

Modelo de Aprendizaje Supervisado

INF-393 Aprendizaje Automático
2-2018 DI UTFSM

Contenidos

- Formalización del problema de aprendizaje.
- Definiciones y conceptos básicos.
- Problemas fundamentales.

Definición



“A program is said to learn from experience **E** with respect to some task **T** and some performance measure **P**, if its performance on **T**, as measured by **P**, improves with experience **E**”

Tom Mitchell, *Machine Learning*, 1997.

Machine Learning Basics

Tarea (T)

Problema que queremos que el programa resuelva.

- Ejemplos:
 - Determinar si una opinión es positiva o negativa.
 - Detectar la presencia de una cara en una imagen.
 - Describir una imagen.
 - Auto-completar una frase.

Machine Learning Basics

- **Clasificación:** problema en que queremos que el programa determine la categoría o etiqueta \mathbf{y} que corresponde a un determinado input \mathbf{x} de entre un conjunto predefinido de posibilidades.



\mathcal{X} input space

\mathcal{Y} output space

$$f : \mathcal{X} \longrightarrow \mathcal{Y} = \{c_1, \dots, c_k\}$$

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

hipótesis implementada
por el programa

Machine Learning Basics

- **Multi-label classification:** en este caso input x puede pertenecer a más de una categoría simultáneamente.



\mathcal{X} input space

$\mathcal{Z} = 2^{\mathcal{Y}}$ output space

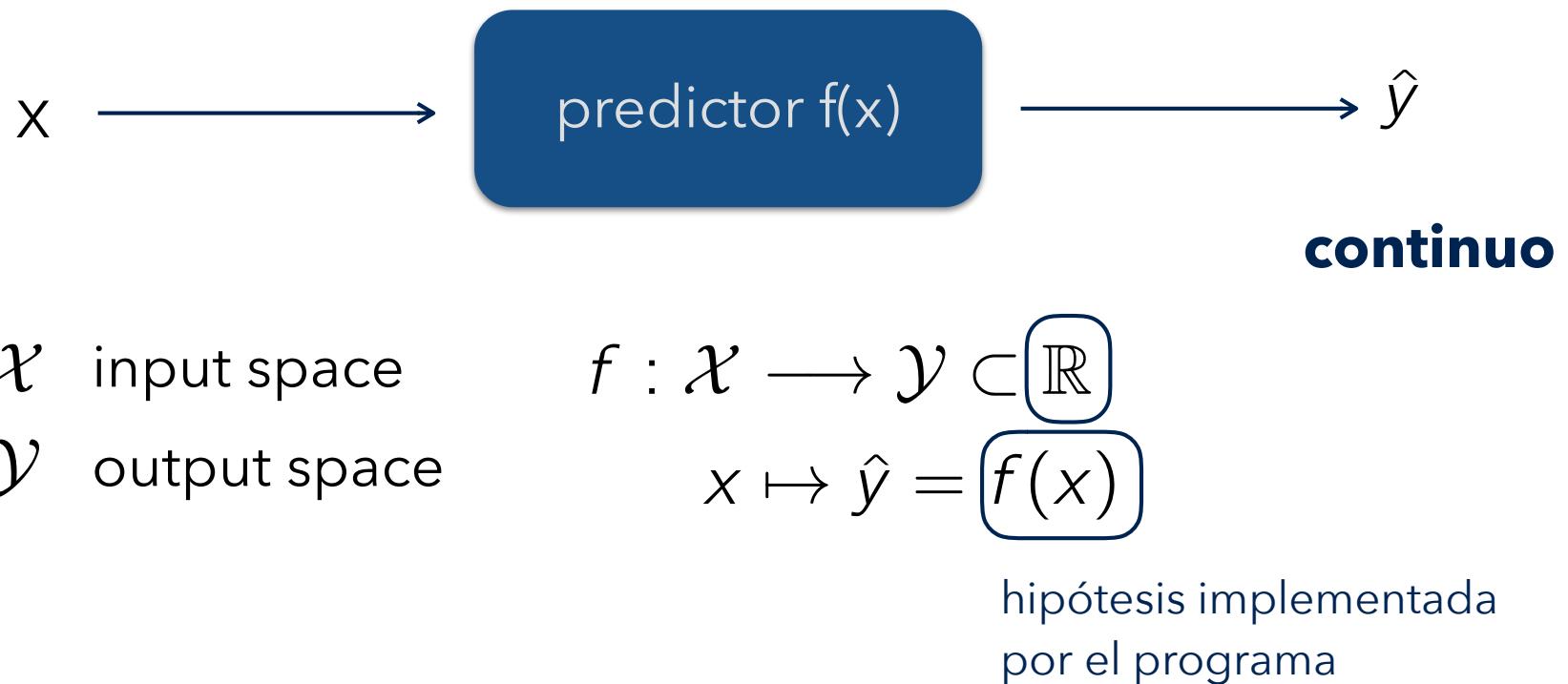
$$f : \mathcal{X} \longrightarrow 2^{\mathcal{Y}}$$

$$x \mapsto \{c_{i_1}, \dots, c_{i_{k_i}}\} = f(x)$$

hipótesis implementada
por el programa

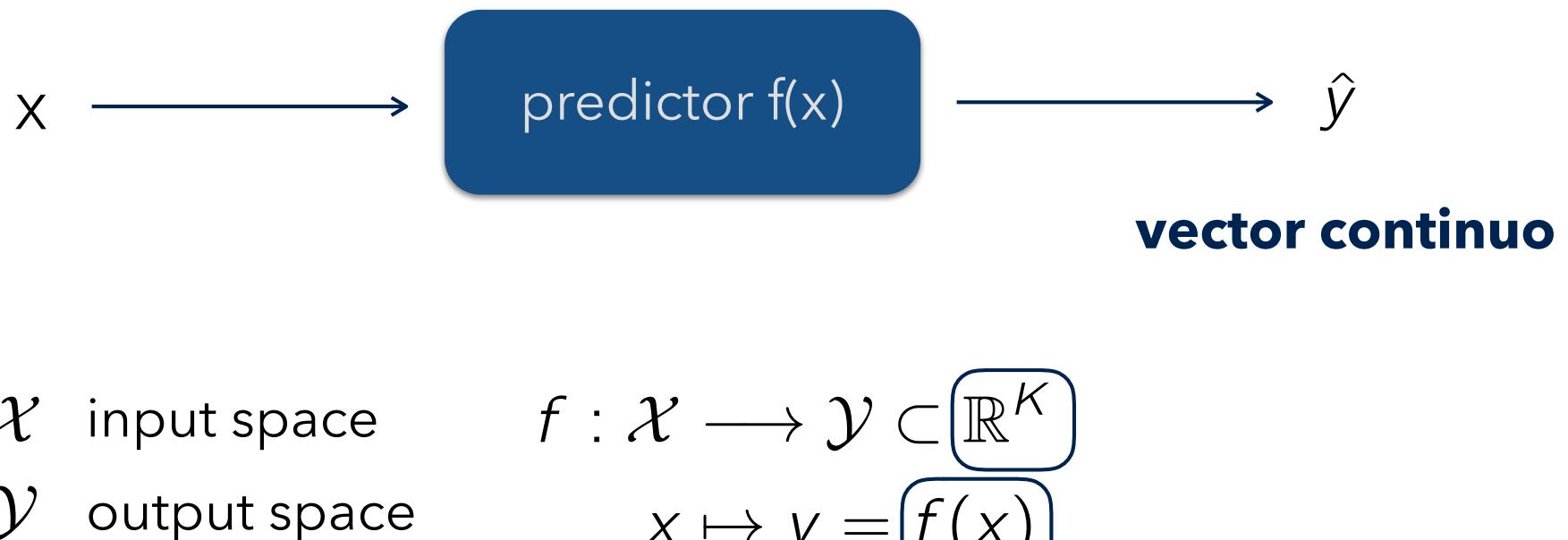
Machine Learning Basics

- **Regresión:** problema en que queremos que el programa prediga un valor numérico (e.g. distancia, velocidad, aceleración, tiempo, precio) \mathbf{y} asociado a \mathbf{x} .



Machine Learning Basics

- **Multi-output regression:** problema en que queremos que el programa prediga una tupla de valores numéricos.

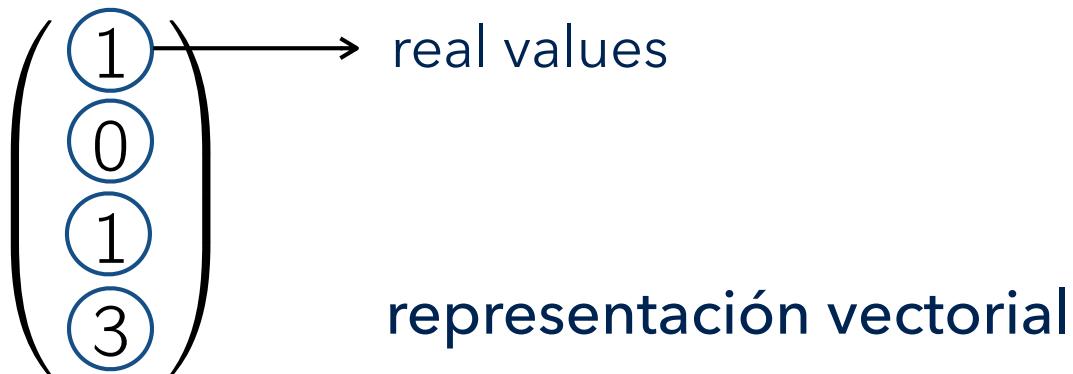


hipótesis implementada
por el programa

Machine Learning Basics

- **Espacio de entrada típico.** La gran mayoría de los modelos/ métodos asumen que el input al sistema (\mathbf{x}) es un vector de atributos numéricos de dimensionalidad fija.

$$\mathcal{X} \subset \mathbb{R}^d \quad d : \text{dimensionalidad}$$



Machine Learning Basics

- **Espacio de entrada típico.** En muchos problemas prácticos el input al sistema es pobremente estructurado y se requiere algún tipo de **representación**.

SCENE FROM “ DAN'L DRUCE.”

