

UNIVERSIDAD  
COMPLUTENSE  
DE MADRID



Autor: María José  
Gómez Silva

# Machine Learning con Python. Semana 1.

## CAPÍTULO 1. INTRODUCCIÓN AL MACHINE LEARNING.

### 1.4 ¿Qué es aprender para una máquina? Problemas del entrenamiento

#### ¿QUÉ ES APRENDER PARA UNA MÁQUINA? PROCESO DE ENTRENAMIENTO

Para una máquina o programa, aprender es ajustar los parámetros de un modelo de modo que le permita solucionar una determinada tarea (como las de regresión, clasificación, clustering, etc.), con cierto rendimiento.

El ajuste de los parámetros del modelo se realiza mediante un proceso cíclico de prueba y error, llamado *entrenamiento*, que requiere de la observación de múltiples ejemplos.

Gracias al entrenamiento, las máquinas adquieren la capacidad para generalizar una solución. A diferencia de la inteligencia humana, la artificial requiere de muchas muestras (ejemplos) de una determinada tarea antes de aprender a realizarla con precisión. Por ejemplo, si un niño es llevado al zoo y ve por primera vez un elefante, instantáneamente conceptualiza la idea de “qué es un elefante”. La próxima vez que vea un elefante, aunque esté en otra posición, a otra distancia, sea de otro tamaño, o esté parcialmente oculto por árboles, en cualquier caso, el niño será capaz de generalizar y predecir que efectivamente esos nuevos “ejemplos” también son elefantes. Sin embargo, una máquina necesita ser entrenada con miles o incluso millones de ejemplos hasta adquirir tal capacidad de generalización.

En el entrenamiento se establece un determinado objetivo. En cada iteración (ciclo) el modelo ofrece una solución o predicción para los datos de entrada dados. Se calcula cuánto se desvía la solución dada por el modelo respecto al objetivo a cumplir y se reajustan los parámetros del modelo para intentar minimizar esa desviación.

En el caso concreto del aprendizaje supervisado, tenemos datos de entrenamiento formados no sólo por las variables de entrada, sino también por sus respectivas etiquetas (solución conocida). Existe una relación entre las entradas y sus etiquetas, pero es desconocida, y la labor del entrenamiento es encontrar tal relación, o lo que es lo mismo definir el modelo que calcula la salida (etiqueta) en función de la entrada.

Supongamos que tenemos que entrenar un modelo que permita predecir el número de préstamos de libros diarios que se realizan en una biblioteca, en función de la edad media de la población en la localidad en la que se encuentra la biblioteca. Tenemos los siguientes datos correspondientes a distintas fechas, que han sido recogidos previamente y se emplearán como datos de entrenamiento,  $x=[34, 39, 43, 47, 52, 55, 60, 64, 69, 72, 78]$  e  $y'=[40, 42, 50, 65, 68, 88, 87, 104, 112, 127, 128]$ , donde  $x$  son los datos de entrada, la edad media de la población, e  $y'$  son sus respectivas etiquetas, las visitas diarias contabilizadas, que será la variable a predecir. Los datos están representados en la Figura 1.

