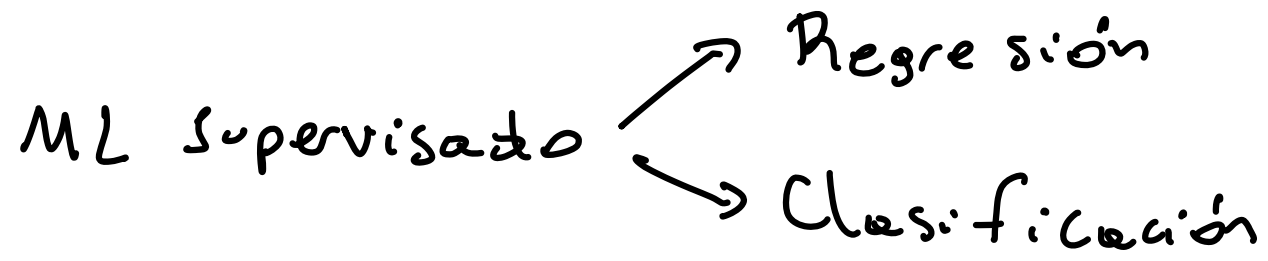


# Diplomado de ciencia de datos - UNAM

## Módulo 6 - Aprendizaje de máquina supervisado



### Regresión

Entrada:  $X \in \mathbb{R}^d \rightarrow$  features, variables, dimensiones

Salida:  $y \in \mathbb{R} \rightarrow$  target, variable dependiente

Objetivo: Encontrar una función  $f: \mathbb{R}^d \rightarrow \mathbb{R}$  tal que  $y \approx f(x, w)$  para el par  $(x, y)$  donde  $w$  son los parámetros del modelo

Regresión lineal: Método de regresión en el que  $f$  es una función lineal de los parámetros  $w$  y de las variables  $x$ .

Línea recta:

$$\begin{aligned} y &= w_0 x + b \\ &= m x + b \\ &= w_1 x_1 + w_0 \\ &= \beta_1 x_1 + \beta_0 \end{aligned}$$

Modelo de regresión lineal:

Si escribimos la ecuación de una recta en  $d$  dimensiones para  $i$  variables tenemos

$$y_i = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_n x_n$$

$$y_i = w_0 + \sum_{j=1}^d w_j x_{ij}$$

$$y_i = w_0 + \bar{w}_1 \bar{x}_1^T + \bar{w}_2 \bar{x}_2^T + \dots$$

$$y_i \approx f(x_i, w) = w_0 + \sum_{j=1}^d w_j x_{ij}$$

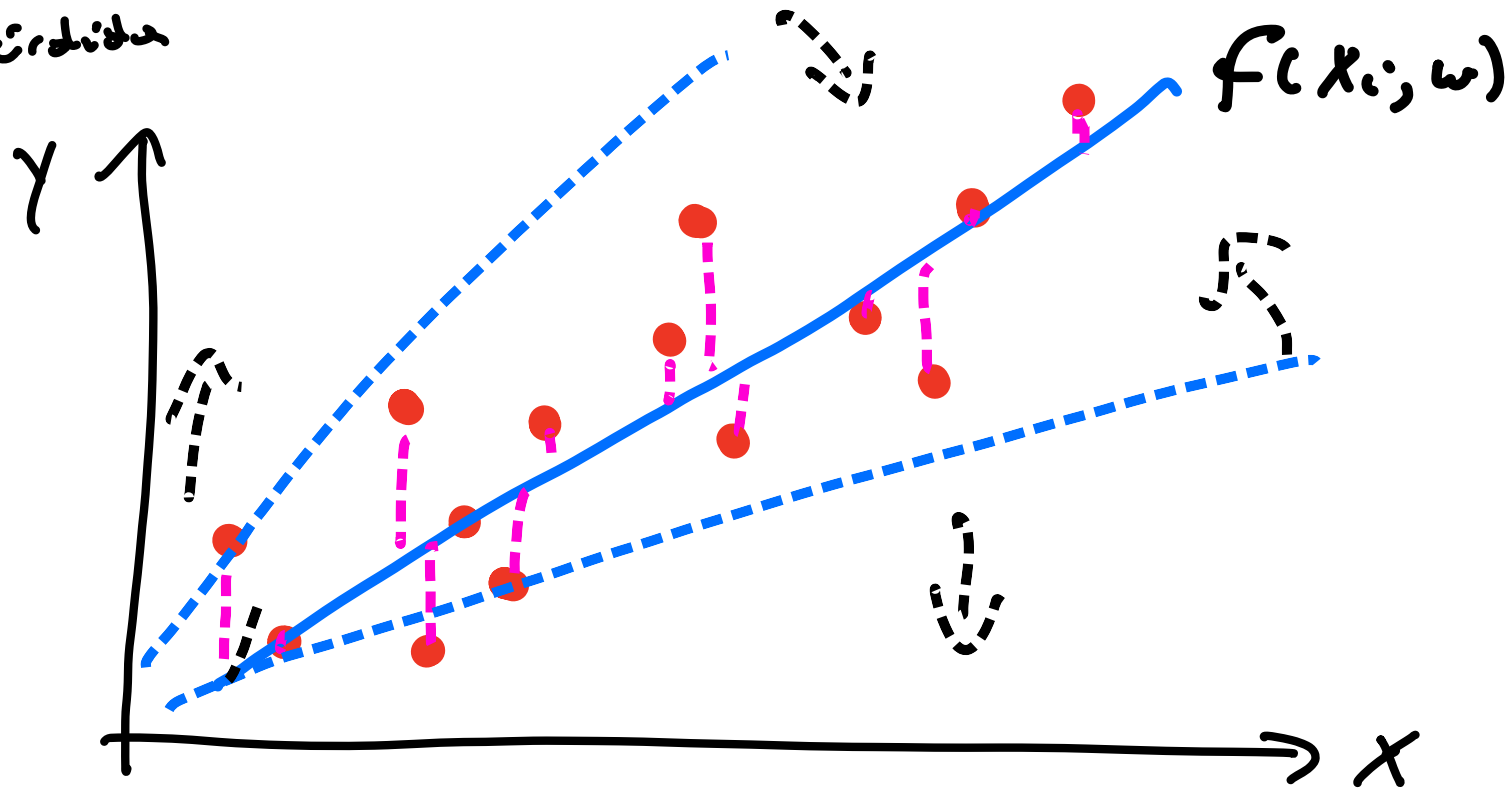
Model learning: Tenemos datos de entrenamiento  $(x_1, y_1), (x_2, y_2) \dots (x_d, y_d)$ . Queremos usar estos datos para aprender unos  $w$  tal que  $y_i \approx f(x_i, w)$ . Para eso necesitamos una función objetivo que nos diga cuán "bueno" es el valor de  $\textcircled{w}$ .