This interesting domestic drama, by Mr. W. S. Gilbert, has continued to engage the sympathies of a nightly sufficient audience at the Haymarket Theatre, where it has now been represented more than sixty times. Its subject and character were described by us, in the ordinary report of theatrical novelties, about two months ago. Our readers will probably not need to be reminded that the hero of the story, Dan'l Druce, the blacksmith, is a solitary recluse dwelling on the coast of Norfolk, where his lone cottage is visited by fugitives from party vengeance during the civil wars of the Commonwealth. His hoard of money is stolen; but a different sort of treasure, a helpless female infant, is left by some mysterious agency, and may be accepted, as in George Eliot's tale of “Silas Marner,” for a Divine gift to the sad-hearted misanthrope, far better than riches. In this spirit, at least, he is content to receive the precious human charge; and so to those who would remove it from his home, Dan'l Druce here makes answer with the solemn exclamation, “Touch not the Lord's gift!” This character is well acted by Mr. Hermann Vezin.

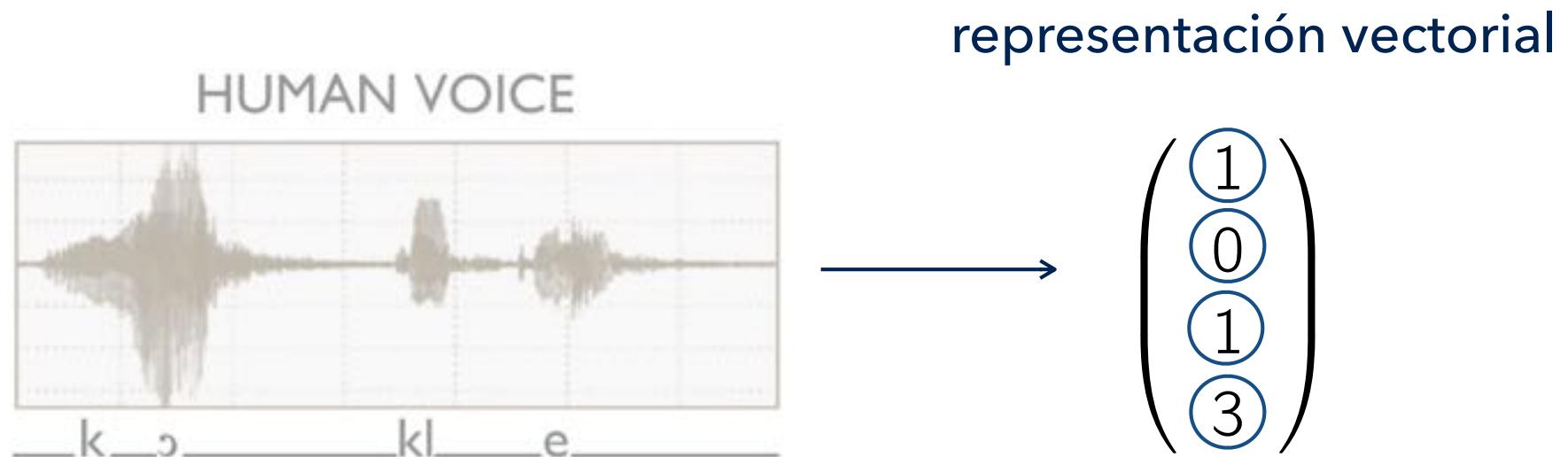
representación vectorial



$$\begin{pmatrix} 1 \\ 0 \\ 1 \\ 3 \end{pmatrix}$$

Machine Learning Basics

- **Espacio de entrada típico.** En muchos problemas prácticos el input al sistema es pobemente estructurado y se requiere algún tipo de **representación**.



Machine Learning Basics

- **Predicción Estructurada:** se desea “transformar” el input en un conjunto de varios ($>>1$) valores relacionados entre sí de modo relevante semánticamente.

Inglés▼

I love this university

x

Español▼

Me encanta esta
universidad

y

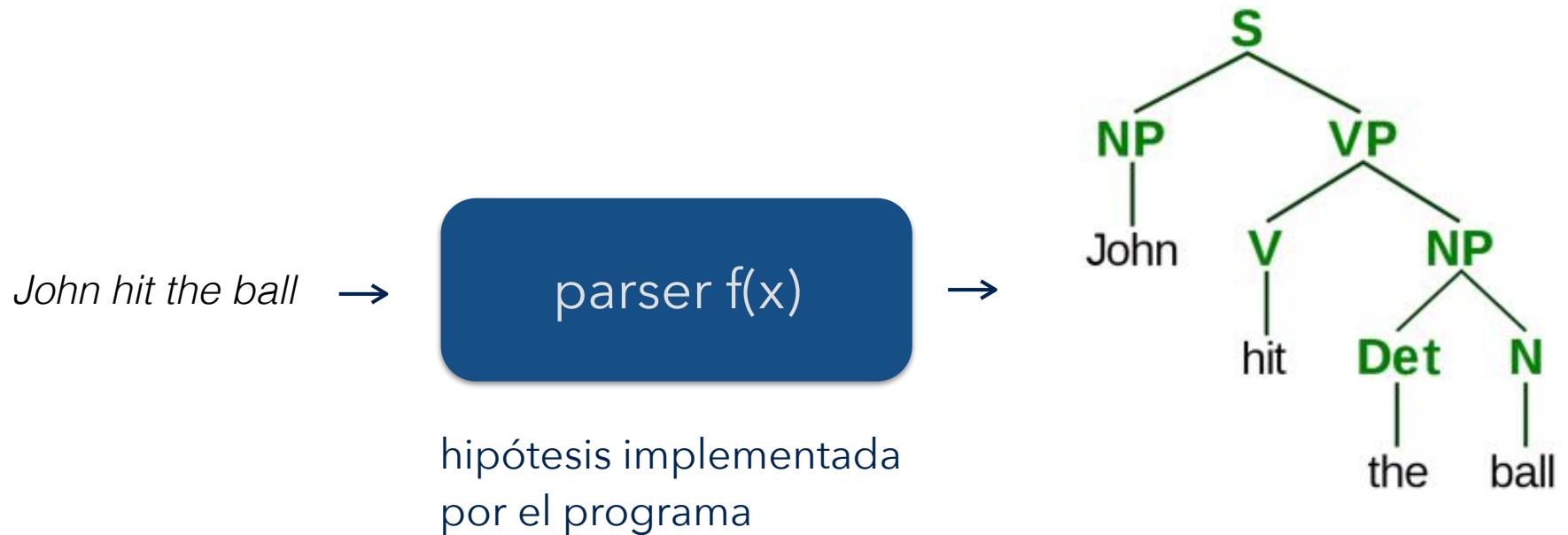
traductor $f(x)$

hipótesis implementada
por el programa

estructurado

Machine Learning Basics

- **Predicción Estructurada:** se desea “transformar” el input en un conjunto de varios ($>>1$) valores relacionados entre sí de modo relevante semánticamente.



Machine Learning Basics

- **Detección de Anomalías:** se desea que el programa lance una alarma frente a inputs inusuales ó atípicos.



→ detector $f(x)$



$$f : \mathcal{X} \longrightarrow \{\emptyset, \text{alarm}\}$$
$$x \mapsto \hat{y} = f(x)$$

Machine Learning Basics

- **Denoising:** Dada una señal corrupta se desea que el programa reconstruya la versión original



$$f : \mathcal{X} \longrightarrow \mathcal{X}$$

$$\tilde{x} \mapsto x = f(\tilde{x})$$



Machine Learning Basics

- **Completación de data faltante:** Dada una señal corrupta se desea que el programa reconstruya la versión original



$$f : \mathcal{X} \longrightarrow \mathcal{X}$$

$$\tilde{x} \mapsto x = f(\tilde{x})$$



Machine Learning Basics

- **Estimación de densidad de probabilidad:** se desea que el programa aproxime la densidad de probabilidad asociada a un input.



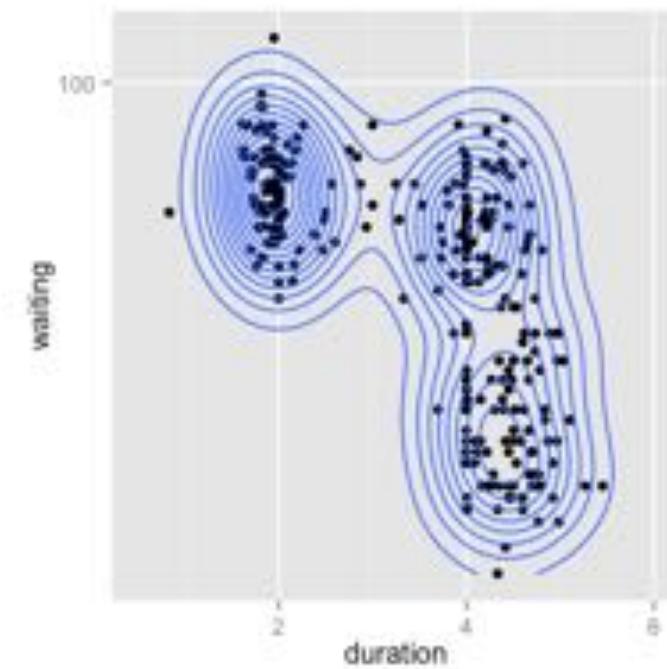
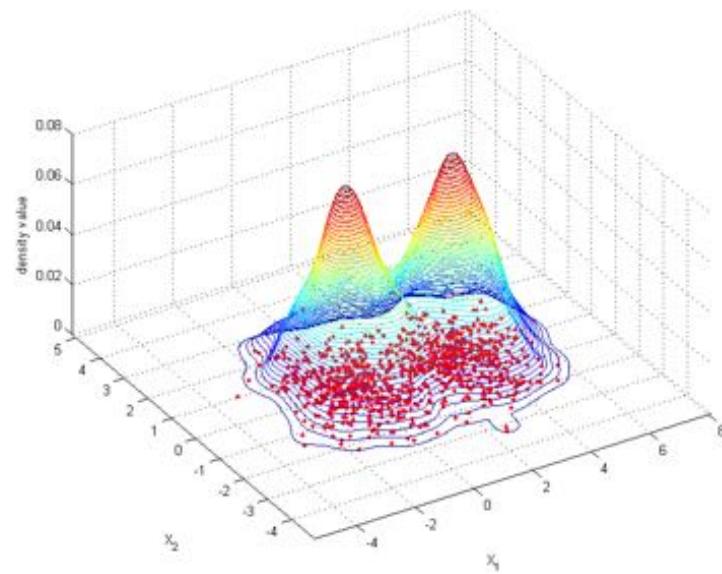
$$f : \mathcal{X} \longrightarrow [0, 1]$$

