

Clase 2

Aprendizaje Supervisado

Marcelo Luis Errecalde^{1,2}

¹Universidad Nacional de San Luis, Argentina

²Universidad Nacional de la Patagonia Austral, Argentina
e-mails: merreca@unsl.edu.ar, merrecalde@gmail.com



Curso: Minería de Datos
Universidad Nacional de San Luis - Año 2018

Resumen

- 1 **Introducción**
- 2 **Aprendizaje automático (supervisado)**
- 3 **Aprendizaje de una hipótesis: esquema general**
- 4 **Evaluación de una hipótesis**

De acuerdo a la **retroalimentación**:

Repaso: Tipos de aprendizaje

De acuerdo a la **retroalimentación**:

- **Aprendizaje supervisado:** experiencia es un conjunto de ejemplos $\langle x, f(x) \rangle$, de la función f a ser aproximada.

Repaso: Tipos de aprendizaje

De acuerdo a la **retroalimentación**:

- **Aprendizaje supervisado**: experiencia es un conjunto de ejemplos $\langle x, f(x) \rangle$, de la función f a ser aproximada.
- **Aprendizaje por refuerzos**: experiencia son secuencias de tri-uplas $\langle s, a, r \rangle$, donde a es la acción tomada por el agente en el estado s , y r es la evaluación numérica recibida desde el ambiente por la realización de esta acción.

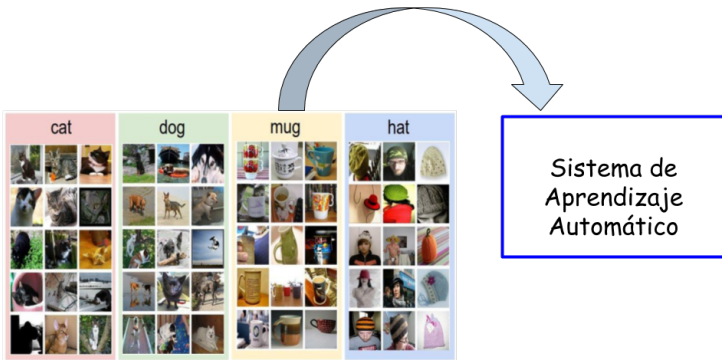
Repaso: Tipos de aprendizaje

De acuerdo a la **retroalimentación**:

- **Aprendizaje supervisado:** experiencia es un conjunto de ejemplos $\langle x, f(x) \rangle$, de la función f a ser aproximada.
- **Aprendizaje por refuerzos:** experiencia son secuencias de tri-uplas $\langle s, a, r \rangle$, donde a es la acción tomada por el agente en el estado s , y r es la evaluación numérica recibida desde el ambiente por la realización de esta acción.
- **Aprendizaje no supervisado:** no existe una retroalimentación explícita desde el ambiente.

Retroalimentación en Aprendizaje Automático: supervisado

Retroalimentación = Ejemplos Etiquetados

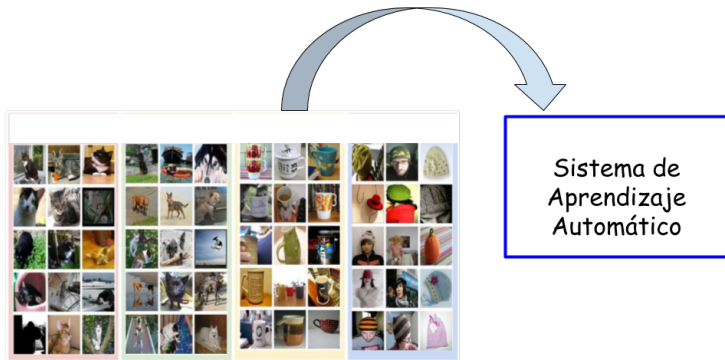


Tareas que se abordan con aprendizaje supervisado

- Clasificación (o discriminación) (múltiples clases, única etiqueta)
 - Aprendizaje de conceptos (clasificación binaria)
- Categorización (múltiples clases, múltiples etiquetas)
- Regresión (predicción numérica)

