Introducción a Modelos Supervisados

Camilo Esteban Núñez Fernández

INF396 - Introducción a la Ciencia de Datos
Departamento de Informática

2025-04-11

Definición de *Aprendizaje*

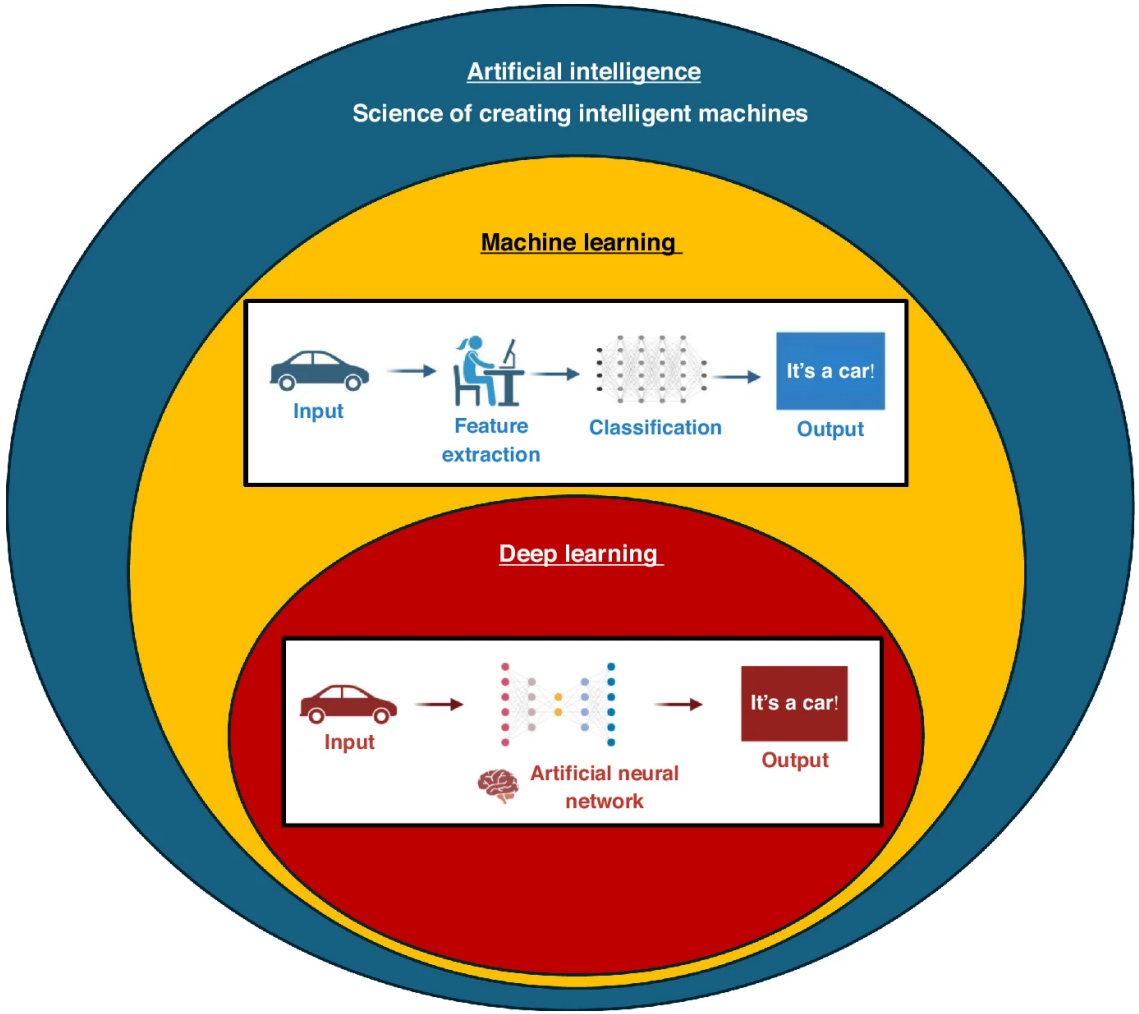
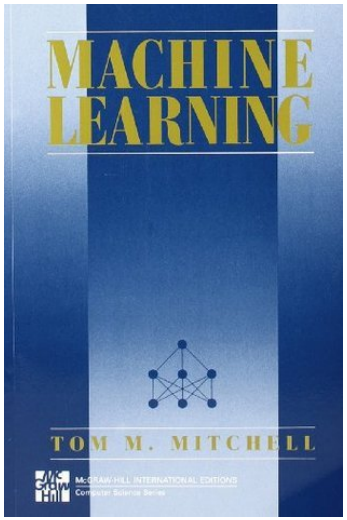


Fig.: Diagrama de Venn para relaciones entre AI, ML y DL.[1]

[1] O'Connor, O., McVeigh, T.P. Increasing use of artificial intelligence in genomic medicine for cancer care- the promise and potential pitfalls. BJC Rep 3, 20 (2025). https://doi.org/10.1038/s44276-025-00135-4

Tom Michael Mitchell



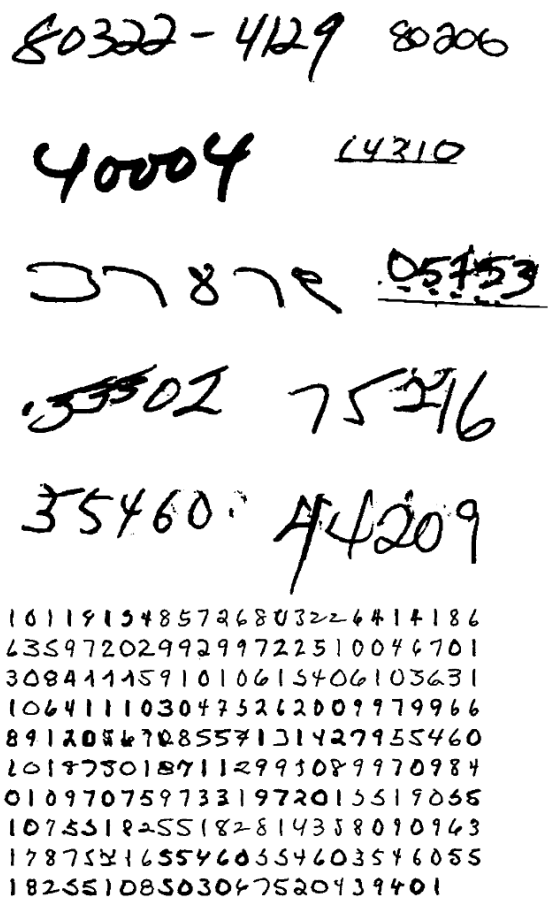
Machine Learning, Mitchell, T.M., 1997, McGraw-Hill Education.

Definición

A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Clasificación de Dígitos Escritos a Mano

- *Task T*: Reconocer y clasificar dígitos escritos a mano desde una imagen.
- *Performance Measure P*: Porcentaje de dígitos clasificados correctamente.
- *Training Experience E*: Secuencia de imágenes etiquetas con dígitos.