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

hipótesis implementada
por el programa

Machine Learning Basics

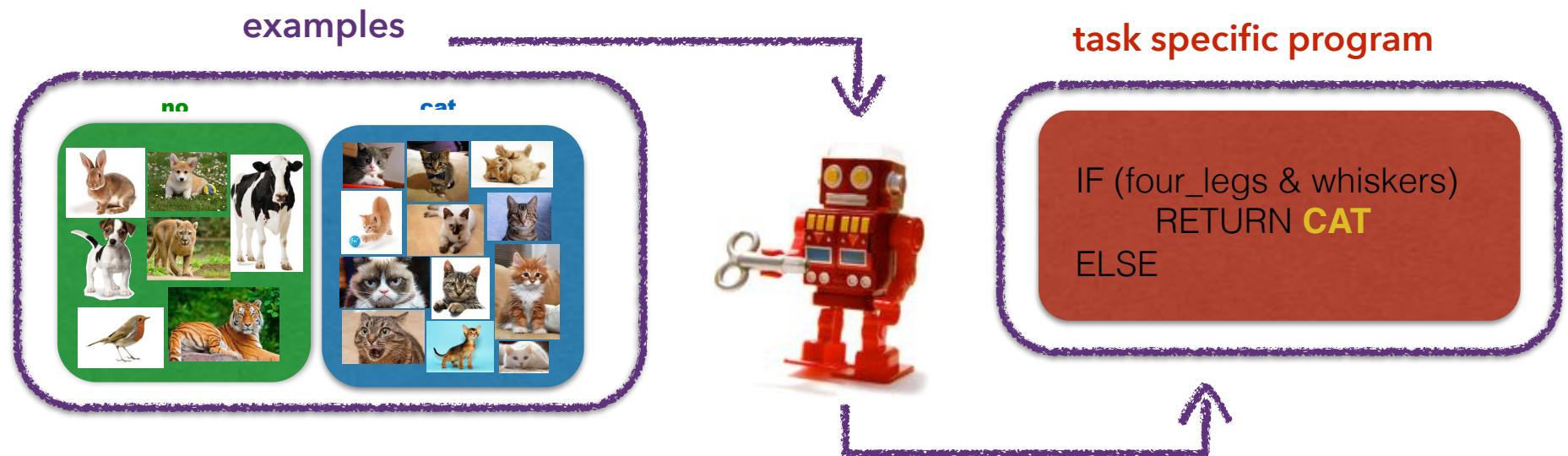
- **Estimación de densidad de probabilidad:** se desea que el programa aproxime la densidad de probabilidad asociada a un input.



Machine Learning Basics

Experiencia (E)

Información que se le proporcionan al programa durante la fase de entrenamiento para que construya, mejore o adapte su solución al problema. Típicamente un conjunto de datos que representan ejemplos de la solución deseada. Este conjunto se llama **conjunto de entrenamiento** o **conjunto de ejemplos S**.



Machine Learning Basics

Aprendizaje Supervisado

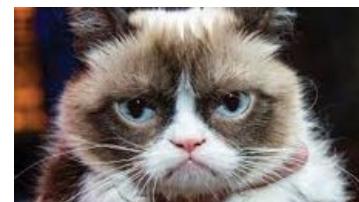
Se dispone de un conjunto de **n** inputs con la respectiva salida o respuesta deseada.

$$S = \{x_i, y_i\}_{i=1}^n \equiv \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$$

Supuesto típico en regresión y clasificación.



x_1



x_2



x_3

$y_1 = \text{'perro'}$

$y_2 = \text{'gato'}$

$y_3 = \text{'gato'}$

Machine Learning Basics

Abstracción típica

Name	Thread pitch (mm)	Minor diameter tolerance	Nominal diameter (mm)	Head shape	Price for 50 screws	Available at factory outlet?	Number in stock	Flat or Phillips head?
M4	0.7	4g	4	Pan	\$10.08	Yes	276	Flat
M5	0.8	4g	5	Round	\$13.89	Yes	183	Both
M6	1	5g	6	Button	\$10.42	Yes	1043	Flat
M8	1.25	5g	8	Pan	\$11.98	No	298	Phillips
M10	1.5	6g	10	Round	\$16.74	Yes	488	Phillips
M12	1.75	7g	12	Pan	\$18.26	No	998	Flat
M14	2	7g	14	Round	\$21.19	No	235	Phillips
M16	2	8g	16	Button	\$23.57	Yes	292	Both
M18	2.1	8g	18	Button	\$25.87	No	654	Both
M20	2.4	8g	20	Pan	\$29.09	Yes	486	Both
M24	2.55	9g	24	Round	\$33.01	Yes	982	Phillips
M28	2.7	10g	28	Button	\$35.66	No	1067	Phillips
M36	3.2	12g	36	Pan	\$41.32	No	434	Both
M50	4.5	15g	50	Pan	\$44.72	No	740	Flat

Ejemplo 1
Ejemplo 2

Ejemplo n

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1d} \\ x_{21} & x_{22} & \dots & x_{2d} \\ \dots \\ x_{n1} & x_{n2} & \dots & x_{nd} \end{pmatrix}$$

n x d real matrix

Machine Learning Basics

Aprendizaje NO supervisado

Se dispone de un conjunto de **n** inputs sin la respuesta óptima / correcta / deseada.

$$S = \{x_i\}_{i=1}^n \equiv \{(x_1, x_2, \dots, x_n)\}$$

Escenario típico en detección de anomalías, denoising, estimación de densidades de probabilidad e imputación de valores faltantes.

x_1



x_2



x_3



Machine Learning Basics

Aprendizaje SEMI supervisado

Se dispone de un conjunto de **n** inputs algunos con la respuesta óptima / correcta / deseada y otros (gran mayoría) no.

$$S = \{\{x_i\}_{i=1}^{n_1}, \{y_i\}_{i=1}^{n_2}\}$$

Escenario típico en la práctica.

x_1



$y_1 = \text{'perro'}$

x_2

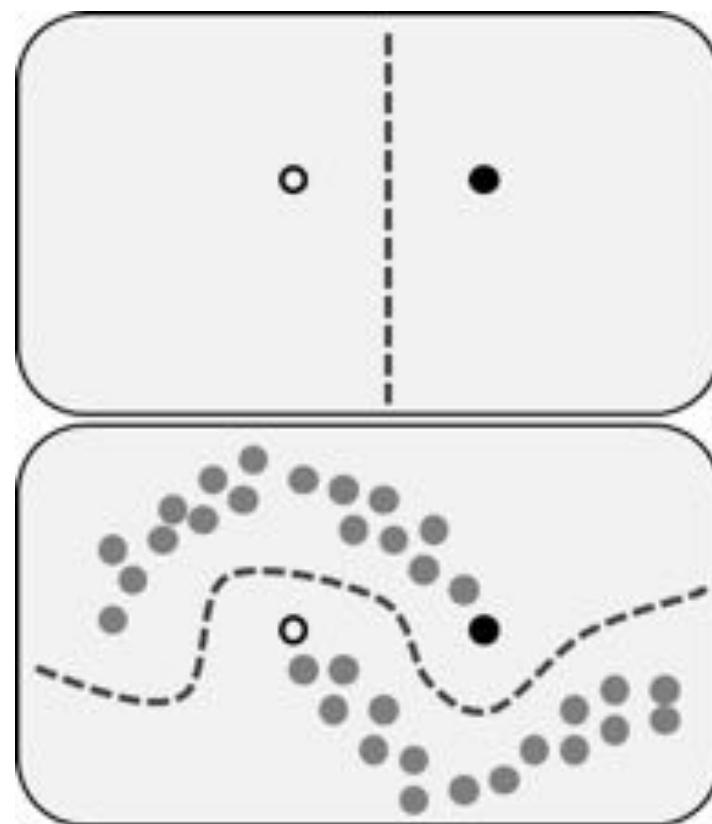


$y_2 = \text{'gato'}$

x_3



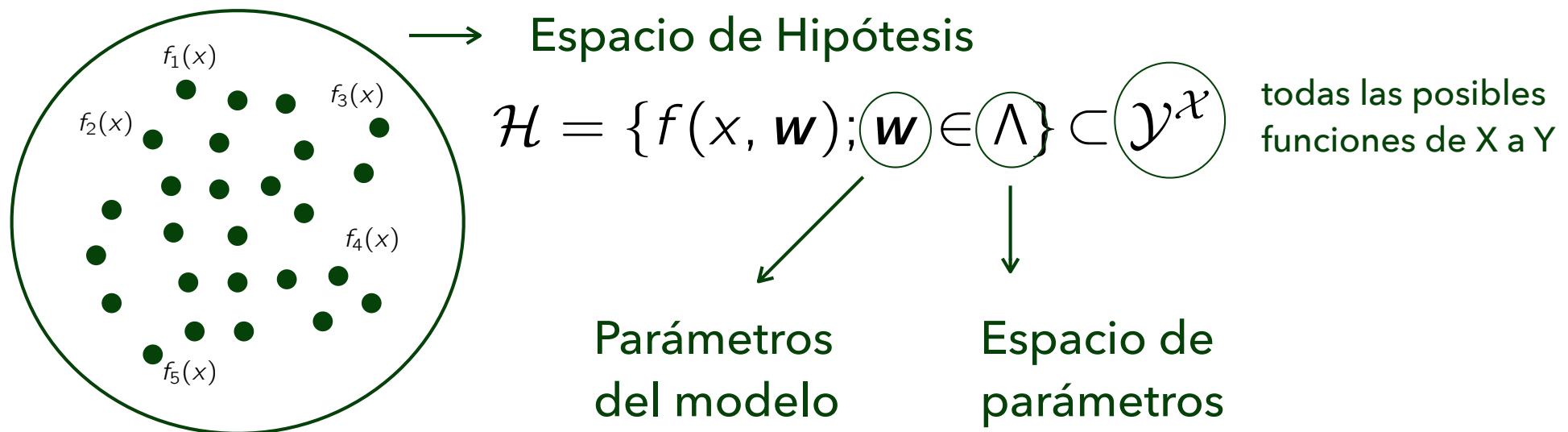
Machine Learning Basics



Machine Learning Basics

Medida de Desempeño (P)

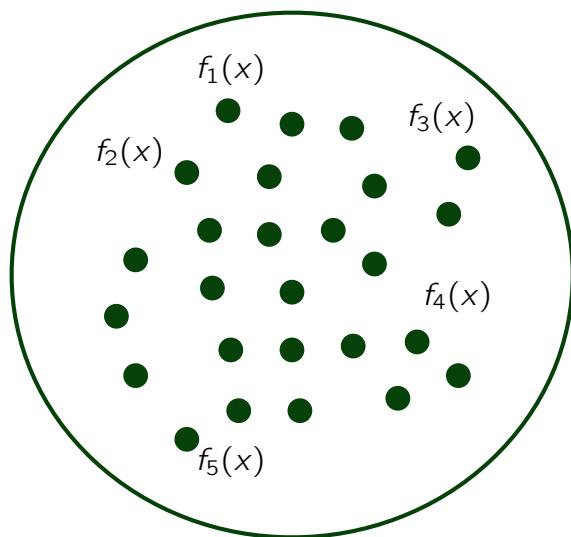
Funcional **R** que permite medir cuantitativamente la calidad de la función **f(x)** implementada por el programa.