Retroalimentación en Aprendizaje Automático: no-supervisado

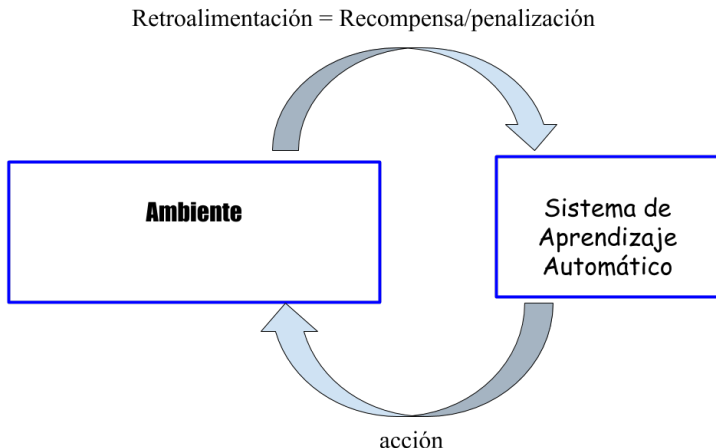
Retroalimentación = Ejemplos No Etiquetados



Tareas que se abordan con aprendizaje no-supervisado

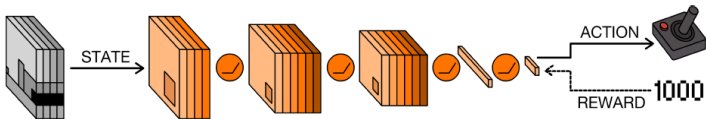
- Clustering
- Aprendizaje de reglas de asociación
- Técnicas no supervisadas de reducción de dimensionalidad mediante transformación del espacio de características

Retroalimentación en Aprendizaje Automático: refuerzo



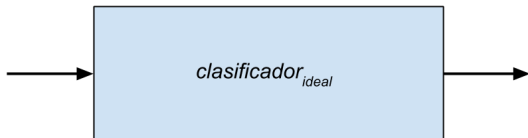
Ejemplo: aprendiendo los juegos de Atari

Deep Reinforcement Learning



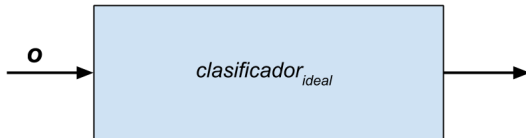
Aprendizaje automático (supervisado)

Idea intuitiva: intentar **reproducir** un proceso de clasificación **correcto/ideal** (*clasificador_{ideal}*),



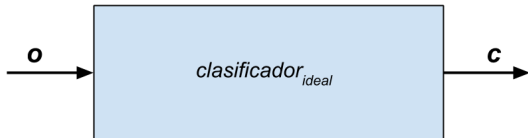
Aprendizaje automático (supervisado)

Idea intuitiva: ... que para cada **entrada** (el objeto que se quiere clasificar) ***o***



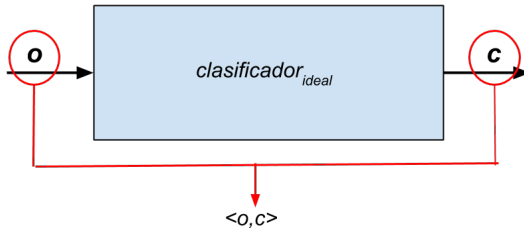
Aprendizaje automático (supervisado)

Idea intuitiva: ... que para cada **entrada** (el objeto que se quiere clasificar) ***o***, genera una salida ***c*** (la clase de ***o***)



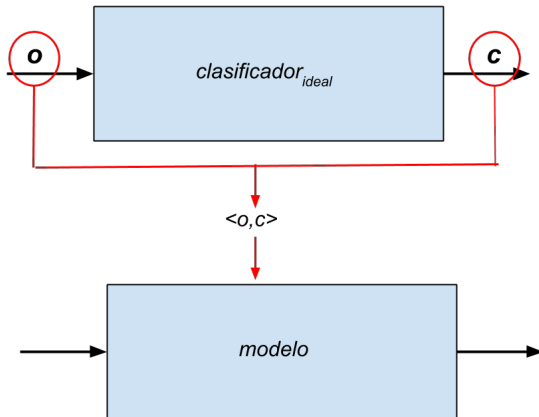
Aprendizaje automático (supervisado)

Idea intuitiva: ... usando ejemplos $\langle o, c \rangle$ del comportamiento de $\text{clasificador}_{ideal}$,



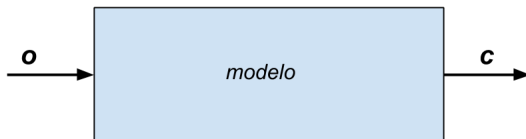
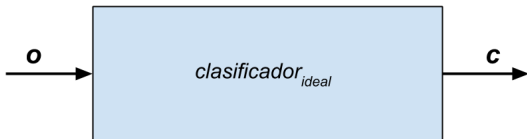
Aprendizaje automático (supervisado)

Idea intuitiva: ... usando **ejemplos** $\langle o, c \rangle$ del comportamiento de $\text{clasificador}_{ideal}$, para entrenar otro clasificador (**modelo**)



Aprendizaje automático (supervisado)

Idea intuitiva: ... cuyos comportamientos sean **tan parecidos** como sea posible.