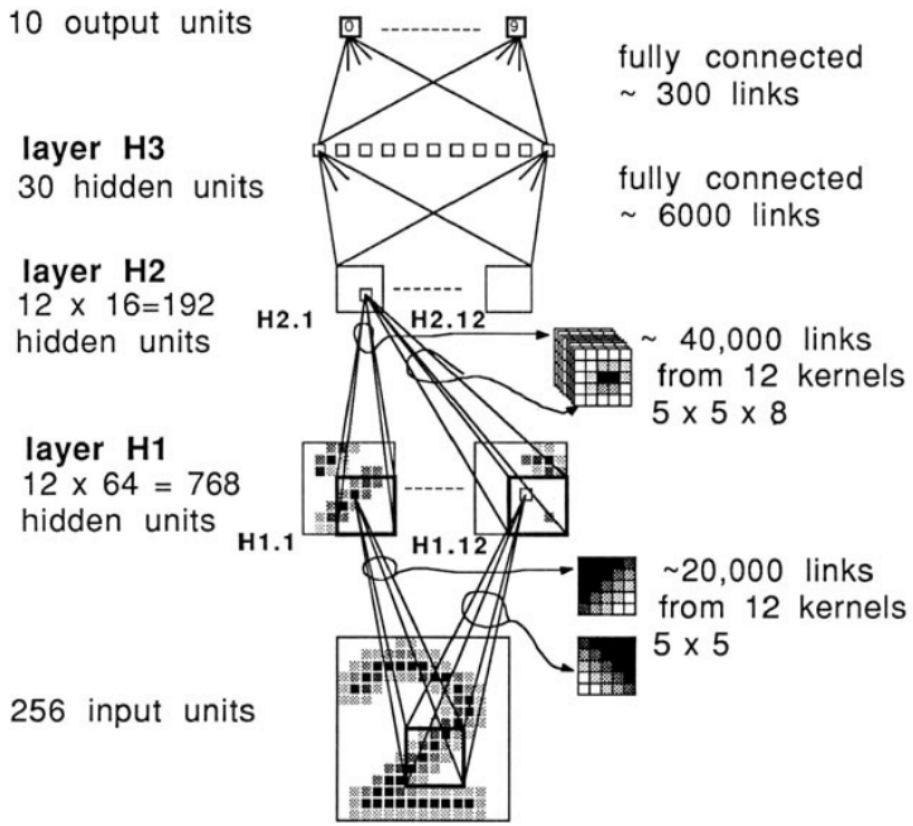


Fig.: Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel; Backpropagation Applied to Handwritten Zip Code Recognition. Neural Comput 1989; 1 (4): 541–551. doi: `https://doi.org/10.1162/neco.1989.1.4.541`

Definición *Task T*

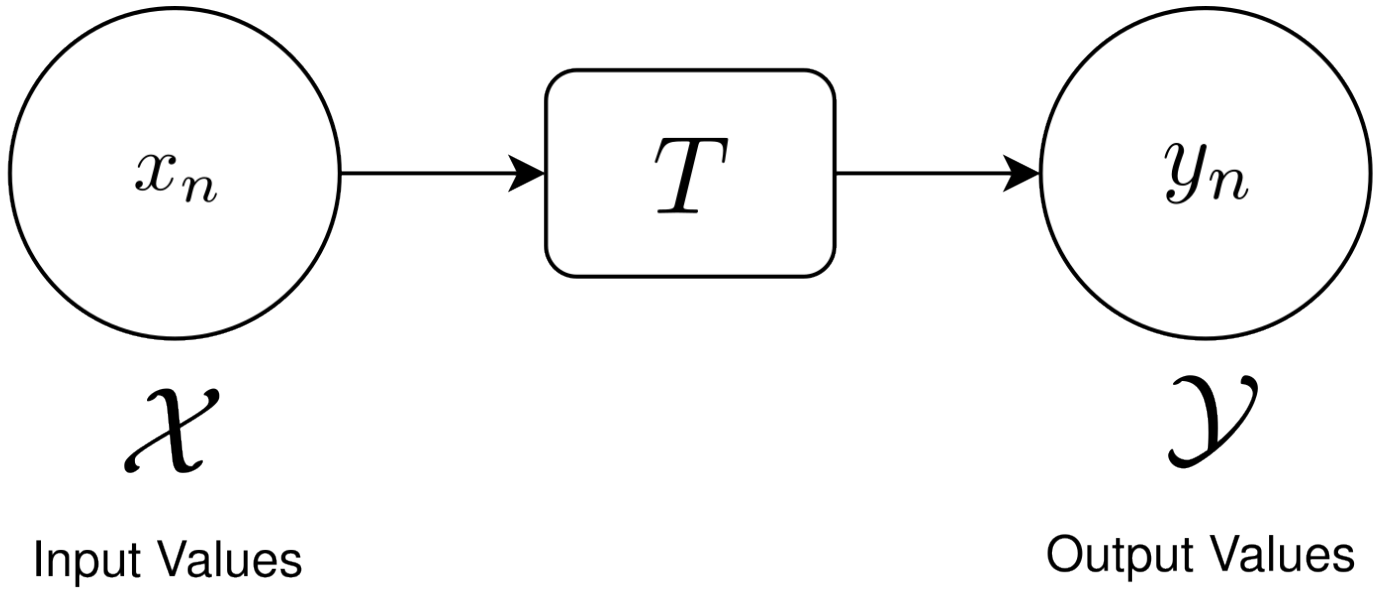
Definición *Task T*

Es el **problema** que busca resolver nuestro programa.

- Ejemplos:
  - Clasificar imagen de un dibujo.
  - Detectar la figura de los autos en una imagen.
  - Clasificar una anomalía en una serie de tiempo.
  - Generar la descripción de una imagen.

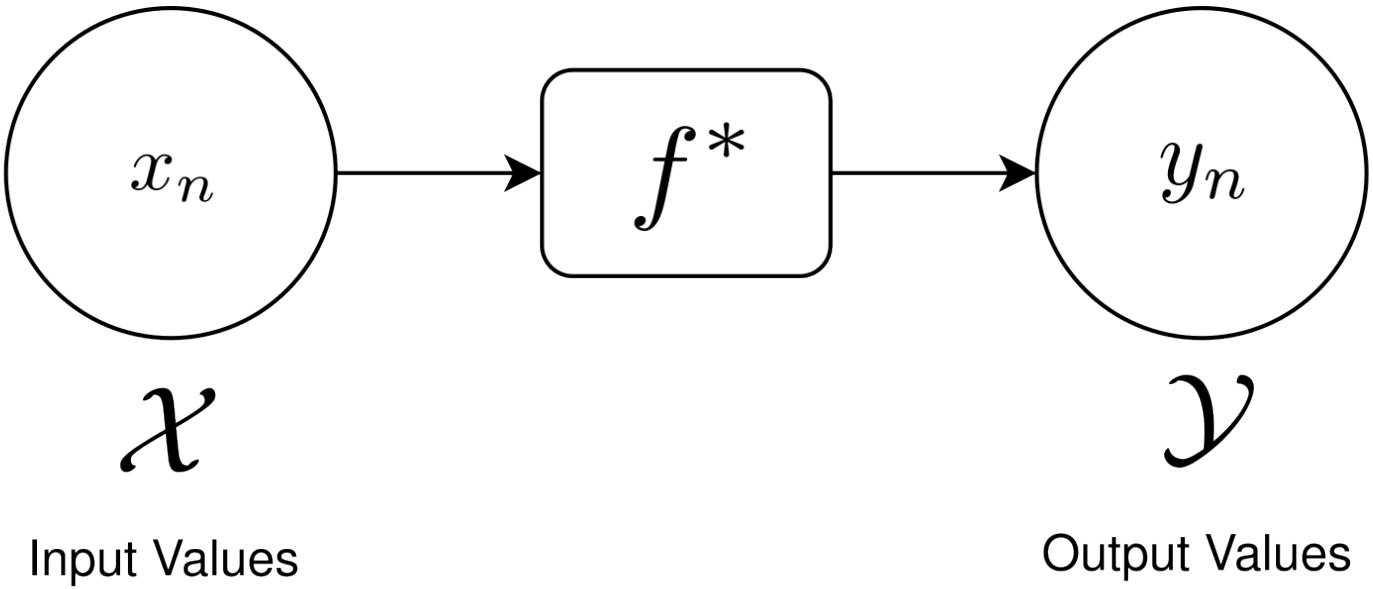
Definición

La *Task T* se puede formalizar como una función o transformación f^* tal que: $f^* : \mathcal{X} \rightarrow \mathcal{Y}$.



Definición

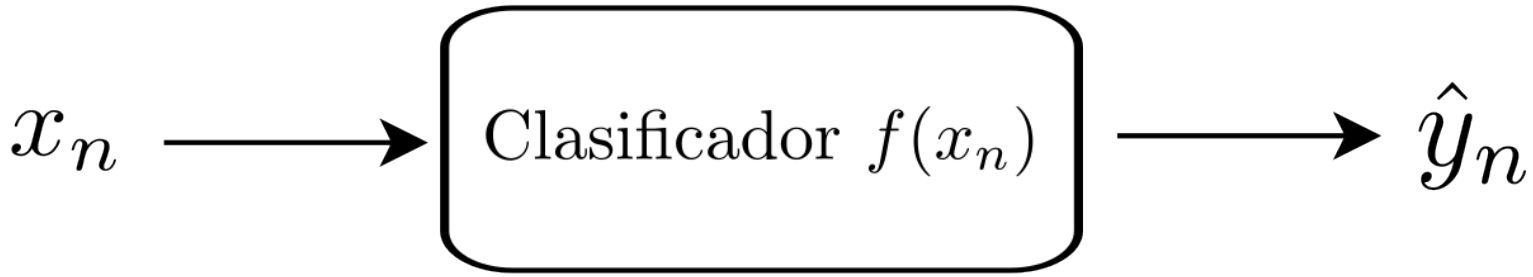
La f^* es **desconocida**, sólo sabes qué elementos toma en un **dominio** $\mathcal{X}$ y qué elementos toma en el **codominio** $\mathcal{Y}$.