Método de los mínimos cuadrados (Least squares)

El método de LS nos permite encontrar una  $w$  que minimice la suma de los errores al cuadrado.

$$\mathcal{L} = \sum_{i=1}^d (y_i - f(x_i; w))^2$$

Función de pérdida



$$w_{LS} = \underset{w}{\operatorname{argmin}} \sum_{i=1} (y_i - f(x_i, w))^2$$

$$w_{LS} = \underset{w}{\operatorname{argmin}} \mathcal{L}$$

Queremos encontrar un  $w_{LS}$  tal que se minimize la distancia promedio entre  $y_i$  y  $f(x_i, w)$

$$\begin{aligned} f(x_i, w) &= \underline{\bar{x}_i^T w} \\ &= w_0 + \bar{w}_1 \bar{x}_1^T + \bar{w}_2 \bar{x}_2^T \dots \end{aligned}$$

Entonces,

$$\mathcal{L} = \sum_{i=1}^d (y_i - x_i^T w)^2$$

$$= \|y - Xw\|^2 \quad \downarrow \text{Denominator}$$

$$= (y - Xw)^T (y - Xw)$$

Ahora calculamos el gradiente con respecto a  $w$ .

$$\nabla_w \mathcal{L} = 2X^T Xw - 2X^T y$$

Para encontrar el mínimo igual a 0

$$2X^T Xw - 2X^T y = 0$$

Resolvemos para  $w$

$$w_{LS} = \frac{X^T y}{X^T X} = \frac{X^T y}{X^T X}$$

Recordando la inversa

$$w_{LS} = (X^T X)^{-1} X^T y$$

↳ Lo que vamos a programar en el siguiente notebook.

---

Workflow de la ciencia de datos

↳ enfocado en ML

1: Entendimiento de negocio y contexto de problema.

2: Recolección de datos

2.1 Activa

2.2 Pasiva

3: Exploratory data analysis (EDA)

↗  
Entendimiento de datos

4: Limpieza de datos (puede ser más complejo)

4.1 Missing data

4.2 Trabajo con outliers

4.3 Eliminar columnas constantes

4.4 Eliminar columnas con muchos errores

4.5 Reconocimiento de entidades

4.6 Limpieza de texto (eliminar acentos,



eliminar caracteres especiales, poner todo en minúsculas)

5: Preparación de datos (no todo es requerido siempre)

5.1 Castear si es necesario

5.2 Normalizar

5.3 Estandarizar

5.4 Binning

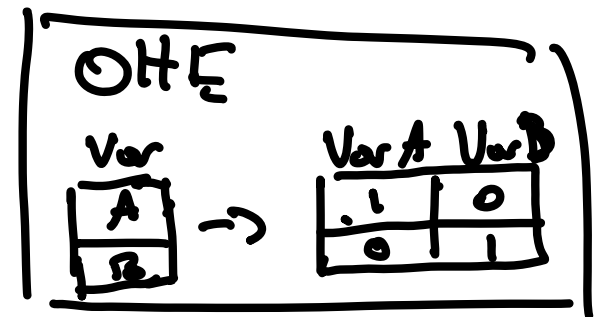
5.5 Creación de interacciones y características polinomiales