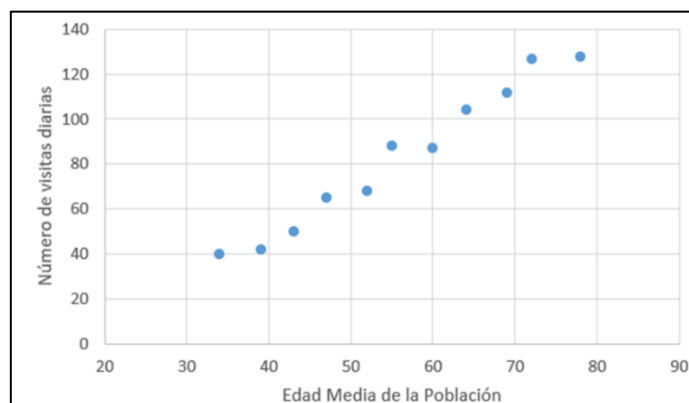


Figura 1. Representación de los datos de entrenamiento

Los datos parecen presentar una relación (tendencia) lineal entre los valores de entrada y sus respectivas etiquetas. Por lo tanto, en este caso, se empleará un modelo lineal de regresión para realizar predicciones. Es decir, partimos de la idea de que la función que relaciona una determinada entrada,  $x_i$ , y su correspondiente salida,  $y_i$ , sigue la ecuación de una recta:  $y_i = a + b * x_i$ . Esa ecuación de la recta será el modelo a entrenar, donde  $a$  y  $b$  son los parámetros del modelo a ajustar.

En el proceso de entrenamiento se parte de una hipótesis o modelo inicial, es decir, de unos valores iniciales para los parámetros,  $a$  y  $b$ , del modelo. Suponiendo que  $a = -20$  y  $b = 1$  son los valores iniciales, nuestra hipótesis sería una línea recta como la representada en la Figura 2, que se aleja bastante de modelar correctamente la tarea.

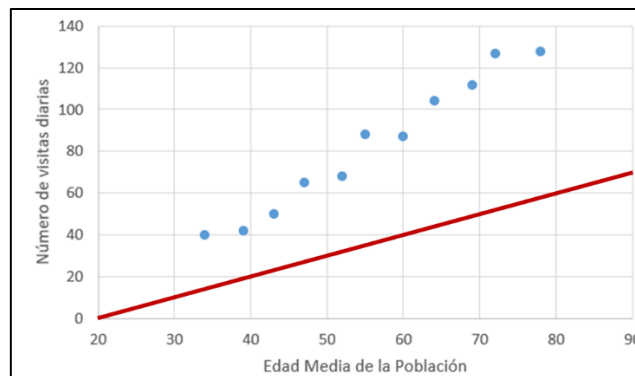


Figura 2. Hipótesis inicial del modelo  $y_i = a + b * x_i$ , con  $a = -20$  y  $b = 1$ .

Si empleamos la hipótesis anterior para realizar predicciones,  $y$ , en función de los datos  $x$  de entrada, o lo que es lo mismo, si calculamos el valor de  $y$  ofrecido por la ecuación  $y_i = -20 + 1 * x_i$ , para cada valor  $x$  de los datos de entrenamiento, obtenemos los siguientes valores  $y = [14, 19, 23, 27, 32, 35, 40, 44, 49, 52, 58]$ , que distan mucho de las etiquetas reales  $y'$  de referencia. Concretamente la diferencia entre las etiquetas de referencia y las predicciones del modelo son  $y' - y = [26, 23, 27, 38, 36, 53, 47, 60, 63, 75, 70]$ . La figura 3 muestra tales diferencias (en verde) entre la altura real de los puntos (etiqueta  $y'$  de referencia) y la altura que tendrían según la predicción,  $y$ , del modelo.

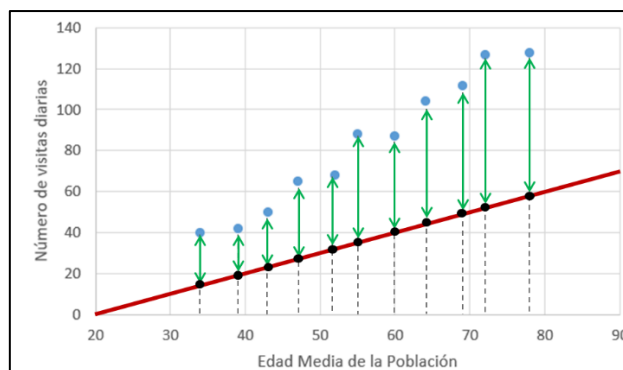


Figura 3. Desviación de la predicción del modelo,  $y$ , respecto a los valores de referencia,  $y'$ , de los datos de entrenamiento.