Machine Learning Basics

Medida de Desempeño (P)

Funcional **R** que permite medir cuantitativamente la calidad de la función **f(x)** implementada por el programa.



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

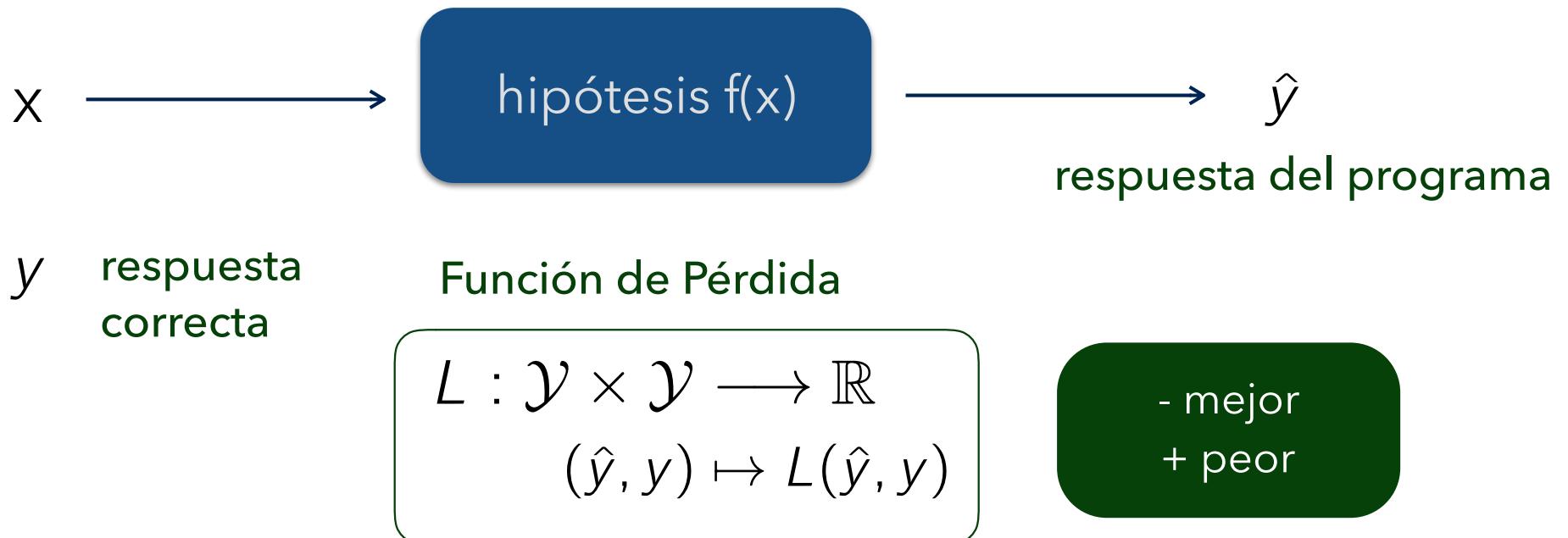
Medida de desempeño

$$R : \mathcal{H} \rightarrow \mathbb{R}$$

Machine Learning Basics

Función de Pérdida (**Loss function**)

Función **L** que permite medir la calidad de la hipótesis **f(x)** implementada por el programa sobre un input específico **x** y posiblemente la respuesta correcta/deseada/óptima para ese input.



Machine Learning Basics

- **Clasificación**



- **Misclassification Loss**

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

Machine Learning Basics

- **Regresión**



- **Squared Loss** $L(\hat{y}, y) = (\hat{y} - y)^2$
- **Epsilon Insensitive Loss**

$$L(\hat{y}, y) = (|\hat{y} - y| - \epsilon)_+ = \begin{cases} |\hat{y} - y| & \text{si } |\hat{y} - y| \geq \epsilon \\ 0 & \text{en otro caso} \end{cases}$$

Machine Learning Basics

- **Estimación de densidad de probabilidad**

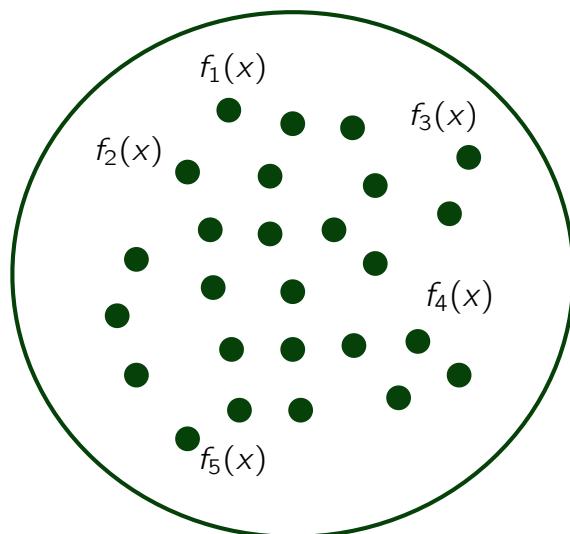


- **KL Divergence Loss** $L(\hat{y}) = -\ln \hat{y} = -\ln q(x)$

Machine Learning Basics

Medida de Desempeño (P)

Funcional **R** que permite medir cuantitativamente la calidad de la función **f(x)** implementada por el programa.



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

Medida de desempeño canónica
(Statistical Learning Theory)

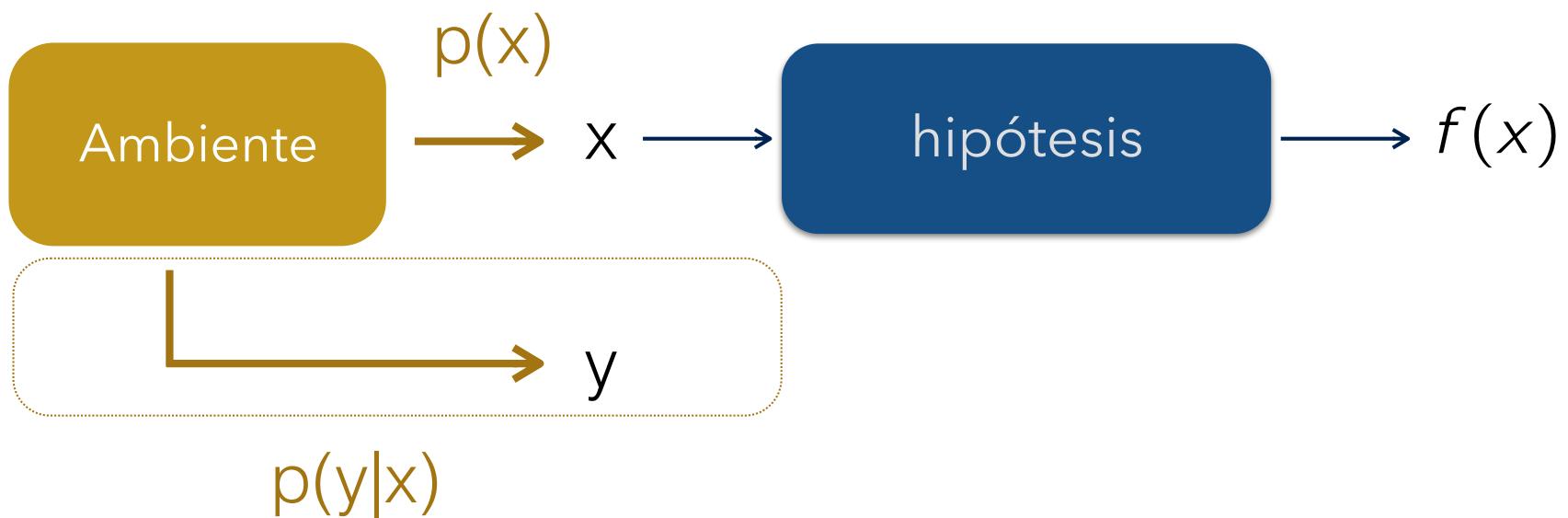
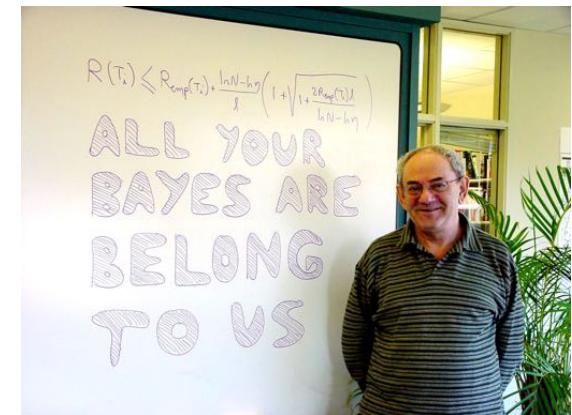
$$R(f) = E(L(f(x), y))$$

