# Tema 3

# Aprendizaje Automático



Gerard Alba Soler

Ejercicio 1:

1. En el problema a resolver, las variables objetivo o dependientes son discretas:
   * **Corresponde a un algoritmo de Clasificación.** En estos algoritmos de clasificación, lo que podemos identificar es que las variables objetivo son discretas y representan categorías o clases. Un buen ejemplo de este caso son clasificar los emails como spam o no.
2. El resultado del modelo son valores continuos:
   * **Corresponde a un algoritmo de Regresión.** En los algoritmos de regresión, tienen como objetivo predecir valores numéricos continuos, muy común en redes neuronales. El modelo es una variable continua, como números reales. Durante este bachelor utilizamos una regresión lineal en una lista de clientes en las cuales debíamos predecir si tomarían o no un crédito en base a distintas variables.
3. El objetivo es predecir categorías o etiquetas de clases:
   * **Corresponde a un algoritmo de Clasificación.** Tal como lo explicamos en el primer caso.
4. Ejemplos de estos modelos son: predicción del precio de una acción en bolsa; predecir el precio de un inmueble; predecir la cantidad de demanda de un producto:
   * **Corresponde a un algoritmo de Regresión.** Este tipo de problemas son típicamente resueltos con modelos de regresión, ya que implican predecir valores continuos, como precios o cantidades.
5. A este grupo pertenecen las regresiones lineales y no-lineales:
   * **Corresponde a un algoritmo de Regresión.** Tanto las regresiones lineales como las no-lineales son tipos de algoritmos de regresión que se utilizan para modelar relaciones entre variables y predecir valores continuos

Ejercicio 2:

En este código, se utiliza un modelo de aprendizaje automático k-NN (k-Nearest Neighbors) para un conjunto de datos de cáncer de mama, en este ejercicio el propósito es evaluar el rendimiento del modelo para diferentes valores de **n\_neighbors**.

Graficamos los resultados en el cual se representa la precisión del modelo tanto en datos de entrenamiento y en datos de prueba.

**Análisis grafico**

Los resultados del gráfico muestran la precisión del modelo de k-vecinos más cercanos (kNN) en los conjuntos de entrenamiento y prueba para diferentes valores de k (número de vecinos). Aquí están mis observaciones:

* En el gráfico, en el eje x, tenemos el número de vecinos (**n\_neighbors**) que tiene un rango entre 1 a 10. En el eje y, se muestra la precisión del modelo k-NN en los datos de entrenamiento y en los datos de prueba para cada valor de **n\_neighbors**.
* La línea naranja representa la precisión del modelo en los valores de “training” mostramos cómo se comportaría el modelo en los datos según la variación de la variable **n\_neighbors**, mientras que la línea azul representa la precisión en los valores “test” muestra cómo se desempeña el modelo en los datos de prueba a medida que aumenta **n\_neighbors**.
* Podemos analizar que la precisión del modelo en el conjunto de “test” en una primera instancia va en aumento, alcanza un punto el cual podríamos inferir que es óptimo ya que luego los valores resultantes del modelo empiezan a disminuir a medida que aumenta k.
* En general, debemos saber que los modelos no necesariamente funcionan mejor teniendo una mayor complejidad (disminuyendo **n\_neighbors)**. Un modelo demasiado complicado podría caer en la falla de sobre ajustar los datos de “training” y como resultado de eso, tener un rendimiento deficiente en los datos de “test”. Esto se conoce como sobreajuste. En contra parte, un modelo demasiado básico podría ser deficiente en la captura estructural de los datos y caer en un subajuste lo cual también afecta el rendimiento de los datos “test”.

La precisión es más alta cuando **n\_neighbors** es aproximadamente 3 o 4. Esto sugiere que el modelo tiene un buen equilibrio entre sesgo y varianza en ese punto.