Gracias a la observación del error cometido en cada iteración, los algoritmos de Machine Learning son capaces de estimar automáticamente si deben aumentar o disminuir el valor actual de los parámetros del modelo. El incremento o decremento del valor de los *parámetros* es lo que llamamos ajuste de los parámetros del modelo, o simplemente *ajuste del modelo*.

Por ejemplo, siguiendo el caso anterior y suponiendo que los nuevos valores de los parámetros son  $a = -18$  y  $b = 1.4$ , el modelo tendría el aspecto de la recta representada en la Figura 4.

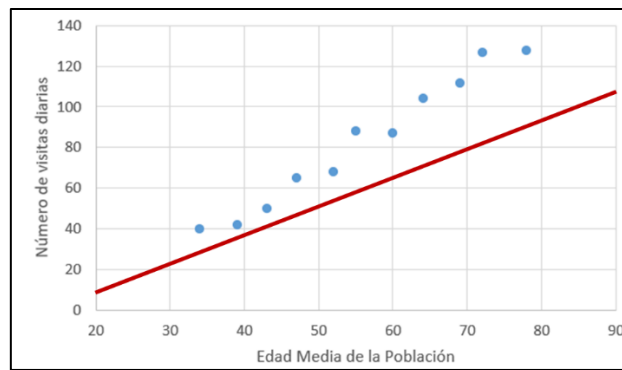


Figura 4. Modelo  $y_i = a + b * x_i$ , con  $a = -18$  y  $b = 1.4$ .

Tras el ajuste de los parámetros, la ecuación del modelo cambia y el ciclo vuelve a repetirse. En cada iteración, se realizan predicciones con el modelo, se calcula la desviación de las predicciones con respecto a las etiquetas de referencia y se vuelven a ajustar los parámetros del modelo. Este proceso cíclico se repetirá automáticamente hasta que el modelo se ajuste correctamente a los datos, es decir, hasta alcanzar un valor pequeño del error cometido por las predicciones.

Una posible solución al problema planteado en el ejemplo sería la mostrada en la Figura 5, donde podemos observar que el modelo aprendido ( $a = -46.7$ ,  $b = 2.33$ ) sigue la ecuación de una recta (rojo) que se ajusta bastante bien a los datos.

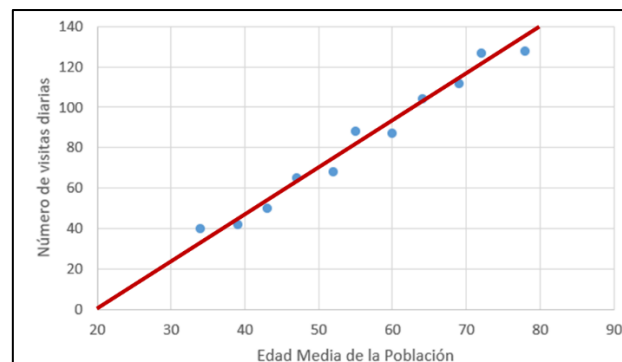


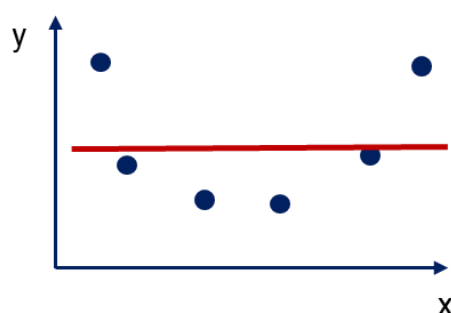
Figura 5. Modelo final, ajustado a los datos de entrenamiento

El entrenamiento no tiene por qué anular por completo el error existente entre las predicciones ofrecidas por el modelo para los datos de entrada y las etiquetas de referencia de los mismos. El objetivo es tratar de minimizar la suma de los errores, para obtener una estimación de la tendencia que siguen los datos. De este modo cuando llegue un nuevo dato de entrada, por ejemplo, la edad media de la población según el último censo, el modelo podrá ofrecer una predicción de los préstamos de libros que se espera que sean solicitados en la biblioteca. Puede que finalmente el valor real no sea exactamente el predicho por el modelo, pero el modelo nos permite tener una estimación.