Riesgo

Machine Learning Basics

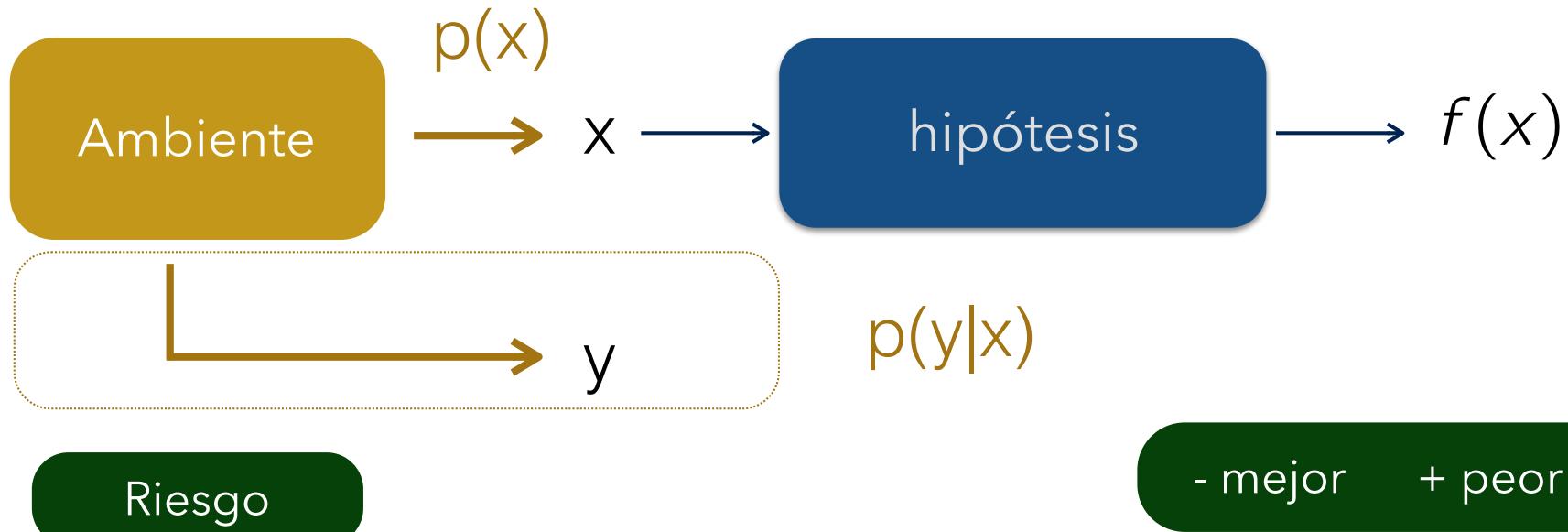
Medida de desempeño canónica
(Statistical Learning Theory)

$$R(f) = E(L(f(x), y))$$



Machine Learning Basics

Medida de desempeño canónica
(Statistical Learning Theory)



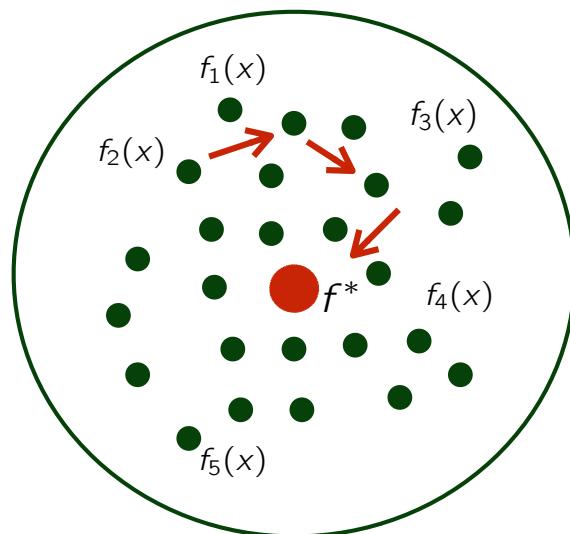
$$R(f) = E(L(f(x), y)) = \int L(f(x), y)p(x, y) dx dy$$

Machine Learning Basics

Objetivo del Aprendizaje
(Statistical Learning Theory)

Minimizar el Riesgo

$$\min R(f) = E(L(f(x), y)) \text{ s.t. } f \in \mathcal{H}$$



Learning as Search

Machine Learning Basics

Learning as Search

$$\min R(\mathbf{w}) = E(L(f(x, \mathbf{w}), y)) \text{ s.t. } \mathbf{w} \in \Lambda$$



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

Parámetros
del modelo

Espacio de
parámetros

Machine Learning Basics

Misclassification loss

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

Riesgo

$$\begin{aligned} R(f) &= E(L(f(x), y)) = \int I(f(x) \neq y) p(x, y) dx dy \\ &= P(f(x) \neq y) \end{aligned}$$

$$f^*(x) = \arg \max_k p(c_k | x)$$

elegir $f(x)$ para minimizar la probabilidad de error

Machine Learning Basics

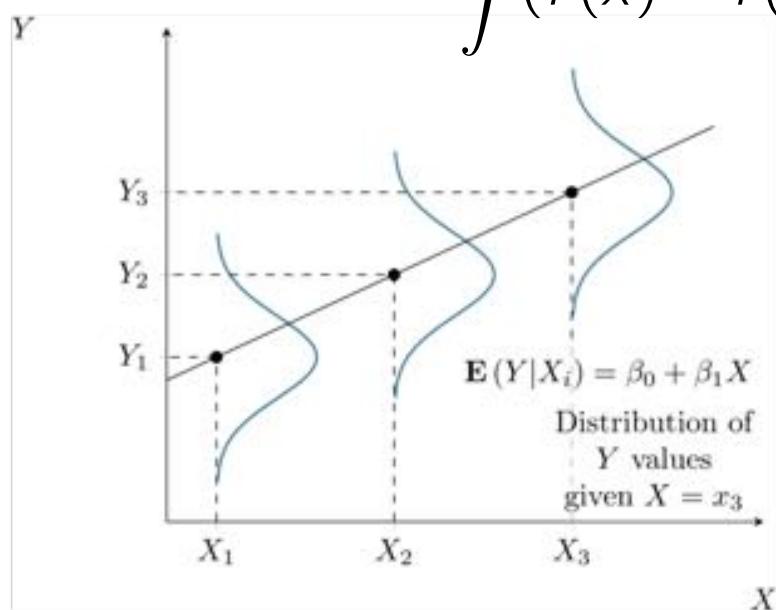
Squared loss

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

Riesgo

$$R(f) = \int (f(x) - y)^2 p(x, y) dx dy$$

$$= \int (f(x) - r(x))^2 p(x, y) dx dy + \int (r(x) - y)^2 p(x, y) dx dy$$



$$r(x) = E(y|x)$$

función de regresión

Machine Learning Basics

- Squared loss $L(\hat{y}, y) = (\hat{y} - y)^2$
- Risk

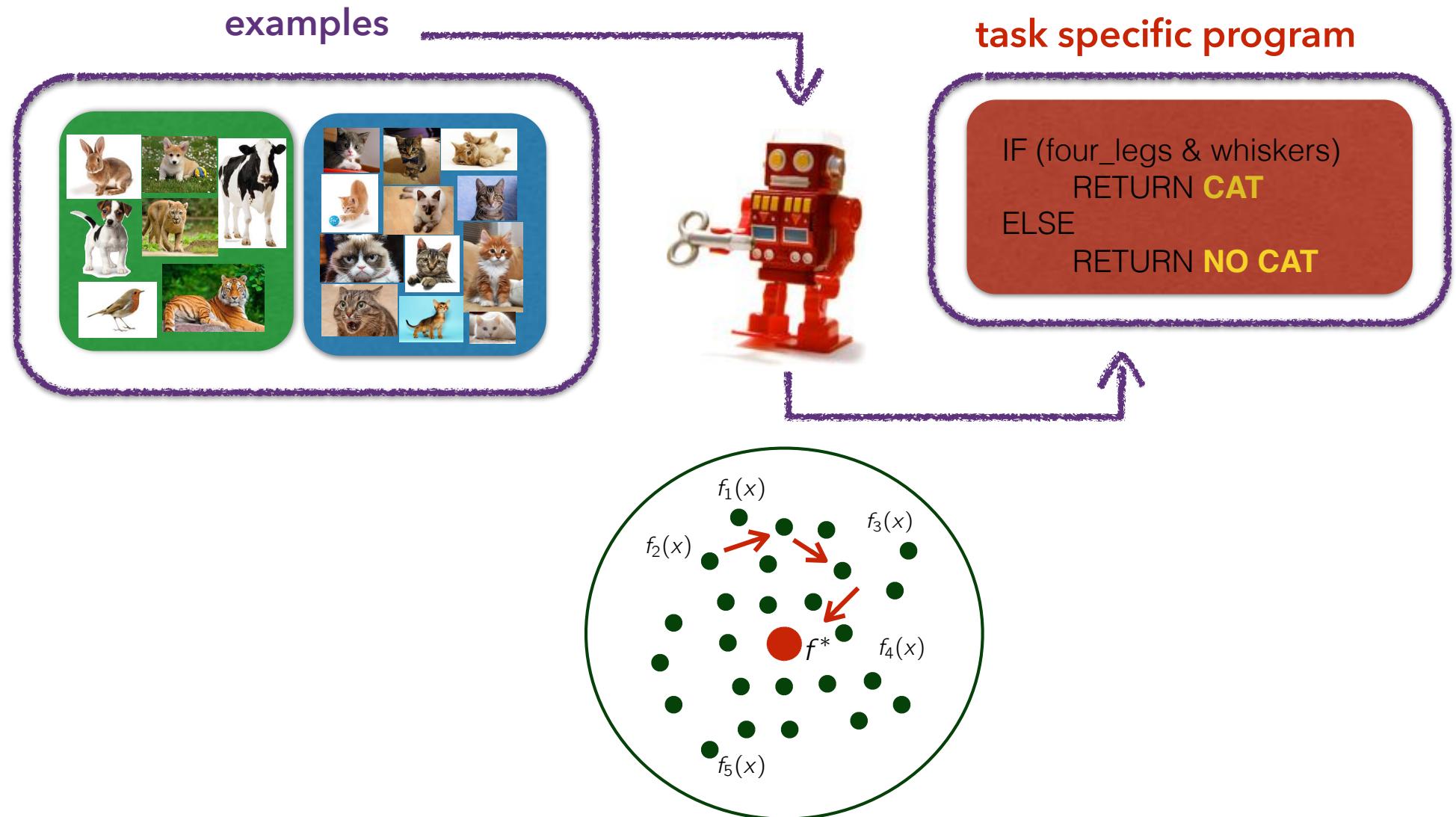
$$\begin{aligned} R(f) &= \int (f(x) - y)^2 p(x, y) dx dy \\ &= \int (f(x) - r(x))^2 p(x, y) dx dy + \boxed{\int (r(x) - y)^2 p(x, y) dx dy} \end{aligned}$$

independiente de $f(x)$

Elegir la función $f(x)$ más parecida posible a la función de regresión.
Si ésta está en el espacio de hipótesis elegirla!