Buscamos aproximar f^* según nuestros datos !

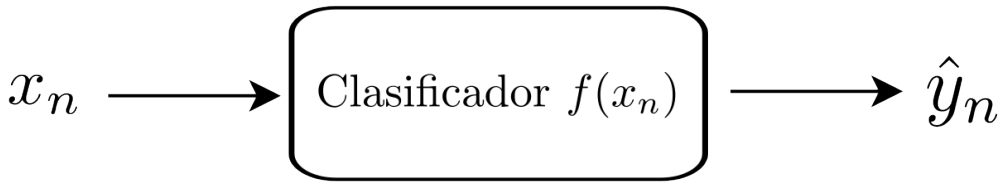
T: Clasificación

- Problema que busca predecir aquellos valores cualitativos o categóricos del espacio $\mathcal{Y}$.



$$f : \mathcal{X} \rightarrow \mathcal{Y} = \{c_0, c_1, \cdots, c_k\}$$
$$x_n \mapsto \hat{y}_n = f(x_n)$$

T: Clasificación



$$f : \mathcal{X} \rightarrow \mathcal{Y} = \{c_0, c_1, \cdots, c_k\}$$
$$x_n \mapsto \hat{y}_n = f(x_n)$$

- Por definición canónica (o histórica), de la ecuación $\hat{y}_n = f(x_n)$, desprenderemos el termino $f(x_n)$, y lo llamaremos la **hipotiposis**.[2]
- En este caso, x_n, **SOLO** puede ser asignado a **UN** elemento (o categoría) c. Las categorías c son mutuamente excluyentes, no se pueden dar simultáneamente.

[2] Es normal encontrar textos donde se sobrescribe $f^* : \mathcal{X} \rightarrow \mathcal{Y}$, como $h : \mathcal{X} \rightarrow \mathcal{Y}$, y donde h es la **hipótesis** y a su vez de función que busca predecir el valor de y_n.

T: Clasificación Multi-Label

- Problema que busca predecir aquellos valores **múltiples** cualitativos o categóricos del espacio $\mathcal{Y}$.

$$f : \mathcal{X} \rightarrow \mathcal{Z} = 2^{\mathcal{Y}}$$

$$x_n \mapsto \{c_0, c_1, \cdots, c_k\} = f(x_n)$$

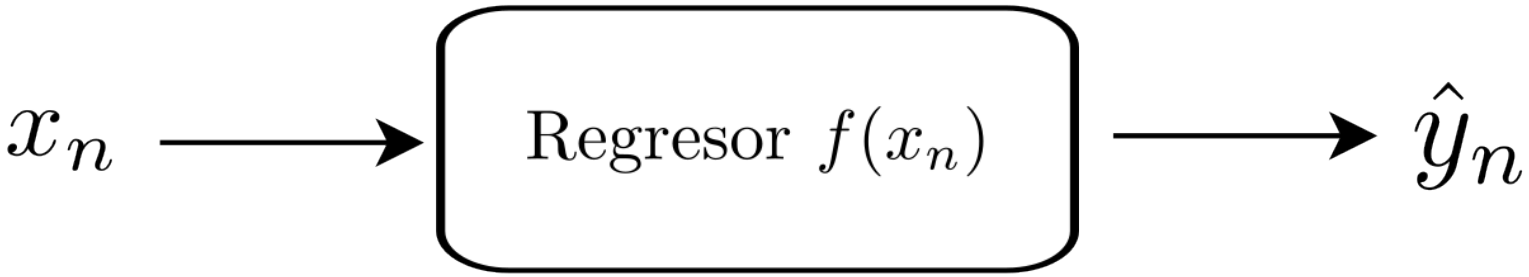
T: Clasificación Multi-Label

$$f : \mathcal{X} \rightarrow \mathcal{Z} = 2^{\mathcal{Y}}$$
$$x_n \mapsto \{c_0, c_1, \cdots, c_k\} = f(x_n)$$

- En este caso, x_n, puede ser asignado a **UNA o MÁS** categorías c. Las categorías c no son mutuamente excluyentes, y sí pueden dar simultáneamente.

T: **Regresión**

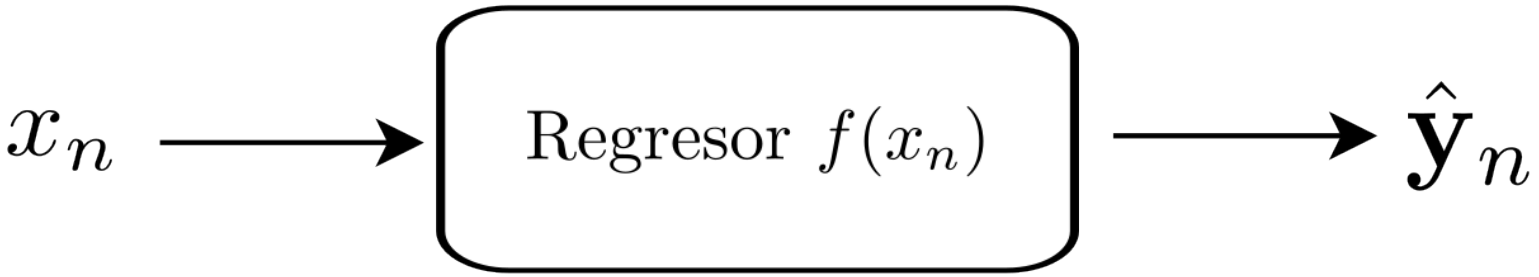
- Problema que busca predecir aquellos valores cuantitativos o continuos en $\Re$ del espacio $\mathcal{Y} \subset \Re$.



$$f : \mathcal{X} \rightarrow \mathcal{Y} \subset \Re$$
$$x_n \mapsto \hat{y}_n = f(x_n)$$

T: Regresión Múltiple

- Problema que busca predecir aquel vector K-dimensional de valores cuantitativos o continuos en $\Re$ del espacio $\mathcal{Y} \subset \Re^K$.



$$f : \mathcal{X} \rightarrow \mathcal{Y} \subset \Re^K$$
$$x_n \mapsto \hat{\mathbf{y}}_n = f(x_n)$$

T: Regresión Múltiple

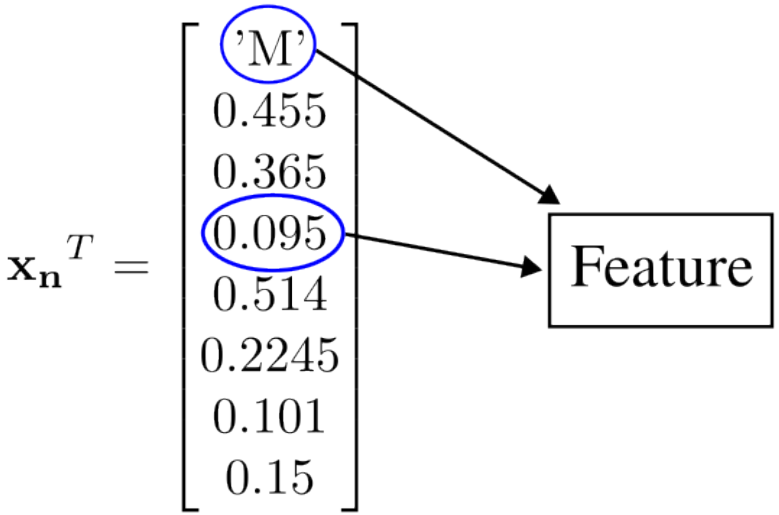
- Una regresión múltiple se puede transformar en K regresiones simples.
- Pero! En una regresión múltiple pueden existir una *correlaciones* entre las dimensiones del espacio $\mathcal{Y}$.