Aprendizaje automático (supervisado)

Puntos **claves**:

- las salidas (clasificaciones) de *clasificador_{ideal}* y *modelo* deberían coincidir respecto a los ejemplos de entrenamiento pero (y más importante),

Aprendizaje automático (supervisado)

Puntos **claves**:

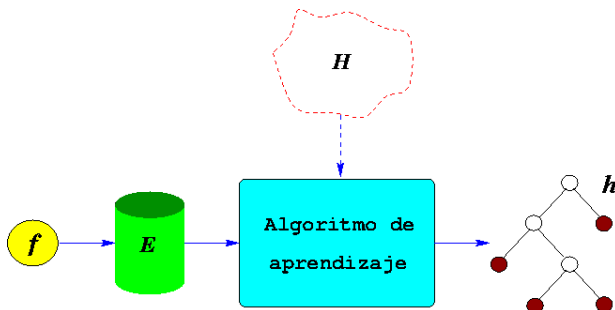
- las salidas (clasificaciones) de *clasificador_{ideal}* y *modelo* deberían coincidir respecto a los ejemplos de entrenamiento pero (y más importante),
- deberían coincidir sobre casos (objetos) no presentes en el conjunto de entrenamiento (**generalizar**)
- Este proceso, en matemática, se conoce como **aproximación de una función**

Aprendizaje de un clasificador (supervisado)

Idea: aproximar la función **ideal** de clasificación:

$$f : \mathcal{O} \mapsto \mathcal{C}$$

con un conjunto de entrenamiento E , de ejemplos $\langle o, c \rangle$, tal que $o \in \mathcal{O}$ es un **objeto**, y $c \in \mathcal{C}$ es la **categoría** que f asigna a o .



Aprendizaje de un clasificador (+ formal)

Dados

- Una **función de clasificación** o *clasificación objetivo desconocida*:

$$f : \mathcal{O} \rightarrow \mathcal{C}$$

Aprendizaje de un clasificador (+ formal)

Dados

- Una **función de clasificación** o *clasificación objetivo desconocida*:

$$f : \mathcal{O} \rightarrow \mathcal{C}$$

- Un **conjunto de entrenamiento** E , tal que cada ejemplo es una instancia rotulada con una de las posibles clases, $\langle o, f(o) \rangle$ donde $o \in \mathcal{O}$ y $f(o) \in \mathcal{C}$

Aprendizaje de un clasificador (+ formal)

Dados

- Una **función de clasificación** o *clasificación objetivo* desconocida:

$$f : \mathcal{O} \rightarrow \mathcal{C}$$

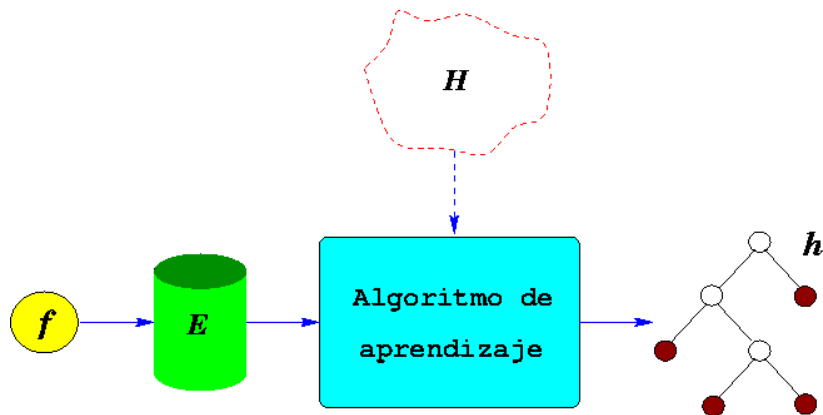
- Un **conjunto de entrenamiento** E , tal que cada ejemplo es una instancia rotulada con una de las posibles clases, $\langle o, f(o) \rangle$ donde $o \in \mathcal{O}$ y $f(o) \in \mathcal{C}$

Tarea: **estimar** f , es decir, encontrar una función:

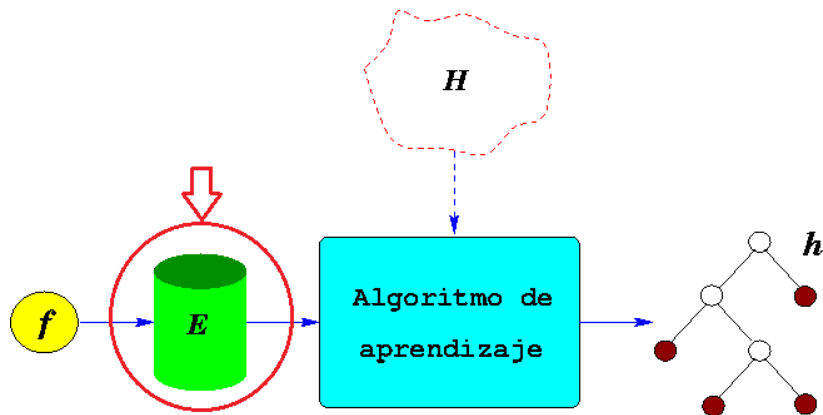
$$h : \mathcal{O} \mapsto \mathcal{C}$$

denominada *hipótesis clasificadora* o *clasificador*, tal que $h(o) = f(o)$ para todo $o \in \mathcal{O}$.

Aprendizaje de una hipótesis: esquema general



La entrada del proceso: el conjunto de entrenamiento



Ejemplo: Problema del tiempo

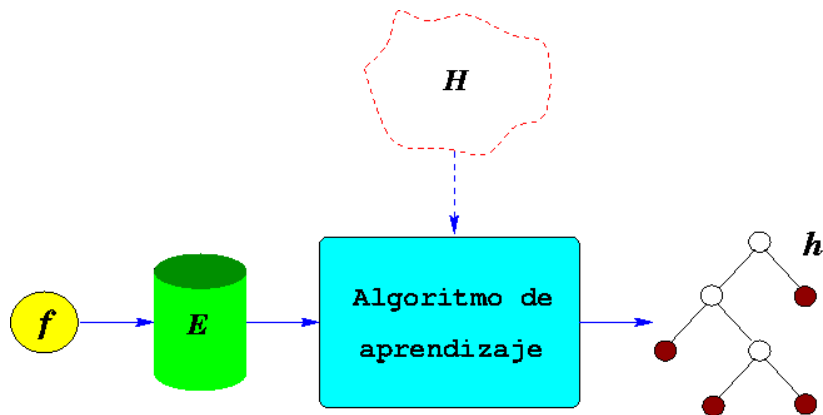
Conjunto de Entrenamiento

Ej.	Atributos				
	<i>Estado</i>	<i>Temp</i>	<i>Humedad</i>	<i>Viento</i>	<i>JT</i>
e_1	<i>soleado</i>	<i>caluroso</i>	<i>alta</i>	<i>falso</i>	<i>no</i>
e_2	<i>soleado</i>	<i>caluroso</i>	<i>alta</i>	<i>verdadero</i>	<i>no</i>
e_3	<i>nublado</i>	<i>caluroso</i>	<i>alta</i>	<i>falso</i>	<i>si</i>
e_4	<i>lluvioso</i>	<i>templado</i>	<i>alta</i>	<i>falso</i>	<i>si</i>
e_5	<i>lluvioso</i>	<i>fresco</i>	<i>normal</i>	<i>falso</i>	<i>si</i>
e_6	<i>lluvioso</i>	<i>fresco</i>	<i>normal</i>	<i>verdadero</i>	<i>no</i>
e_7	<i>nublado</i>	<i>fresco</i>	<i>normal</i>	<i>verdadero</i>	<i>si</i>
e_8	<i>soleado</i>	<i>templado</i>	<i>alta</i>	<i>falso</i>	<i>no</i>
e_9	<i>soleado</i>	<i>fresco</i>	<i>normal</i>	<i>falso</i>	<i>si</i>
e_{10}	<i>lluvioso</i>	<i>templado</i>	<i>normal</i>	<i>falso</i>	<i>si</i>
e_{11}	<i>soleado</i>	<i>templado</i>	<i>normal</i>	<i>verdadero</i>	<i>si</i>
e_{12}	<i>nublado</i>	<i>templado</i>	<i>alta</i>	<i>verdadero</i>	<i>si</i>
e_{13}	<i>nublado</i>	<i>caluroso</i>	<i>normal</i>	<i>falso</i>	<i>si</i>
e_{14}	<i>lluvioso</i>	<i>templado</i>	<i>alta</i>	<i>verdadero</i>	<i>no</i>

