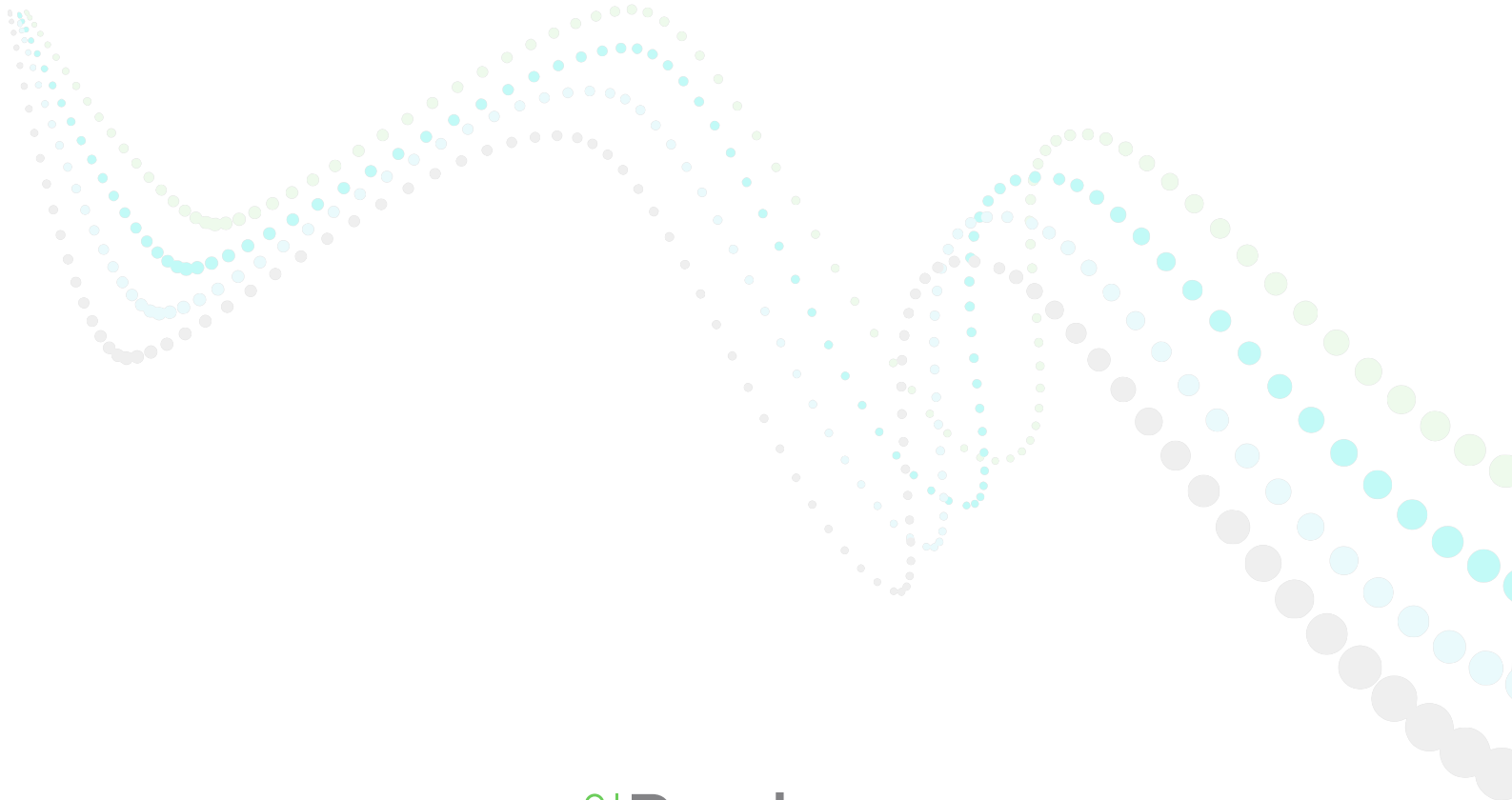
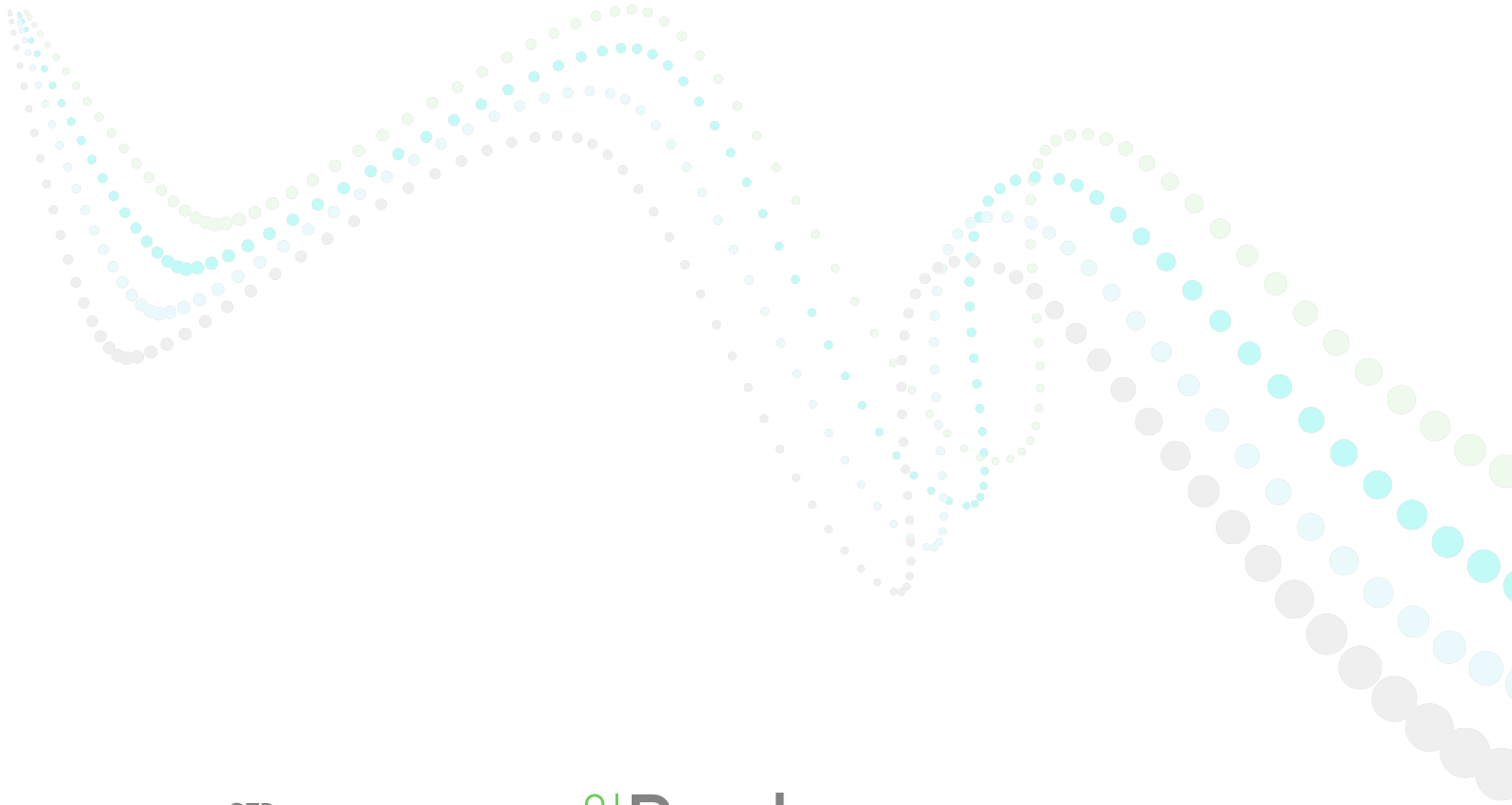