## PROBLEMAS DEL ENTRENAMIENTO

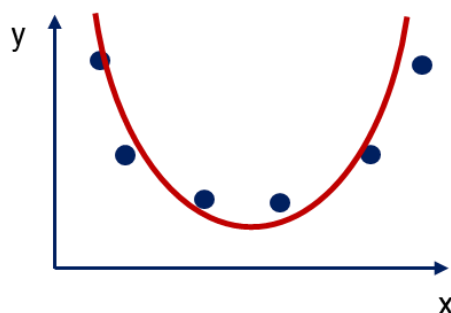
Las ecuaciones que definen un modelo pueden ser muy dispares y con distintos grados de complejidad. En la sección anterior se empleó como ejemplo un problema de regresión lineal, donde se parte de una hipótesis que sigue la ecuación de una línea recta. Sin embargo, dependiendo de la complejidad de los datos, podría ocurrir que la línea recta no fuese adecuada para modelar dichos datos.

En la Figura 6, se puede observar como el modelo lineal no se ajusta bien a los datos disponibles. En este caso, el problema no reside en los valores dados a los parámetros  $a$  y  $b$  del modelo ( $y_i = a + bx_i$ ), sino al hecho de que el modelo es demasiado sencillo para la tarea encomendada.



*Figura 6. Modelo lineal que no se ajusta bien a los datos de entrenamiento.*

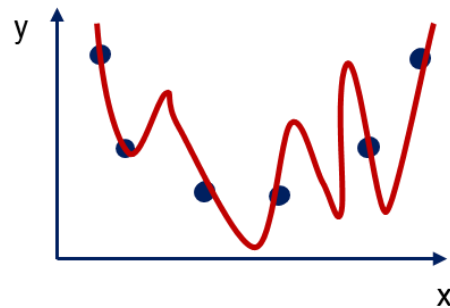
Una posible solución es aumentar la complejidad del modelo, lo cual implica un aumento del número de parámetros. Por ejemplo, en este caso, podríamos pasar de la ecuación de una línea recta a la ecuación de una parábola ( $y = a + bx + cx^2$ ), como muestra la Figura 7. Esto supone pasar de tener que ajustar dos parámetros ( $a$  y  $b$ ) a tener que ajustar tres parámetros ( $a$ ,  $b$  y  $c$ ).



*Figura 7. Modelo parabólico que se ajusta bien a los datos de entrenamiento.*

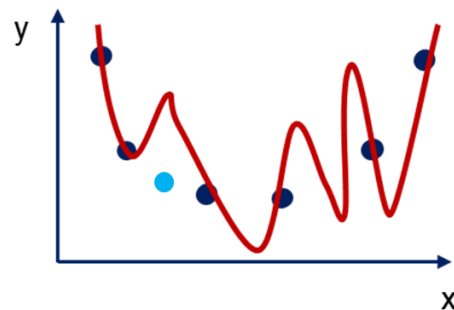
En conclusión, la complejidad del modelo a entrenar debe ser proporcional a la complejidad de la tarea a desempeñar o a la complejidad de los datos disponibles. De igual modo que un modelo demasiado sencillo no es adecuado para modelar datos con

tendencias complejas, un modelo demasiado complejo tampoco dará buenos resultados sobre datos sencillos. Esto se debe a que cuantos más parámetros tiene un modelo, más flexible es su forma y por ello presenta más facilidad para adaptarse a los datos, como se observa en la Figura 8.