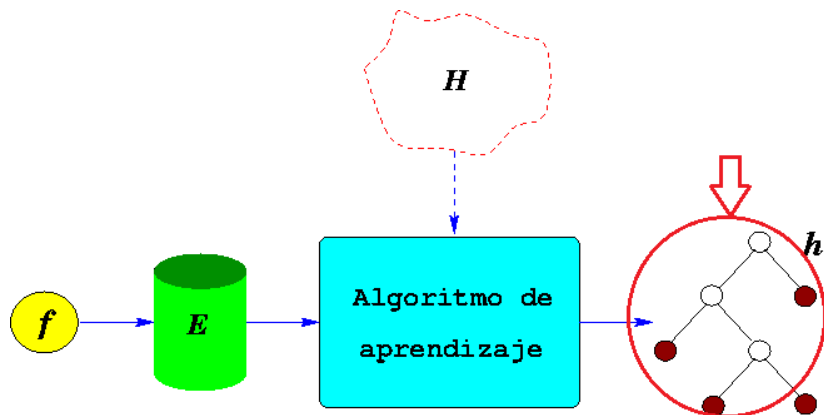
$$e_1 = \langle \bar{x}, f(\bar{x}) \rangle$$

$$e_1 = \langle \langle \text{soleado}, \text{caluroso}, \text{alta}, \text{falso} \rangle, \text{no} \rangle$$

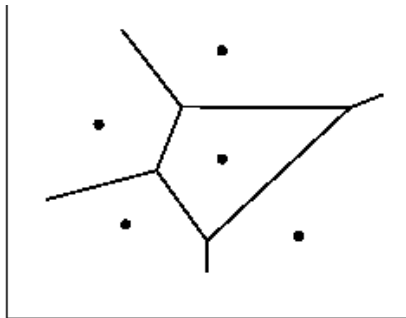
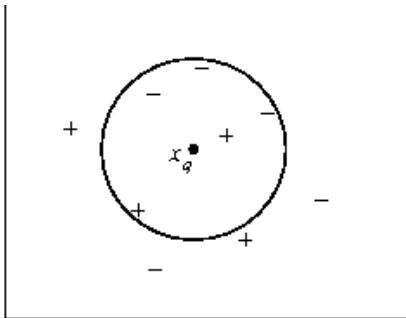
Aprendizaje de una hipótesis: esquema general