# Aprendizaje Supervisado y No Supervisado:



Notas:

Aprendizaje supervisado	3
Aprendizaje no-supervisado	4
Simbología básica	4
Función de ajuste estándar en ML	5
Evaluación de algoritmo	6



## Aprendizaje supervisado

La mayoría de los problemas en aprendizaje de máquina utilizan aprendizaje supervisado. En estos problemas tienen variables de entrada (X) y una variable de salida (Y); se utilizan algoritmos para “aprender” una función que mapee los valores de la entrada a la salida:

$$Y = f(X)$$

El objetivo es aproximar la función que mejor mapee los datos ante la presencia de nuevos datos de entrada (X) prediciendo la variable de salida (Y) para esos datos.

Los problemas de aprendizaje supervisado se agrupan en:

**-Clasificación:** La variable (Y) es categórica (cualitativa). En el caso de que se tengan dos clases, se representan por los dígitos binarios de 0 y 1, o -1 y 1. Cuando hay más de dos categorías, lo más común es utilizar *variables dummy*

**-Regresión:** La variable (Y) es un valor en los reales. (cuantitativa)

## Aprendizaje no-supervisado

En este caso sólo cuentas con datos de entrada ( $X$ ) y **ninguna variable de salida ( $Y$ )**. El objetivo de este tipo de aprendizaje es encontrar una estructura en los datos que permita “aprender” para nuevos datos.