¿Pero qué ocurre en el caso cuando $\mathbf{x_n} \in \Re^D$ con $D > 1$?

- **Tenemos una tarea de predicción estructurada**

Sobre Espacios Típicos $\mathcal{X}$

- La mayoría de los métodos predicción van a considerar *Input Values* del tipo D-dimensional, tal que $\mathcal{X} \subset \Re^{N \times D}$, y donde cada input será un vector del tipo $\mathbf{x_n} \in \Re^D$.[3]
- Cada elemento D del vector $\mathbf{x_n}$ será llamado **feature** o **característica**.



[3] Recuerden que $n = 1, 2, \cdots, N$, y donde n se refiere a la fila n-ésima de un dataframe.

Representación tipo DataFrame

Feature 2 Feature 5 Feature 8

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.1500	
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.0700	$\mathbf{x_1}$
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.2100	
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.1550	$\mathbf{x_3}$
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.0550	

$$\mathcal{X} \subset \Re^{5\times 8} = \begin{bmatrix} \text{M} & 0.455 & 0.365 & 0.095 & 0.514 & 0.2245 & 0.101 & 0.15 \\ \text{M} & 0.35 & 0.265 & 0.09 & 0.2255 & 0.0995 & 0.0485 & 0.07 \\ \text{F} & 0.53 & 0.42 & 0.135 & 0.677 & 0.2565 & 0.1415 & 0.21 \\ \text{M} & 0.44 & 0.365 & 0.125 & 0.516 & 0.2155 & 0.114 & 0.155 \\ \text{I} & 0.33 & 0.255 & 0.08 & 0.205 & 0.0895 & 0.0395 & 0.055 \end{bmatrix}$$

T: Predicación Estructurada

- Problema que busca predecir un output que NO es un número continuo, una categoría, o un vector; sino un conjunto de valores relacionados entre ellos.

$$f : \mathcal{X} \subset \Re^* \rightarrow \mathcal{Y} \subset \Re^*$$

$$\mathbf{x}_n \mapsto \hat{\mathbf{y}}_n = f(\mathbf{x}_n)$$

T: Predicación Estructurada

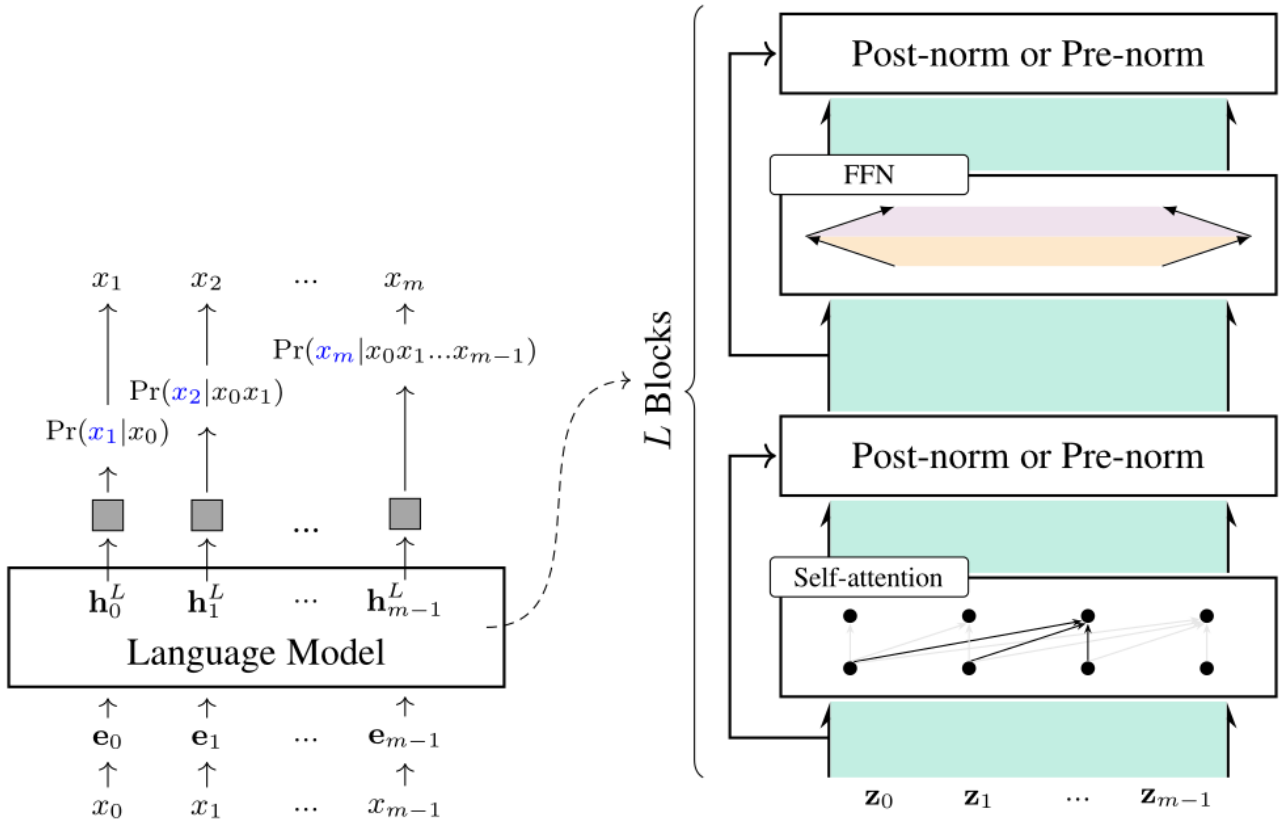
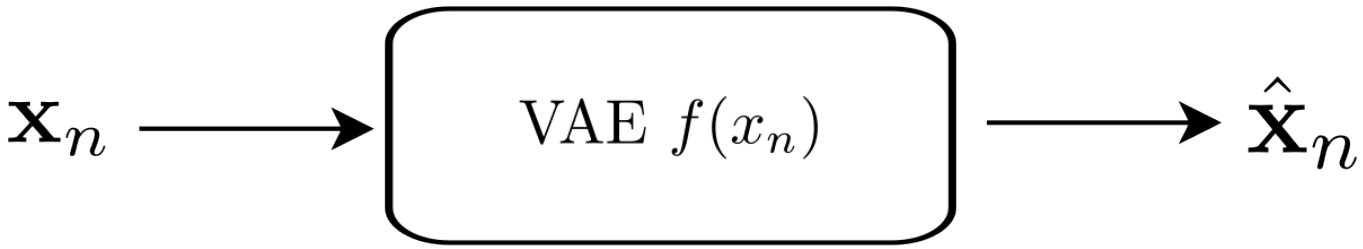


Fig.: Arquitectura Transformer-decoder para el procesamientos e lenguaje natural.[4]

[4] Xiao, T., & Zhu, J. (2025). Foundations of Large Language Models (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2501.09223

T: Predicación Estructurada

- Problema que busca predecir un output que NO es un número continuo, una categoría, o un vector; sino un conjunto de valores relacionados entre ellos.



$$f : \mathcal{X} \subset \Re^* \rightarrow \mathcal{X} \subset \Re^*$$
$$\mathbf{x}_n \mapsto_\epsilon \hat{\mathbf{x}}_n = f(\mathbf{x}_n)$$