$$f^*(x) = E(y|x)$$

Machine Learning Basics



Machine Learning Basics

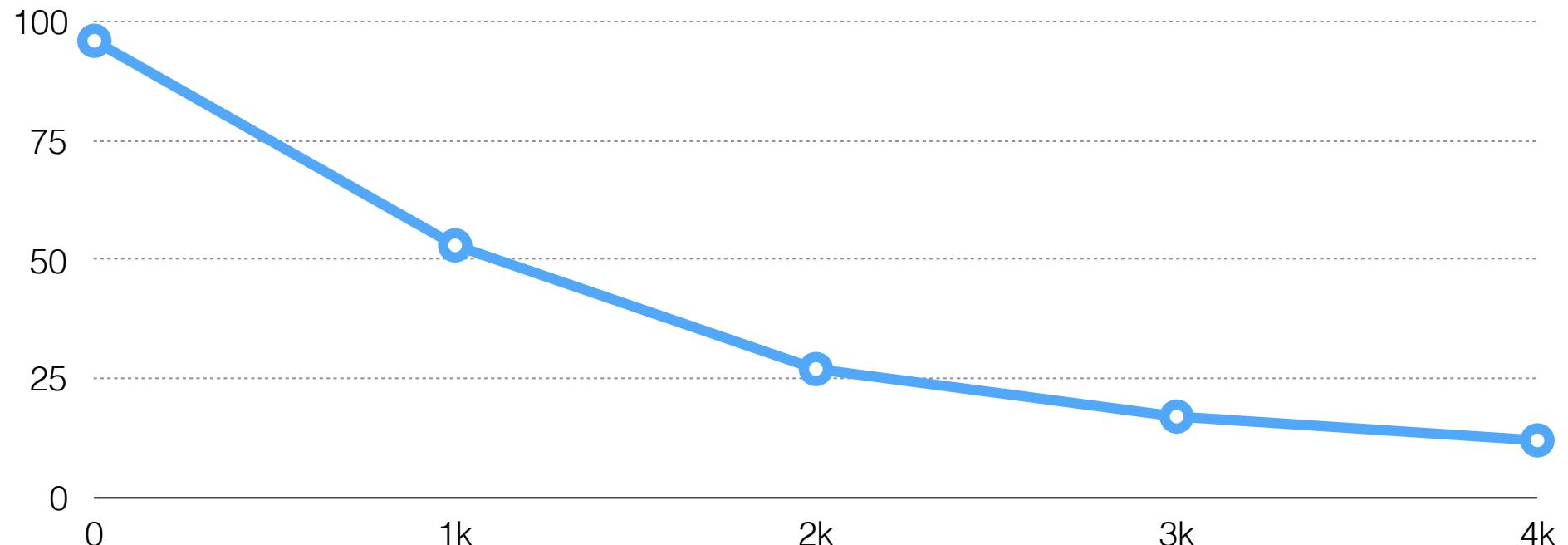


Prof. Tom Mitchell (1997)

“A program is said to learn from experience **E** with respect to some task **T** and some performance measure **P**, if its performance on **T**, as measured by **P**, improves with experience **E**”

Machine Learning Basics

Riesgo

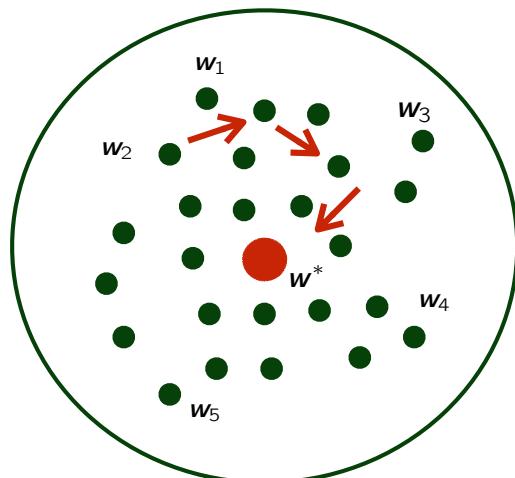


Experience

Machine Learning Basics

Objetivo del Aprendizaje = Minimizar el Riesgo

$$\min_{f \in \mathcal{H}} R(f) = \int L(f(x), y) p(x, y) dx dy$$



Big Problem

Evaluar el riesgo requiere conocer la distribución de probabilidad $p(x,y)$, i.e., saber exactamente cómo se generan los datos.

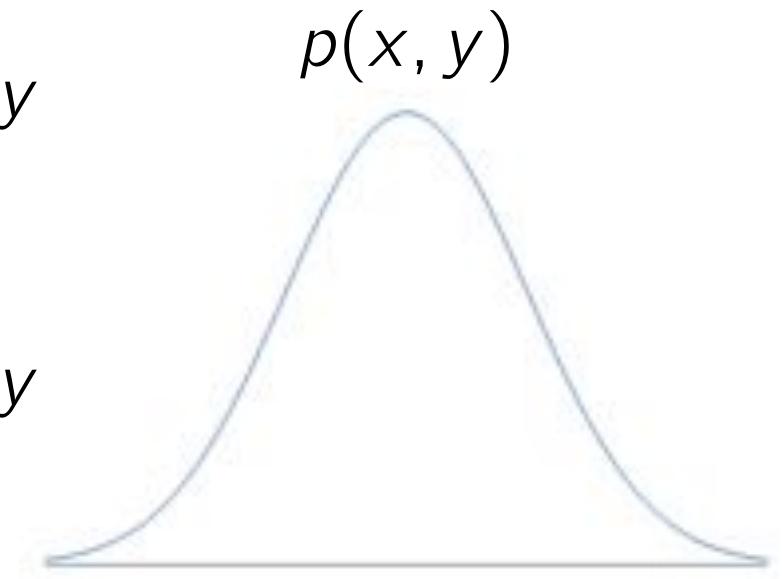
Machine Learning Basics

Principio de Inducción

Medida de desempeño **dependiente de los datos** que permite medir aproximar el riesgo **R**.

$$R(f) = \int L(f(x), y) p(x, y) dx dy$$

$$\hat{R}(f) = \int L(f(x), y) \hat{p}(x, y) dx dy$$



Machine Learning Basics

Error de Entrenamiento (Riesgo Empírico)

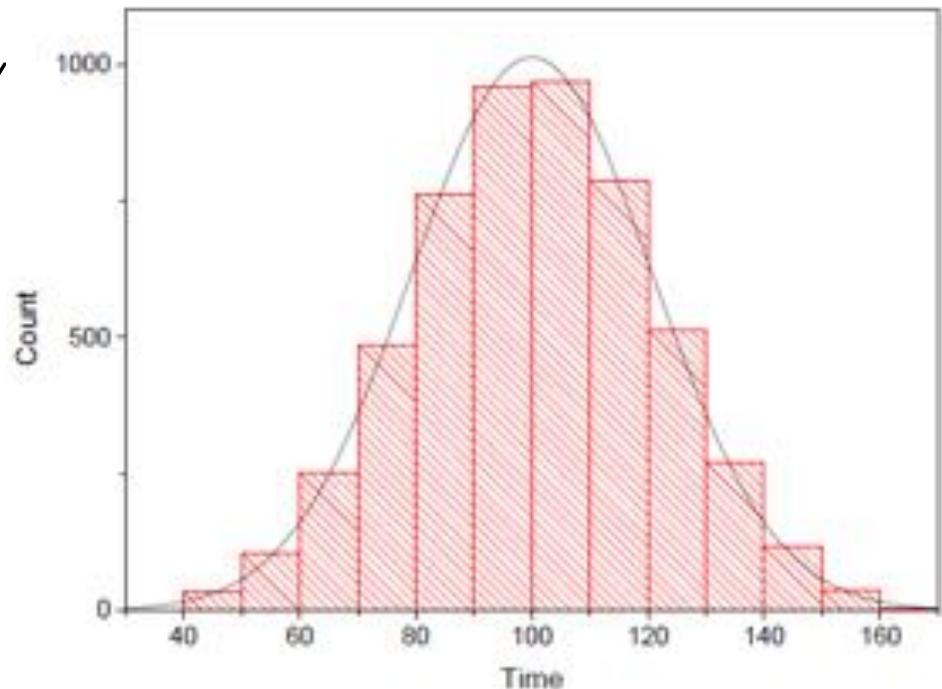
Criterio obtenido al reemplazar el riesgo teórico **R por su versión muestral o empírica.**

$$\hat{R}(f) = \int L(f(x), y) \hat{p}(x, y) dx dy$$

$$\hat{p}_{\text{emp}}(x, y) = \frac{1}{n} \sum_i \delta_i(x, y)$$

Error de Entrenamiento

$$\hat{R}_{\text{emp}}(f) = \sum_i L(f(x_i), y_i)$$



El riesgo empírico es por amplio margen la métrica de desempeño más utilizada en la práctica.