Los problemas de aprendizaje no-supervisado se agrupan en:

**-Clustering:** Quieres descubrir grupos en los datos

**-Asociación:** Quieres descubrir reglas o patrones en grandes volúmenes de datos.

### Simbología básica

Las variables de entrada ( $X$ ) normalmente son vectores en donde sus componentes se especifican con los subíndices  $X_j$ . Los valores observados se escriben en minúsculas, por ejemplo  $x_i$  será el  $i$ -ésimo valor de  $X$ . Las matrices son representados en mayúsculas y negritas. Por ejemplo, un conjunto con  $N$  entradas en  $p$ -vectores  $x_i$ ,  $i = 1 \dots N$  puede ser representado como una matriz  **$X$**  de  $N \times p$

## Función de ajuste estándar en ML

La representación más común es el modelo  $Y = f(X) + e$ . En el caso de aprendizaje supervisado el objetivo es que  $f$  “aprenda” a través de los datos. Para ello, observando las variables de entrada y salida, se selecciona un conjunto de observaciones de *entrenamiento*  $T = (x_i, y_i)$ ,  $i = 1, \dots, N$  el cuál es un subconjunto de las observaciones totales.

A través de un algoritmo de aprendizaje (usualmente un programa computacional) se ajustan los datos a través de una función  $\hat{f}(x_i)$

Notar que ésta función es la aproximación del verdadero  $f$ . El algoritmo de aprendizaje tiene la propiedad de que puede modificar la relación entre las entradas y salidas en  $\hat{f}$  en respuesta a las diferencias entre  $y_i - \hat{f}(x_i)$ . Este proceso es conocido como “learning by example”.

Muchas de las aproximaciones están asociadas a parámetros  $\theta$  que pueden ser modificados para ajustar mejor los datos. Por ejemplo, en el modelo lineal,  $f(x) = x^T \beta$  se considera que  $\theta = \beta$

Para el caso de regresión lineal, el proceso para estimar el parámetro  $\theta$  se lleva a cabo a través de la minimización de RSS:

$$\text{RSS}(\theta) = \sum_{i=1}^N (y_i - f_{\theta}(x_i))^2$$

Sin embargo para los procesos de ML en general, se utiliza el principio de máxima verosimilitud que es una generalización de RSS.

## Evaluación de algoritmo

Se mencionó anteriormente que a través de un algoritmo de aprendizaje (programa computacional) se ajustan los datos a través de una función  $\hat{f}(x_i)$

En ese sentido, existen criterios para determinar el desempeño de los algoritmos. En el caso de regresión, se utiliza validación cruzada y desempeño del error cuadrático para el conjunto de prueba. En el caso de clasificación se utilizan los indicadores de matriz de confusión.

La matriz de confusión nos dará una mejor idea de cómo está clasificando nuestro modelo, dándonos un conteo de los aciertos y errores de cada una de las clases por las que estemos clasificando. Así podremos comprobar si nuestro modelo está confundiéndose entre clases, y en qué medida.

		Clasificador	
		+	-
Valor real	+	TP	FN
	-	FP	TN

Cada columna de la matriz representará el número de predicciones para cada clase realizadas por el modelo, y cada fila los valores reales por cada clase. Con lo cual los conteos quedan divididos en 4 clases, TP, FN, FP y TN, que significan lo siguiente:

**TP – True Positives:** Son el número verdaderos positivos, es decir, de predicciones correctas para la clase +.

**FN – False Negatives:** Son el número de falsos negativos, es decir, la predicción es negativa cuando realmente el valor tendría que ser positivo. A estos casos también se les denomina errores de tipo II.

**FP – False Positives:** Son el número de falsos positivos, es decir, la predicción es positiva cuando realmente el valor tendría que ser negativo. A estos casos también se les denomina errores de tipo I.

**TN – True Negatives:** Son el número de verdaderos negativos, es decir, de predicciones correctas para la clase -.

Gracias a estas cuatro categorías podemos calcular métricas más elaboradas, como, por ejemplo:

- **Sensibilidad:** también se la llama *recall* o *tasa de verdaderos positivos*. Nos da la probabilidad de que, dada una observación realmente positiva, el modelo la clasifique así.

$$\text{Sensibilidad} = \frac{TP}{TP + FN}$$

- **Especificidad:** también llamado *ratio de verdaderos negativos*. Nos da la probabilidad de que, dada una observación realmente negativa, el modelo la clasifique así.

$$\text{Especificidad} = \frac{TN}{TN + FP}$$

- **Precisión:** también llamado *valor de predicción positiva*. Nos da la probabilidad de que, dada una predicción positiva, la realidad sea positiva también.

$$\text{Precisión} = \frac{TP}{TP + FP}$$

- **Valor de predicción Negativa:** Nos da la probabilidad de que, dada una predicción negativa, la realidad sea también negativa.

$$\text{Valor de predicción negativa} = \frac{TN}{TN + FN}$$

- **Error de clasificación:** Porcentaje de errores del modelo.

$$\text{Error de clasificación} = \frac{FP + FN}{TP + TN + FP + FN}$$

- **Accuracy:** Porcentaje total de los aciertos de nuestro modelo.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$