T: Predicación Estructurada

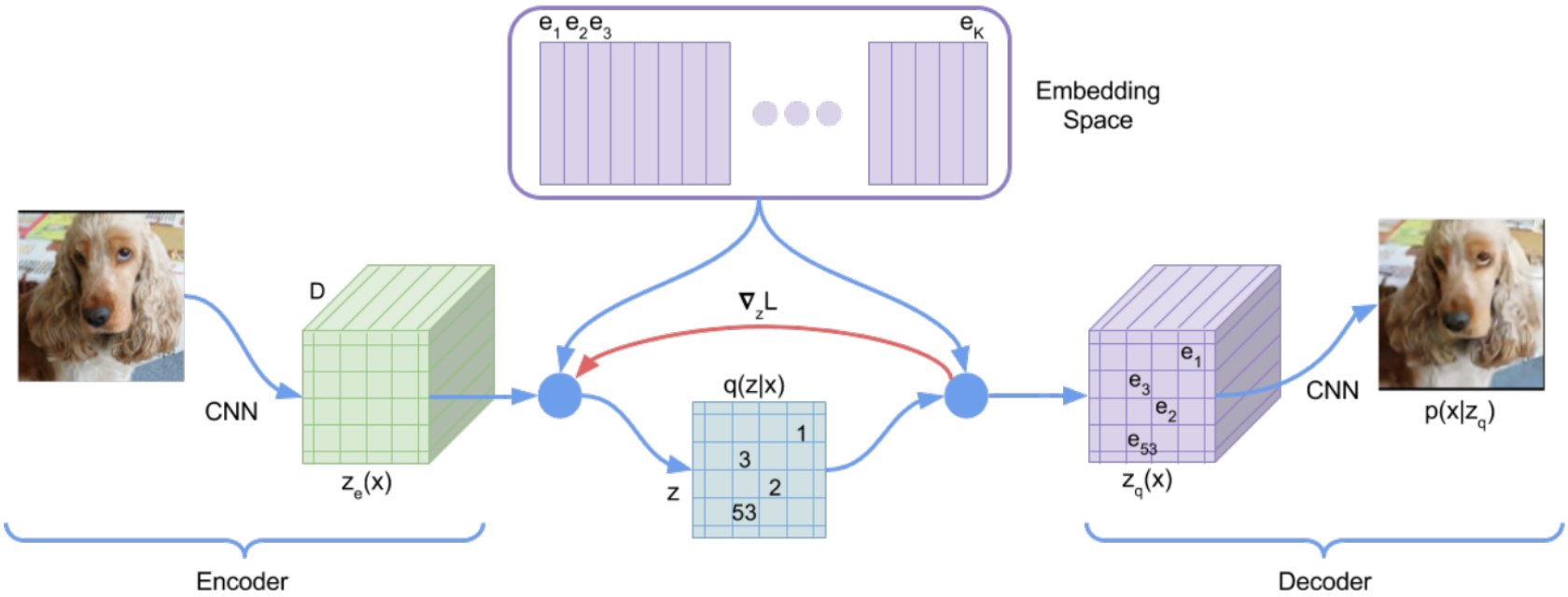
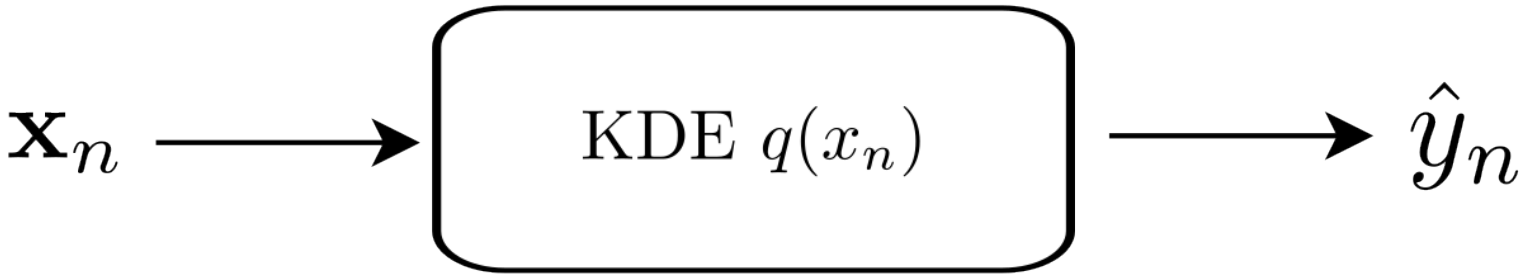


Fig.: Arquitectura VQ-VAE.[5]

[5] Oord, A. van den, Vinyals, O., & Kavukcuoglu, K. (2017). Neural Discrete Representation Learning (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1711.00937

T: Predicación Estructurada

- Problema que busca predecir un output que es una densidad de probabilidad.



$$f : \mathcal{X} \subset \Re^* \rightarrow [0, 1]$$

$$\mathbf{x}_n \mapsto \hat{y}_n = q(\mathbf{x}_n)$$

T: Predicación Estructurada

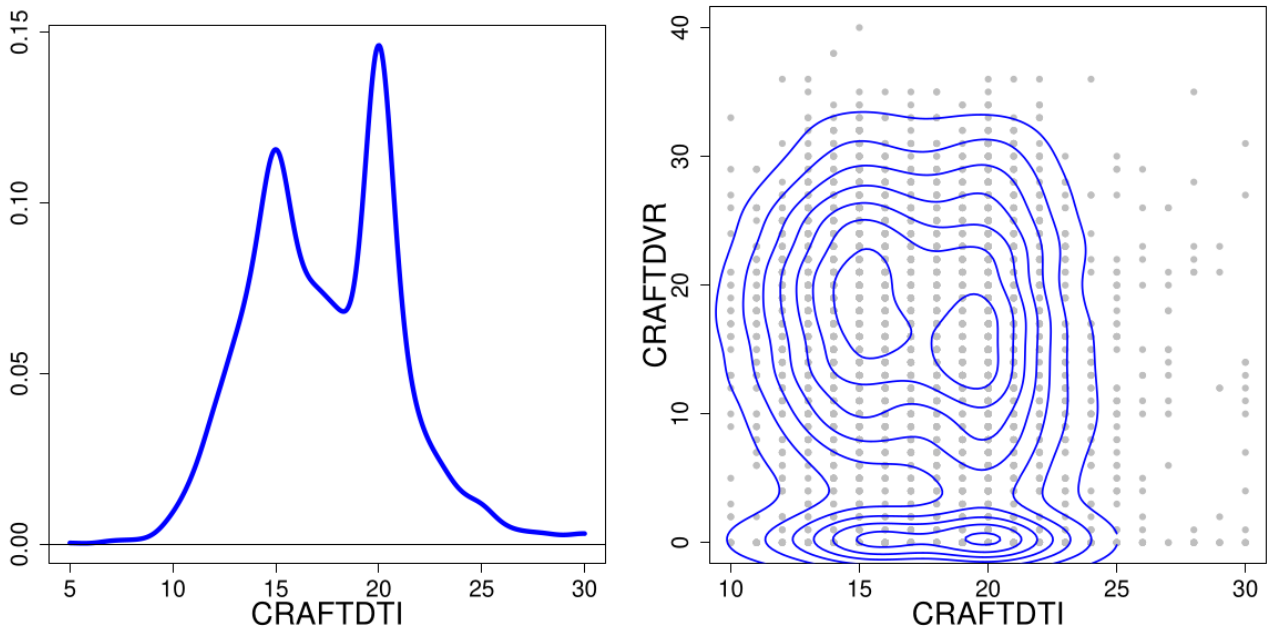


Fig.: Ejemplo de la aplicación de KDE sobre el dataset NACC.[6]

[6] Chen, Y.-C. (2017). A Tutorial on Kernel Density Estimation and Recent Advances (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1704.03924

Definición Training Experience E

Definición Training Experience *E*

Es aquella **información** que se le proporciona al programa durante la fase de **entrenamiento**, en orden de optimizar la solución al **problema**.

- La información entregada corresponde al conjunto de datos que representan ejemplos de una solución esperada al problema.
- Lo llamaremos dataset de entrenamiento o training set S.

Aprendizaje Supervisado

[a] Se dispone de un conjunto de N inputs con el respectivo valor de la solución al problema.

$$\mathcal{S} = \{\mathbf{x_n},\ y_n\}_{n=0}^{N} \coloneqq \{(\mathbf{x_0},\ y_0), (\mathbf{x_1},\ y_1), \cdots, (\mathbf{x_N},\ y_N)\},$$

donde $\mathbf{x_n} \in \Re^D$, $D \geq 1$, e $y_n \in \Re$.

▷ Supuestos típico para las tareas de regresión y clasificación.
▷ Es importante que $\{\mathbf{x_n},\ y_n\} \sim$ idd.

[a] En futuras definiciones, vamos a asumir por defecto siempre que se trata de un entrenamiento supervenido, a menos que se diga lo contrario.