Generalización vs Overfitting

casos/problemas conocidos

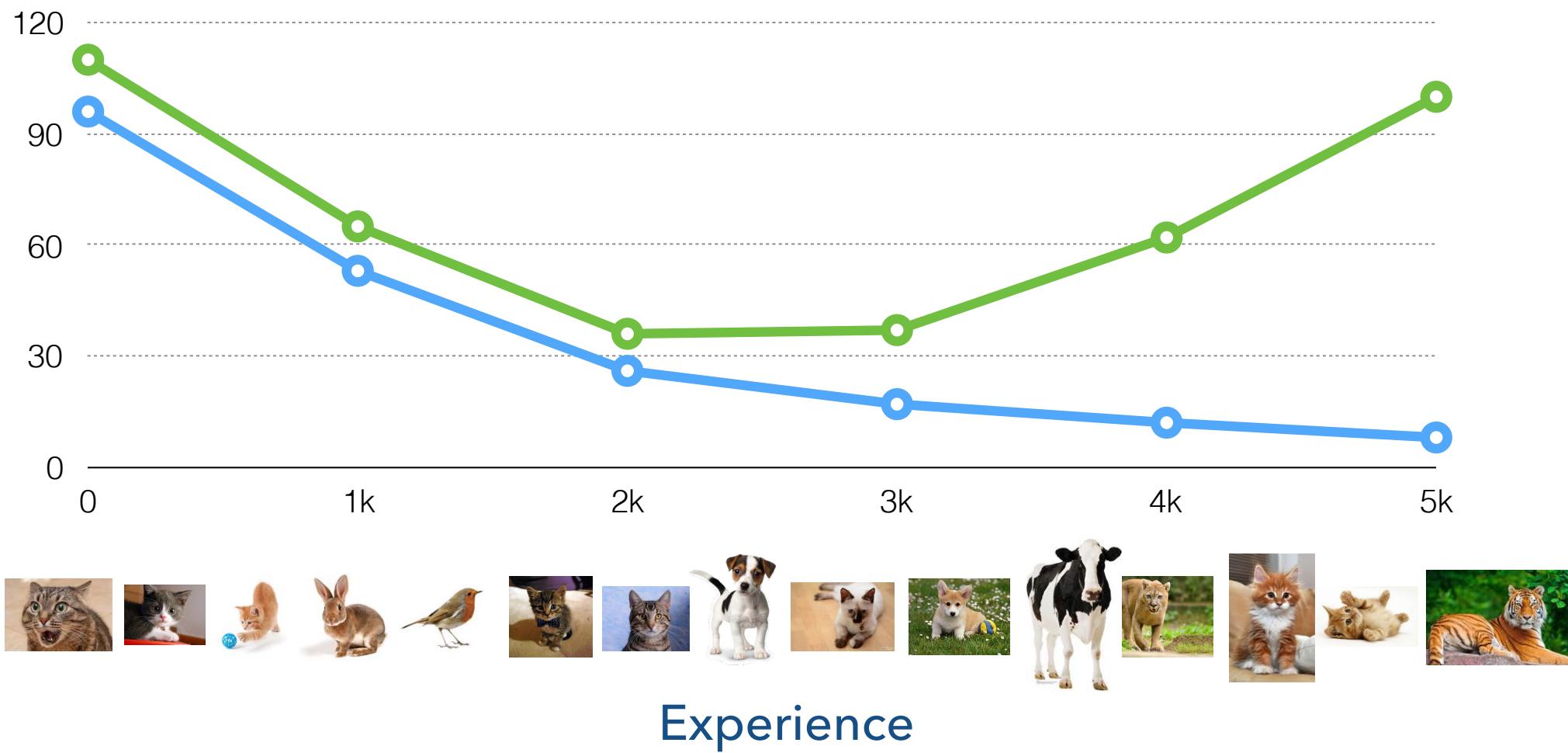


¿Funciona?

En general, NO.

Generalización vs Overfitting

Riesgo versus Riesgo Empírico



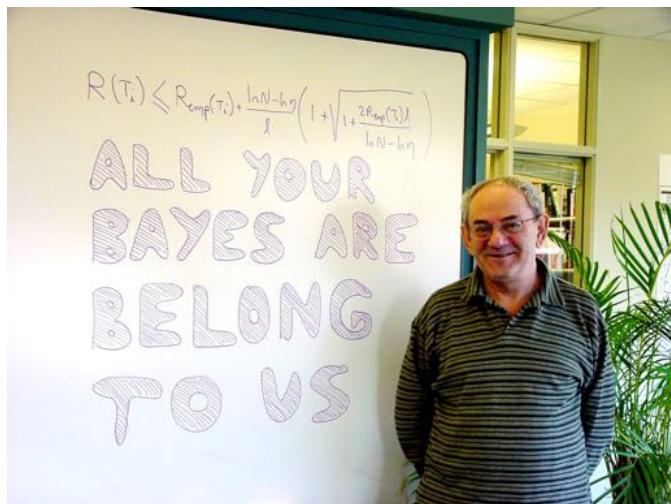
Generalización vs Overfitting

casos/problemas desconocidos



Generalización es la capacidad de una máquina de aprendizaje de mantener la performance que ha adquirido durante el entrenamiento frente a nuevos ejemplos, es decir, sobre casos no incluidos explícitamente en el conjunto de entrenamiento.

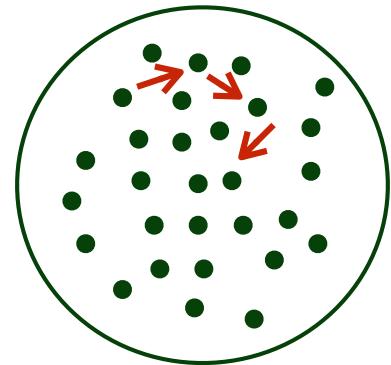
Generalización vs Overfitting



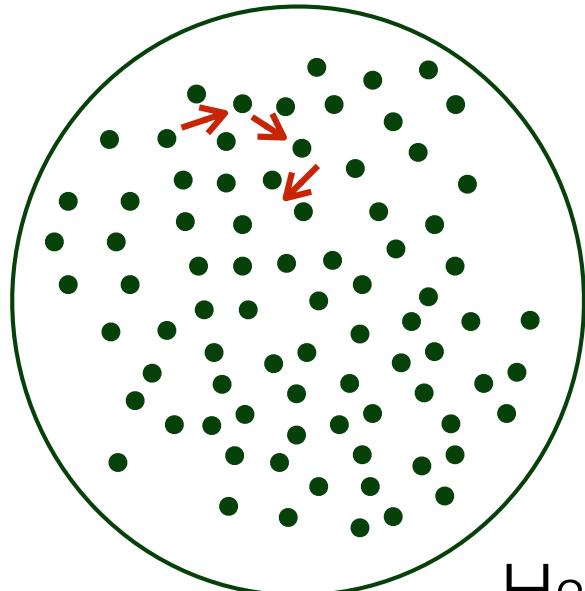
Prof. W. Vapnik (STL)

Minimizar el error de entrenamiento funciona si la complejidad del espacio de hipótesis es “pequeña” en comparación con el número de datos de entrenamiento disponibles.