5.6 One-hot-encoding y label encoding

5.7 Generación de nuevas variables

↳ múltiples técnicas

5.8 Balanceo y remuestreo



6: Selección de mejores variables

## 7: Modelado

7.1 Baseline

7.2 Modelos simples

7.3 Modelos avanzados de prueba

7.4 Cross-validation y búsqueda de hiperparámetros

7.5 Evalúo mis modelos y las mejores variables

7.6 Encontrar explicaciones para mi modelo

7.7 Explicar el modelo de forma técnica a mis pares o jefes técnicos

7.8 Explico el modelo al negocio

↳ sin usar palabras rimbombantes

↳ sin mostrar código

↳ sin mostrar ecuaciones complejas

## 8: Poner modelo en producción → Data Engineers

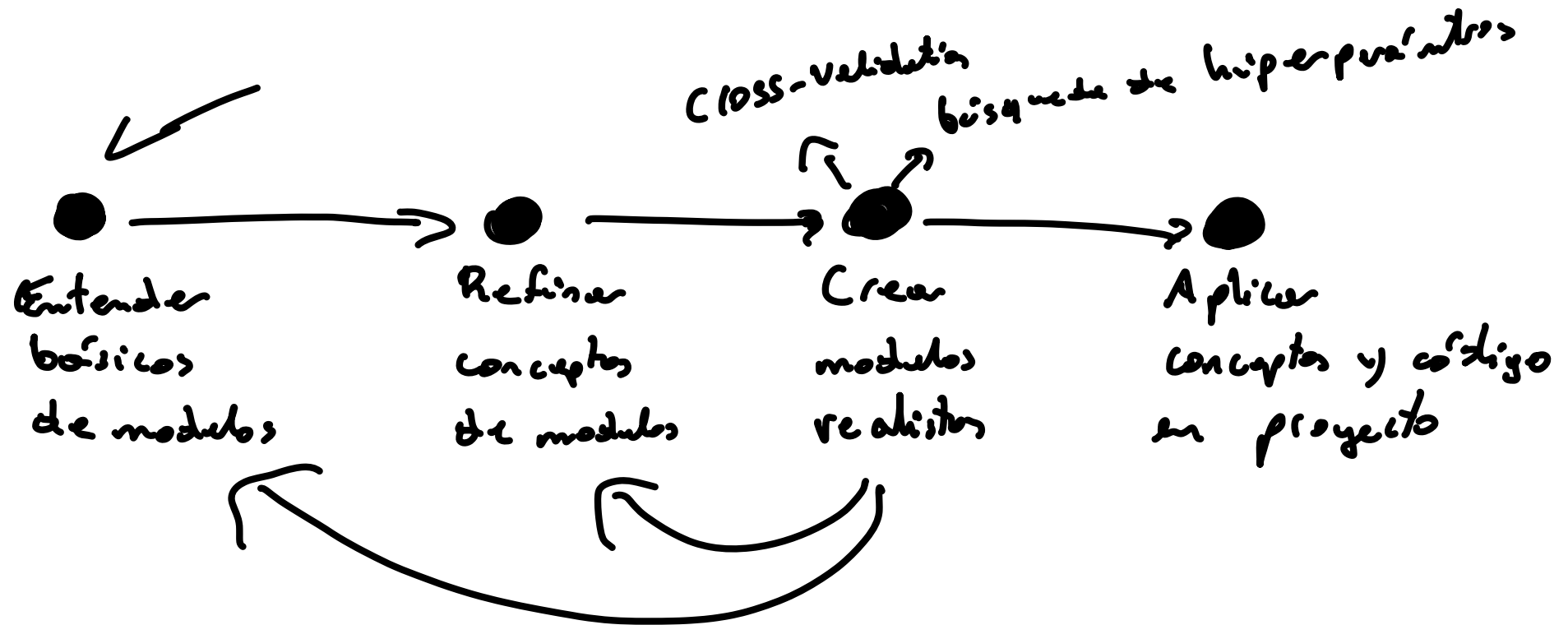
8.1 Empaquetar mi modelo campeón

8.2 Crear backups de mis modelos

8.3 Monitoreo de modelos

## 8.4 Validación de shifting, drift de las variables

### Método "Faro" para aprender ML



---

Parámetros e hiperparámetros

Parámetro: Se obtiene su valor en el proceso de entrenamiento de un modelo de ML. Existen las funciones objetivo y su valor se calcula optimizando (máx. o mín) la función objetivo

$$\hat{y} = w_0 + \bar{w}_1 \bar{x}_1^T$$

\       /  
parámetros

Hiperparámetro: Valores que modifican el comportamiento de un algoritmo de ML, pero que no se aprenden en el proceso de entrenamiento. Se configuran al comienzo del entrenamiento y se mantienen igual durante el mismo. Para encontrar el mejor valor para los hiperparámetros se utilizan técnicas estadísticas, de prueba y error, y últimamente también algoritmos genéticos.

e.g.:

Ridge Regression:

$$\hat{y} = \bar{X}^T \bar{\omega} + \alpha \|\bar{\omega}\|_2^2$$

↖ Parameter      ↖  
↳ hiper parametro