Aprendizaje Supervisado

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight		Rings
17	F	0.440	0.340	0.100	0.4510	0.1880	0.0870	0.130	**17**	10
1131	M	0.565	0.435	0.150	0.9900	0.5795	0.1825	0.206	**1131**	8
299	M	0.370	0.280	0.105	0.2340	0.0905	0.0585	0.075	**299**	9
1338	M	0.580	0.455	0.135	0.7955	0.4050	0.1670	0.204	**1338**	10
2383	F	0.525	0.390	0.135	0.6005	0.2265	0.1310	0.210	**2383**	16

Fig.: Abalone Dataset. Predecir la **edad** de los abalones a partir de características físicas. 1995.

	sepal length	sepal width	petal length	petal width		class
14	5.8	4.0	1.2	0.2	**14**	Iris-setosa
98	5.1	2.5	3.0	1.1	**98**	Iris-versicolor
75	6.6	3.0	4.4	1.4	**75**	Iris-versicolor
16	5.4	3.9	1.3	0.4	**16**	Iris-setosa
131	7.9	3.8	6.4	2.0	**131**	Iris-virginica

Fig.: Iris Dataset. Predecir la **clase** de las plantas Iris a partir de características físicas. Ronald Fisher 1936.

Aprendizaje No Supervisado

Se dispone de un conjunto de N inputs SIN el respectivo valor de la solución al problema.

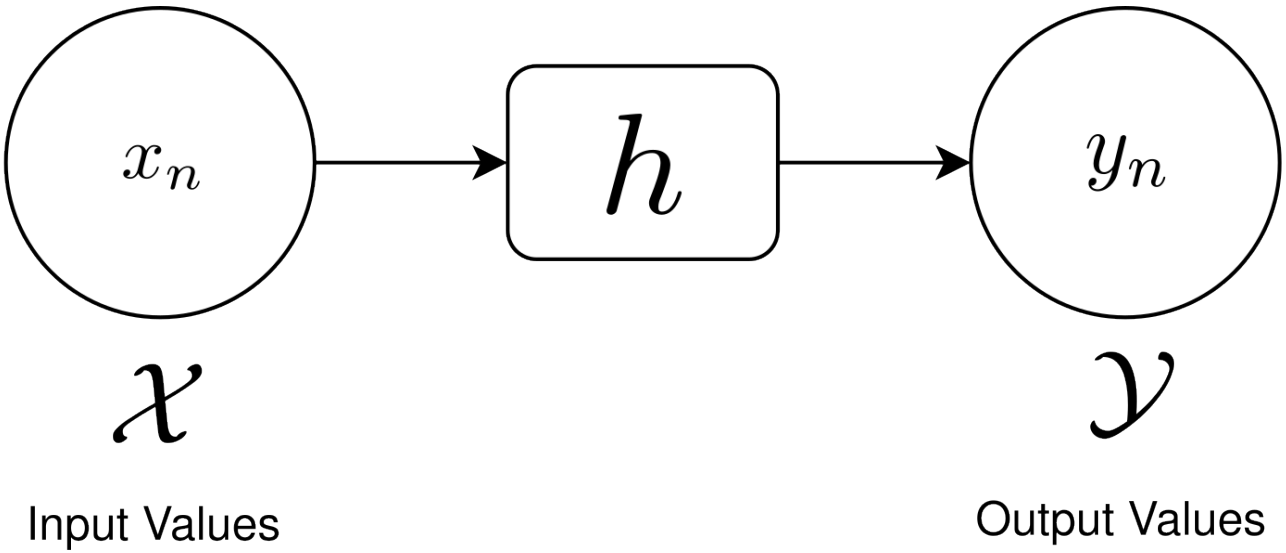
$$\mathcal{S} = \{\mathbf{x_n}\}_{n=0}^{N} := \{(\mathbf{x_0}), (\mathbf{x_1}), \cdots, (\mathbf{x_N})\},$$

donde $\mathbf{x_n} \in \Re^D$, $D \geq 1$.

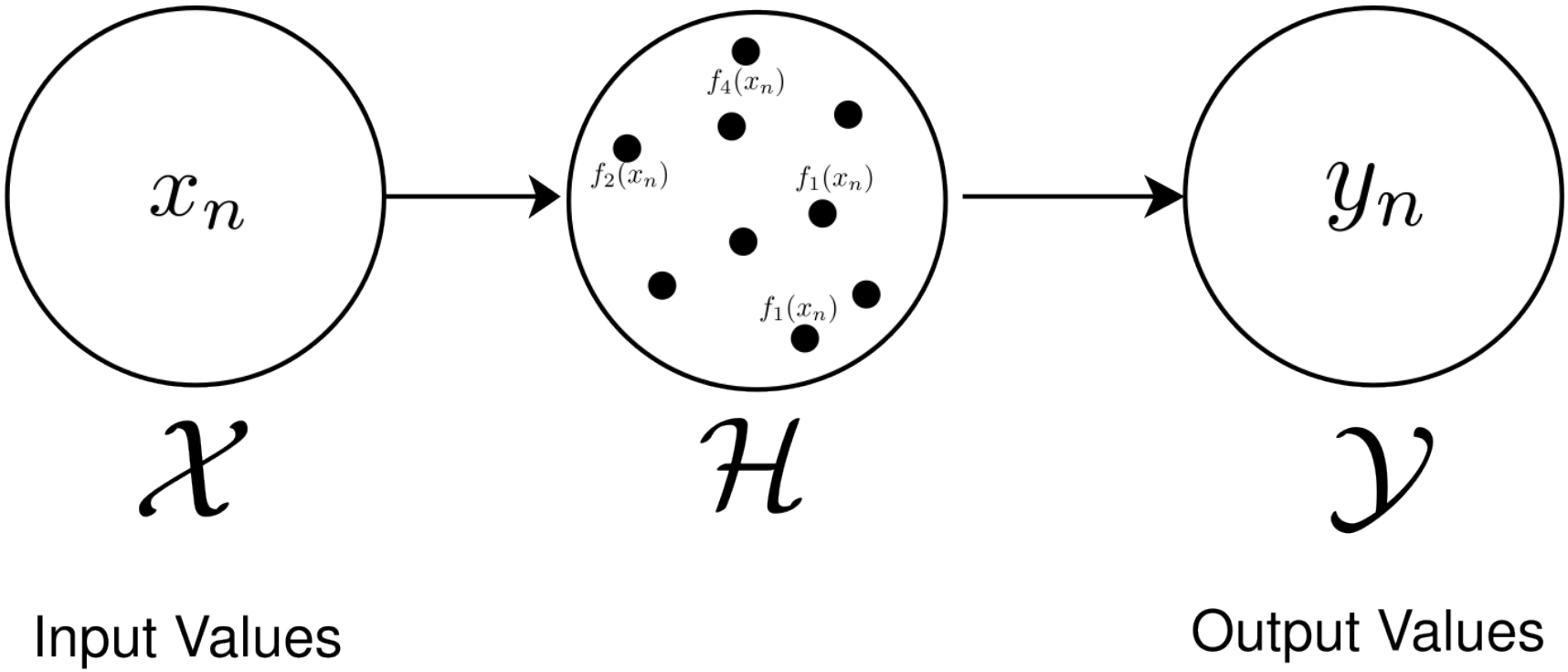
▷ Supuestos típico paras las tareas de detección de anomalías, reconstrucción de imágenes, y estimación de densidades de probabilidad.

Sobre la *Hipotiposis*

- Para aproximar la función desconocida, nuestra máquina debe observar casos del tipo input-output.
- Dado un input, la maquinas tratara de predecir el output más correcto.