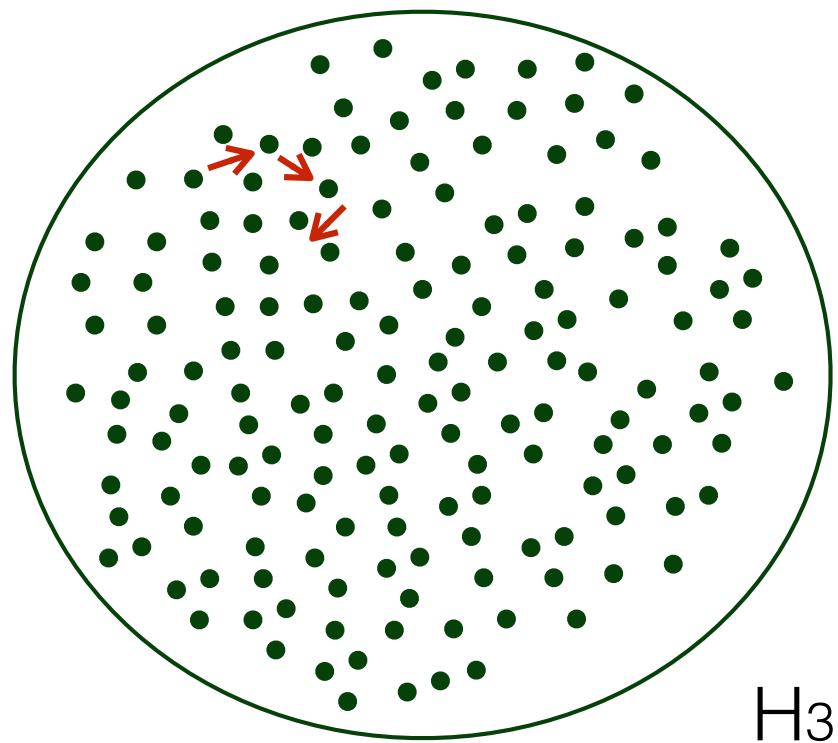
Generalización vs Overfitting



H_1

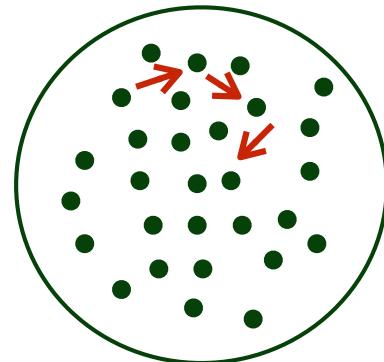


H_2



H_3

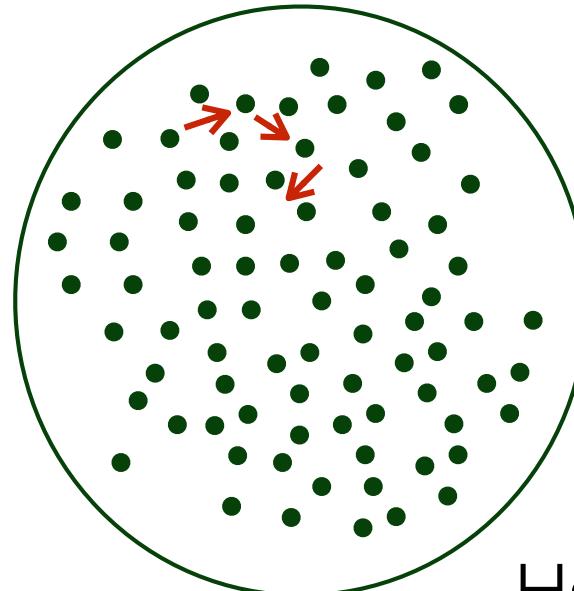
Generalización vs Overfitting



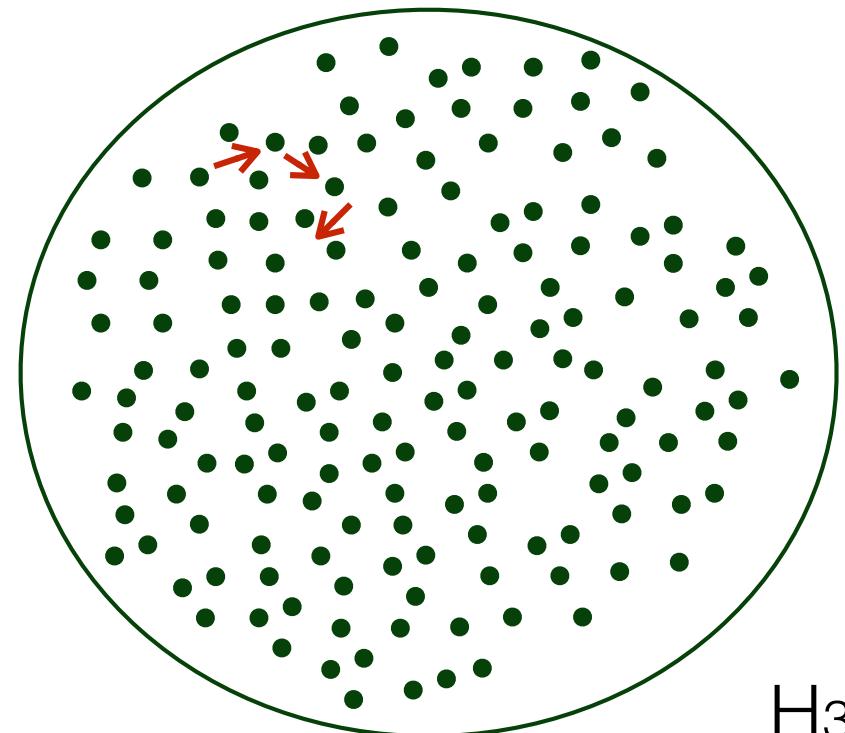
H_1



number of examples: m

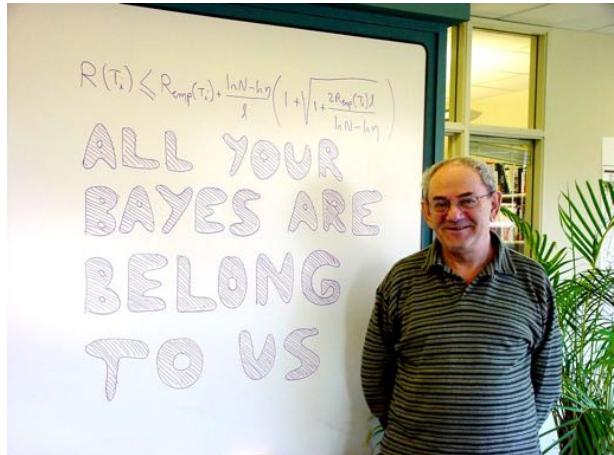


H_2



H_3

Generalización vs Overfitting



Prof. W. Vapnik (STL)

Sea **c** la complejidad del espacio de hipótesis (medida mediante la denominada dimensión VC). Entonces, con probabilidad $1 - \eta$

$$R(f) \leq \hat{R}_{\text{emp}}(f) + \sqrt{\frac{c \log(\frac{2n}{c} + 1) - \log(\frac{\eta}{4})}{n}}$$

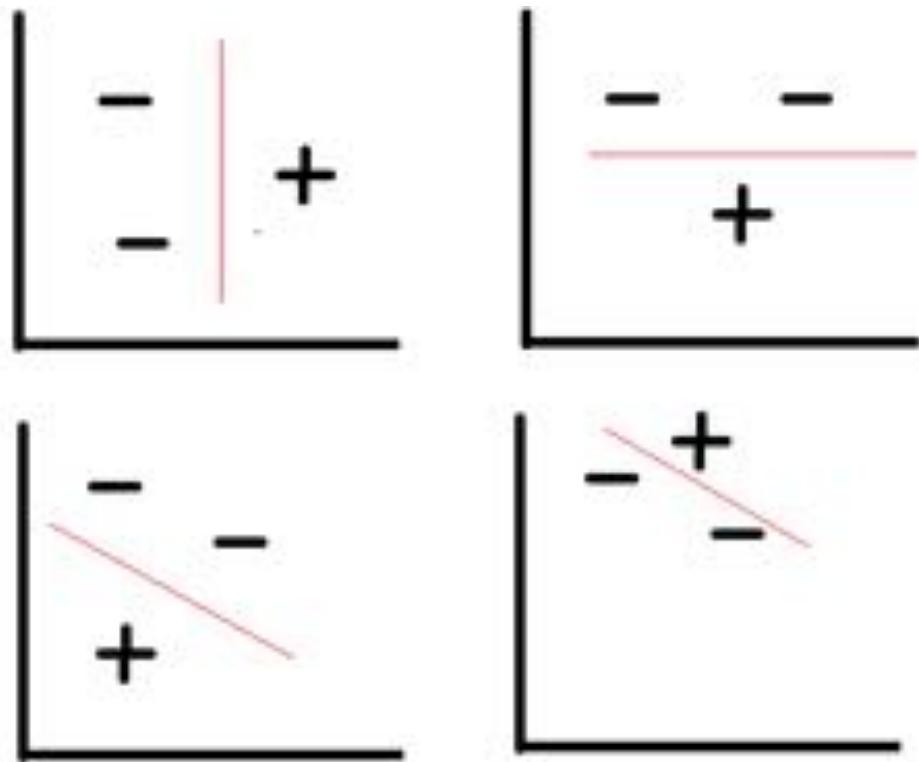
Error "de pruebas"
(riesgo)

Error de
entrenamiento

Complejidad

Generalización vs Overfitting

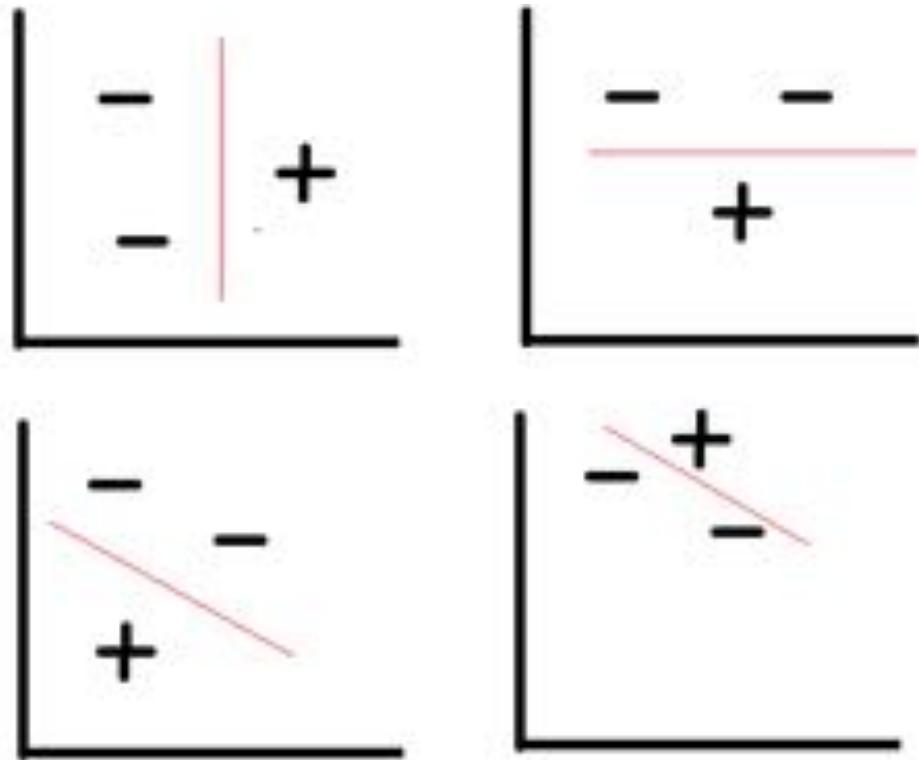
Shattering Dimension



Un espacio de hipótesis “rompe” (shatter) un conjunto de n puntos si para cualquier configuración posible de las etiquetas (2^n) existe una función del espacio que separa correctamente las 2 clases.

Generalización vs Overfitting

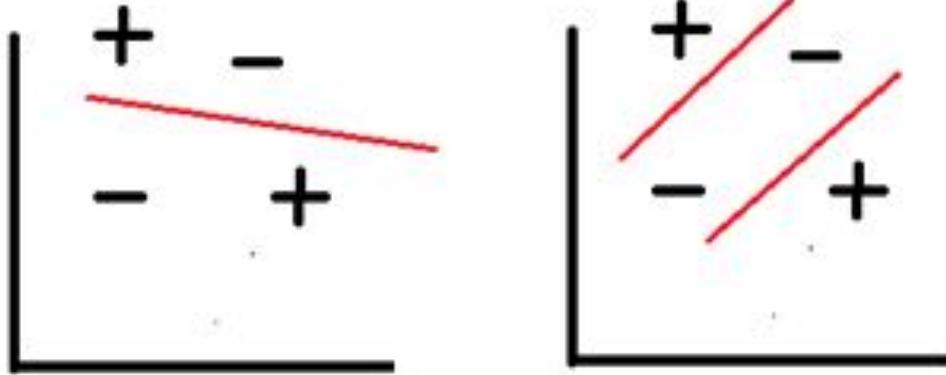
VC Dimension



Número máximo de puntos que pueden ser “rotos” (shutter) por el espacio de hipótesis.

Generalización vs Overfitting

VC Dimension

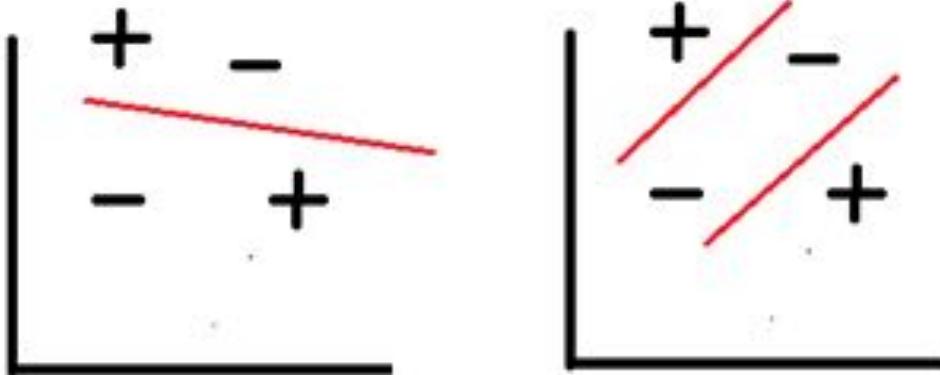


Número máximo de puntos que pueden ser “rotos” (shutter) por el espacio de hipótesis.

El espacio de las “líneas” (funciones lineales) en 2D no puede “romper” ningún conjunto de 4 puntos. Por lo tanto, la dimensiones VC de este espacio es 3.

Generalización vs Overfitting

VC Dimension



En general, el espacio de las "líneas" (funciones lineales) en d dimensiones tiene dimensión VC $d+1$

Generalización vs Overfitting

Sample Complexity $N(\epsilon, \eta)$

Número de ejemplos de entrenamiento requeridos de modo que

$$P(R(f) - R(f^*) > \epsilon) \leq \eta$$

Sample Complexity & VC dim

$$N(\epsilon, \eta) = \Omega\left(\frac{c + \log \frac{1}{\eta}}{\epsilon}\right)$$

En general, el número de ejemplos necesarios para obtener una “buena” hipótesis es lineal en la dimensión VC del espacio.