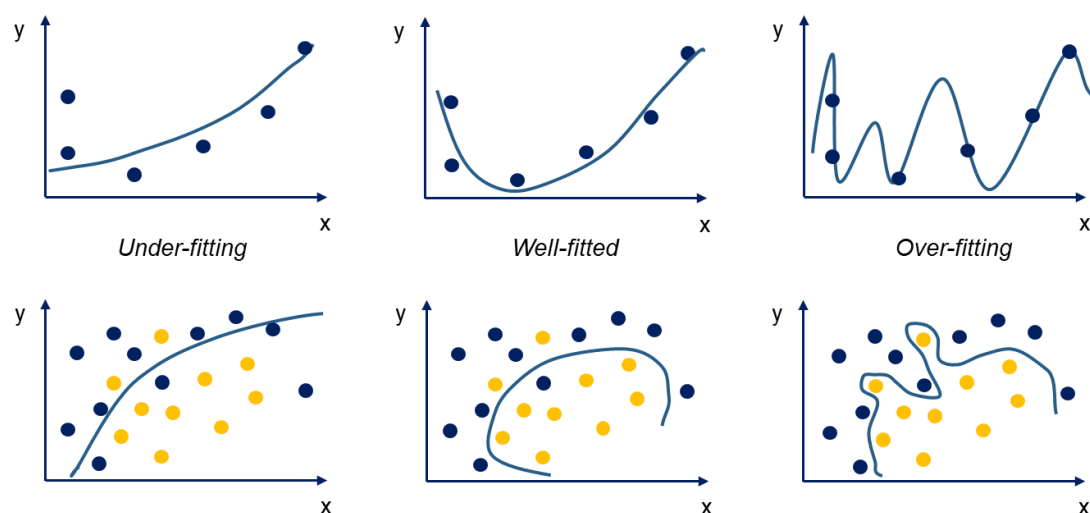
*Figura 8. Modelo complejo perfectamente ajustado a los datos de entrenamiento.*

Sin embargo, no siempre es conveniente que el modelo final aprendido se ajuste con total exactitud a los datos de entrenamiento. El objetivo del entrenamiento es ajustarse a la tendencia de los datos lo máximo que sea posible sin que la capacidad de generalización se vea comprometida. La capacidad de generalización del modelo se refiere a la precisión con la que ofrece una predicción a partir de nuevos datos de entrada, diferentes de los empleados en el entrenamiento. A estos nuevos datos los llamamos datos de prueba o de test. La figura 9 muestra como el modelo anterior no se ajusta bien a un nuevo dato desconocido (azul claro).



*Figura 9. Modelo complejo perfectamente ajustado a los datos de entrenamiento.*

Un buen modelo es aquel que ofrece predicciones precisas para los datos de test. Si esto no sucede, puede deberse a dos causas, que el modelo está *desajustado (under-fitting)* o que el modelo ha sido *sobre-ajustado (over-fitting)*. En la Figura 10 puede observarse el caso de un modelo desajustado, otro ajustado apropiadamente y otro sobre-ajustado, para las tareas de regresión y de clasificación.



*Figura 10. Modelos de regresión y clasificación desajustados, ajustados apropiadamente y sobre-ajustados.*

Cuando el modelo está desajustado (under-fitting), sus predicciones no son precisas ni con los datos de entrenamiento, ni con los de test. El desajuste puede deberse a que el modelo es demasiado sencillo, o a que no se ha entrenado el tiempo suficiente para ajustar bien los parámetros. Puede corregirse, aumentando el tiempo de entrenamiento o la complejidad del modelo, dependiendo del caso.

Cuando el modelo está sobre-ajustado, sus predicciones son muy precisas con los datos de entrenamiento pero no con los de test. El sobre-ajuste puede deberse a que el modelo es demasiado complejo o a que ha sido entrenado durante demasiado tiempo. Puede corregirse, finalizando el entrenamiento de forma temprana, reduciendo la complejidad del modelo o aumentando el número de datos de entrenamiento.

Un modelo bien entrenado alcanza el equilibrio adecuado entre las dos situaciones anteriores.