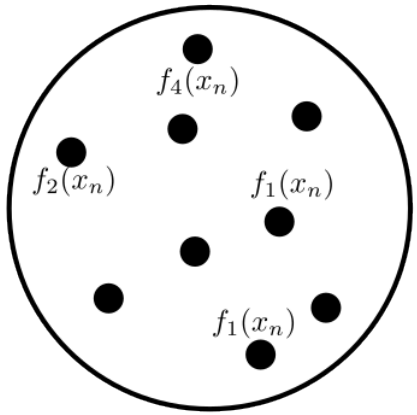


- La función que busca nuestra máquina será la **hipotiposis**.


x_n
$f_4(x_n)$
$f_2(x_n)$
$f_1(x_n)$
$f_1(x_n)$
y_n
$\mathcal{X}$
$\mathcal{H}$
$\mathcal{Y}$
Input Values
Output Values

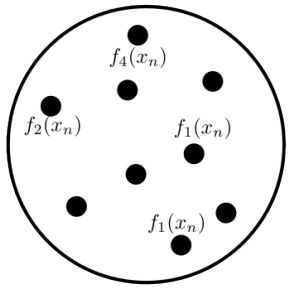
Espacio $\mathcal{H}$

Conjunto de todas las **posibles soluciones** dadas por las **funciones** que la máquina puede implementar para el problema dado.



$$\mathcal{H}$$

$$\mathcal{H} = \{f(\mathbf{x}_n, \mathbf{w}); \left(\mathbf{x_n} \in \Re^D\right) \wedge \left(\mathbf{w} \in \Lambda\right) \wedge \left(\Lambda \subset \Re^*\right)\}$$



$$\mathcal{H}$$

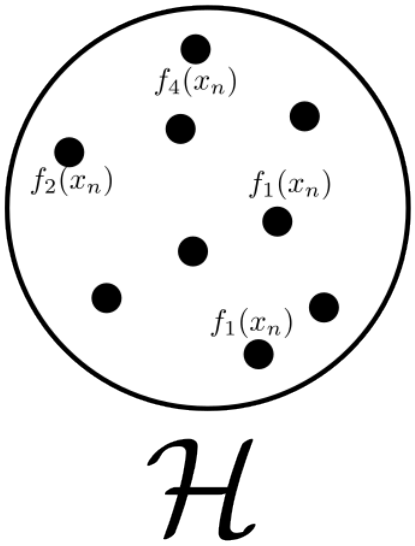
$$\mathcal{H} = \{f(\mathbf{x}_n, \mathbf{w}); \left(\mathbf{x_n} \in \Re^D\right) \wedge (\mathbf{w} \in \Lambda) \wedge (\Lambda \subset \Re^*)\}$$

- El espacio $\mathcal{H}$ esta **parametrizado**, y por lo tanto, cada función $f(\mathbf{x}_n, \mathbf{w})$ queda identificada por un conjunto finito de parámetros $\mathbf{w}$.
- De este modo, vamos a definir $\mathbf{w}$ como los **parámetros del modelo** y conjunto Λ como el espacio de parámetros.
- En la práctica, la máquina no va trabajar sobre función $f(*)$, sino sobre el espacio de parámetros Λ.

Definición Performance Measure P

Definición Performance Measure *P*

Es aquella función R sobre el espacio de hipótesis $\mathcal{H}$ que permite **medir cuantitativamente** la **calidad** de la función $f(\mathbf{x}_n, \mathbf{w})$, implementada por la máquina.

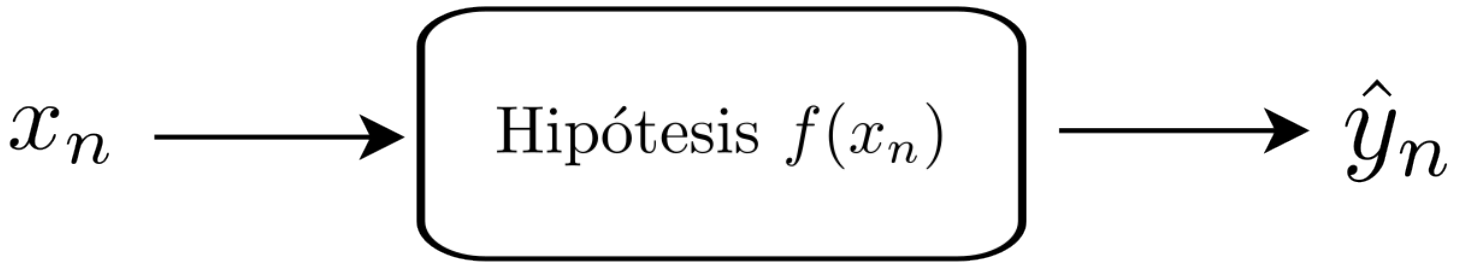


$$\mathcal{H} = \{f(\mathbf{x}_n, \mathbf{w}); \mathbf{w} \in \Lambda\}$$

$$\boxed{R : \mathcal{H} \rightarrow \Re}$$

Definición Función de Perdida ℓ

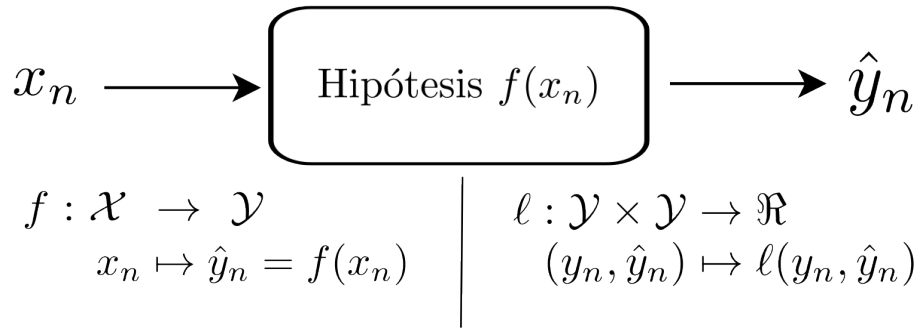
La funcion ℓ permite medir cuantitativamente la calidad de la hipótesis dada por $\hat{y} = f(x_n)$, la cual implementa la maquina para un correspondiente input x_n y su posible respuesta $\hat{y}$



$$f : \mathcal{X} \rightarrow \mathcal{Y} \qquad x_n \mapsto \hat{y}_n = f(x_n)$$

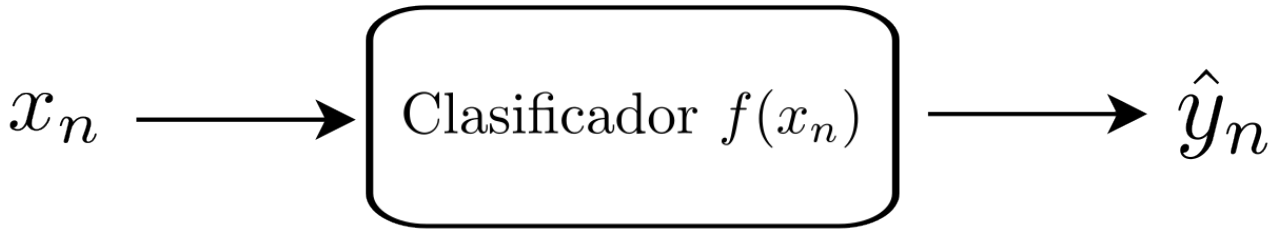
$$\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \Re \qquad (y_n, \hat{y}_n) \mapsto \ell(y_n, \hat{y}_n)$$

Definición Función de Perdida ℓ



- La función ℓ suele ser conocida como: *loss function*.
- La etiqueta *y* suele ser llamada *ground truth*.
- Por lo general ocurre que $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \Re_0^+$.
- $> \ell$ equivale a menor desempeño, mientra que $< \ell$ equivale a un mejor desempeño.

Task: Clasificación



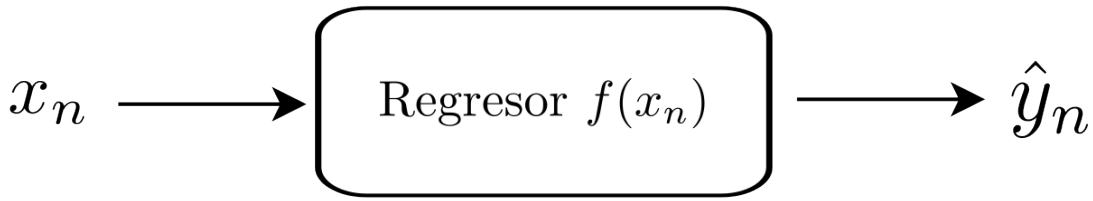
$$f : \mathcal{X} \rightarrow \mathcal{Y} = \{c_0, c_1, \cdots, c_k\}$$
$$x_n \mapsto \hat{y}_n = f(x_n)$$

Misclassification Loss[7]

$$\ell(y, \hat{y}) = \mathbb{I}(y \neq \hat{y}) = \begin{cases} 0 & \text{si } y = \hat{y}, \\ 1 & \text{si } y \neq \hat{y}. \end{cases}$$

[7] Donde $\mathbb{I}(*)$ es la función indicatriz.

