y el valor pronosticado.

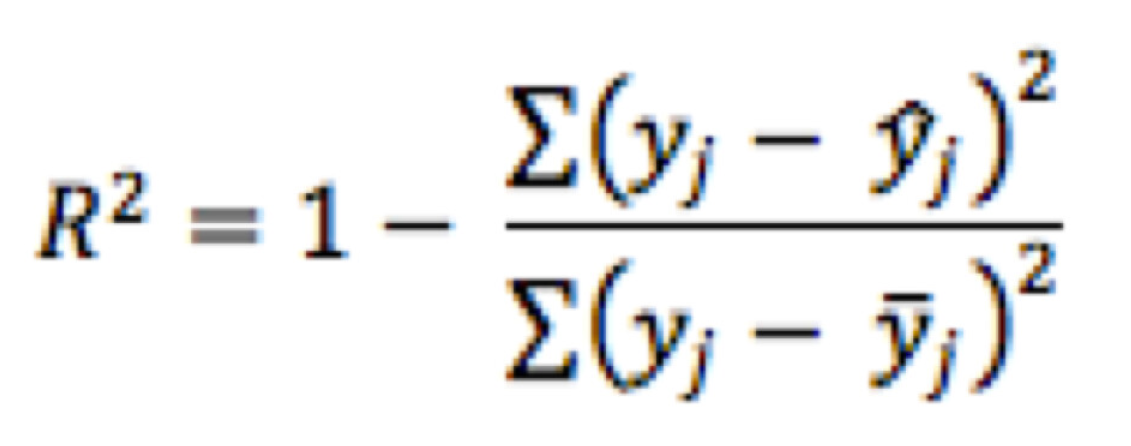


Indica el ajuste absoluto del modelo a los datos, cuán cerca están los puntos de datos observados de los valores predichos del modelo.

**Error absoluto medio (MAE):** Es la diferencia absoluta entre el valor objetivo y el valor predicho por el modelo. El MAE es más robusto para los valores atípicos y no penaliza los errores tan extremadamente como el MSE. Este tipo de métrica, no es adecuada para aplicaciones en las que desea prestar más atención a los valores atípicos.



**R2:** El R-cuadrado indica la bondad o la aptitud del modelo, a menudo se utiliza con fines descriptivos y muestra que también las variables independientes seleccionadas explican la variabilidad en sus variables dependiente. R2 se define como:



R-cuadrado tiene la propiedad útil de que su escala es intuitiva, va de 0 a 1, con 0 indicando que el modelo propuesto no mejora la predicción sobre el modelo medido y 1 indica una predicción perfecta.