Task: Regresión



$$f : \mathcal{X} \rightarrow \mathcal{Y} \subset \Re$$
$$x_n \mapsto \hat{y}_n = f(x_n)$$

Squered Loss

$$\ell(y, \hat{y}) = (y - \hat{y})^2$$

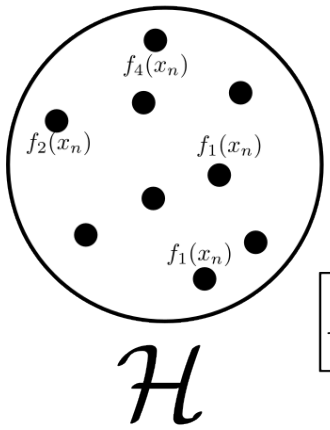
Epsilon Insensitive Loss

$$\ell(y, \hat{y}) = \begin{cases} 0 & \text{si } |y - \hat{y}| \leq \epsilon, \\ |y - \hat{y}| - \epsilon & \text{etoc.} \end{cases}$$

Definición Performance Measure *P*

Es aquella función R sobre el espacio de hipótesis $\mathcal{H}$ que permite **medir cuantitativamente** la **calidad** de la función $f(\mathbf{x}_n, \mathbf{w})$, implementada por la máquina.

▷ Dado los N valores de la evaluación $\ell(y_n, \hat{y}_n)$, con $n = 0, 1, \cdots, N$, el desempeño global de la máquina viene dado por la función de agregación R.



$$\mathcal{H} = \{f(\mathbf{x}_n, \mathbf{w}); \mathbf{w} \in \Lambda\}$$

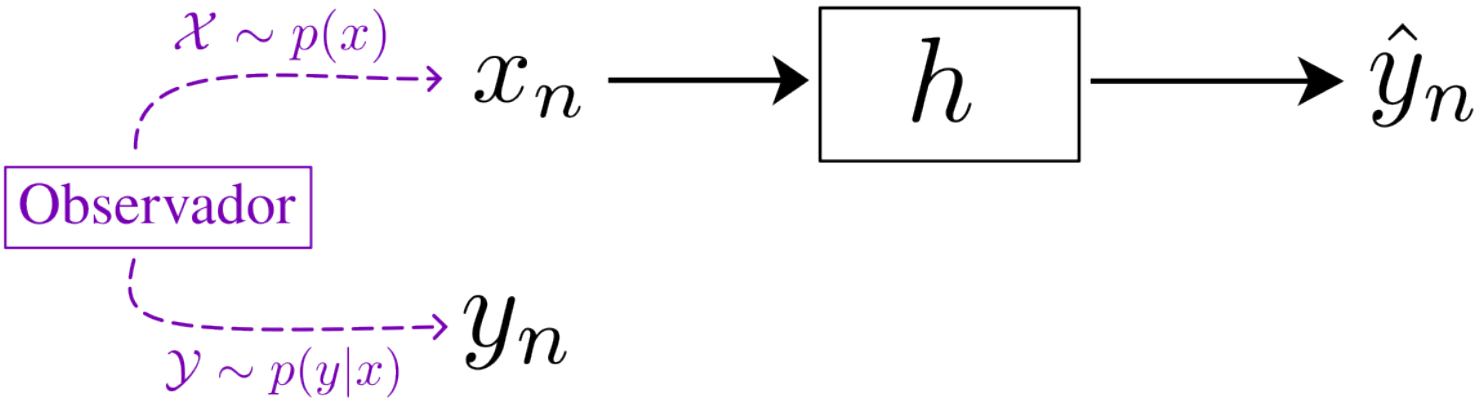
$$\boxed{R : \mathcal{H} \rightarrow \Re}$$

$$\boxed{R(f) = \mathbb{E}\left[\ell\left(y_n, f\left(x_n\right)\right)\right]}$$

Expected Risk

Cual es el valor $\mathbb{E}$ *de un conjunto iid ?*

Cual es el valor $\mathbb{E}$ *de un conjunto iid ?* $\Rightarrow$ la media



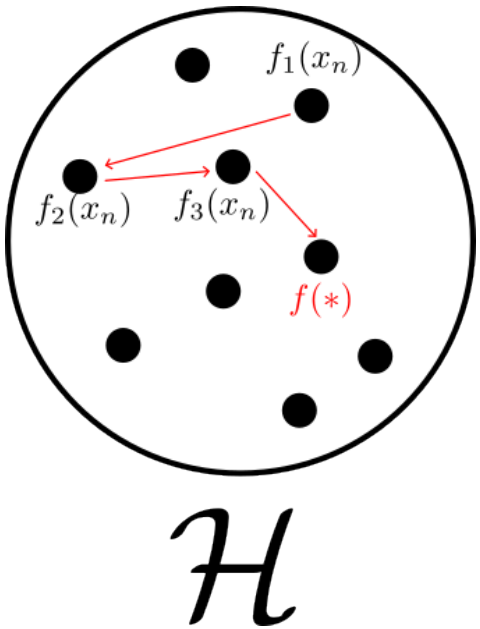
$$\mathbb{E}_{\mathcal{X},\mathcal{Y}}\left[\ell\left(y,\ f\left(x\right)\right)\right] = \int_{\mathcal{X},\mathcal{Y}} \ell\left(y,\ f\left(x\right)\right) \cdot p\left(y,\ f\left(x\right)\right) dx\, dy$$

$$\mathbb{E}_{\mathcal{X},\mathcal{Y}}\left[\ell\left(y,\ f\left(x\right)\right)\right] = \sum_{x\in\mathcal{X},y\in\mathcal{Y}} \ell\left(y,\ f\left(x\right)\right) \cdot p\left(y,\ f\left(x\right)\right)$$

Minimización del **Expected Risk**

Buscamos una función $f(*)$ tal que minimice el **Expected Risk** $\mathbb{E}$.

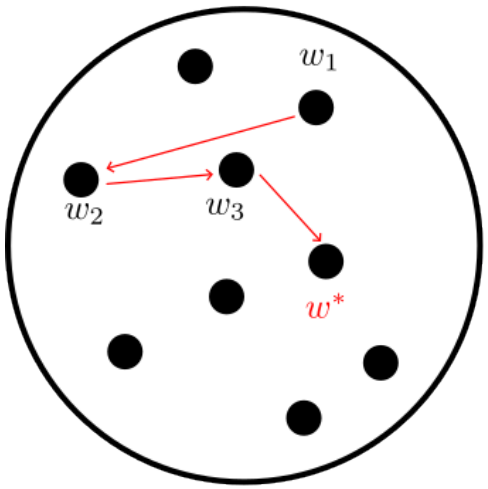
$$\min R(f) = \mathbb{E}\left[\ell\left(y, f\left(x\right)\right)\right] \text{ sujeto a } f \in \mathcal{H}$$



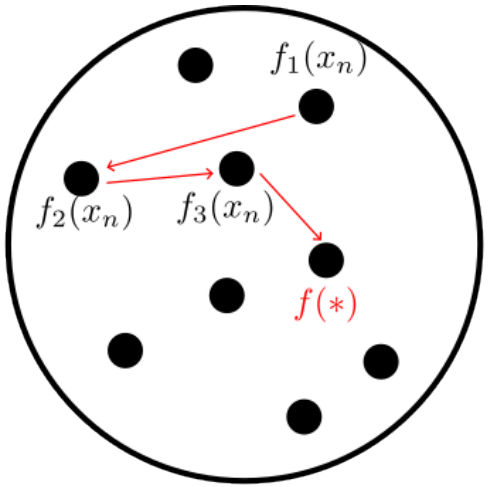
Minimización del **Expected Risk**

Buscamos el vector de paramentos $\mathbf{w}^*$ para la función $f(x, \mathbf{w}^*)$ tal que minimice el **Expected Risk** $\mathbb{E}$.

$$\min R(\mathbf{w}) = \mathbb{E}\left[\ell\left(y, f\left(x, \mathbf{w}\right)\right)\right] \text{ sujeto a } \mathbf{w} \in \Lambda$$



Λ



$$\mathcal{H} = \{f(x, \mathbf{w}); \mathbf{w} \in \Lambda\}$$