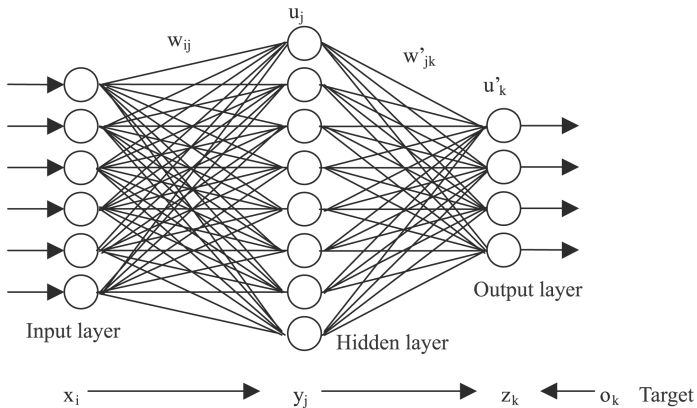
La salida del proceso: el clasificador



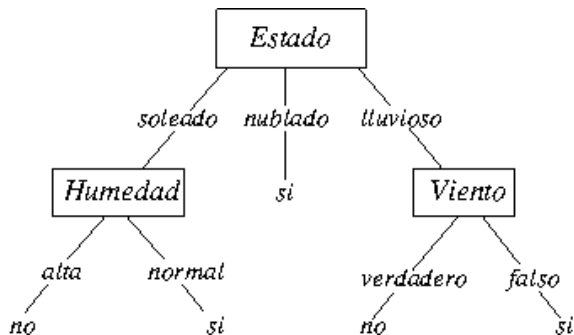
Un clasificador muy simple: k -NN



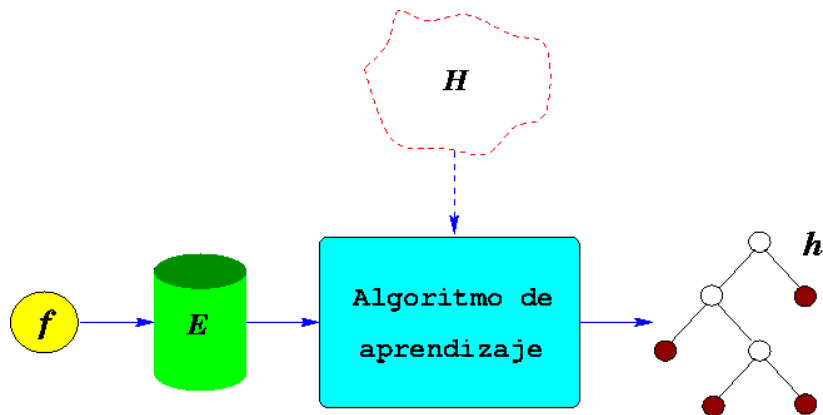
Otro clasificador muy usado: redes neuronales (NN)



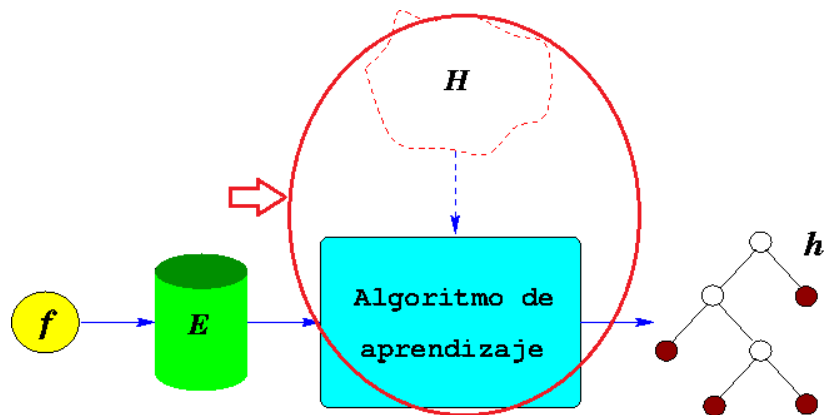
Un clasificador con conocimiento explícito: árboles de decisión



Aprendizaje de una hipótesis: esquema general



El proceso de aprendizaje: búsqueda en un espacio de hipótesis



Generalización como búsqueda

- Una de las decisiones fundamentales en el AA es el **lenguaje de representación** utilizado para describir el concepto a aprender.

Generalización como búsqueda

- Una de las decisiones fundamentales en el AA es el **lenguaje de representación** utilizado para describir el concepto a aprender.
- Las “sentencias” de este lenguaje determinan el **espacio de hipótesis** que el algoritmo de aprendizaje deberá recorrer para encontrar una que sea consistente con los datos de entrenamiento.

Generalización como búsqueda

- Una de las decisiones fundamentales en el AA es el **lenguaje de representación** utilizado para describir el concepto a aprender.
- Las “sentencias” de este lenguaje determinan el **espacio de hipótesis** que el algoritmo de aprendizaje deberá recorrer para encontrar una que sea consistente con los datos de entrenamiento.
- La representación de las hipótesis puede ser de lo más variada:

Generalización como búsqueda

- Una de las decisiones fundamentales en el AA es el **lenguaje de representación** utilizado para describir el concepto a aprender.
- Las “sentencias” de este lenguaje determinan el **espacio de hipótesis** que el algoritmo de aprendizaje deberá recorrer para encontrar una que sea consistente con los datos de entrenamiento.
- La representación de las hipótesis puede ser de lo más variada:
 - Una combinación lineal de n características

$$h(x) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n$$

Generalización como búsqueda

- Una de las decisiones fundamentales en el AA es el **lenguaje de representación** utilizado para describir el concepto a aprender.
- Las “sentencias” de este lenguaje determinan el **espacio de hipótesis** que el algoritmo de aprendizaje deberá recorrer para encontrar una que sea consistente con los datos de entrenamiento.
- La representación de las hipótesis puede ser de lo más variada:
 - Una combinación lineal de n características

$$h(x) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n$$

- Árboles de decisión.

Generalización como búsqueda

- Una de las decisiones fundamentales en el AA es el **lenguaje de representación** utilizado para describir el concepto a aprender.
- Las “sentencias” de este lenguaje determinan el **espacio de hipótesis** que el algoritmo de aprendizaje deberá recorrer para encontrar una que sea consistente con los datos de entrenamiento.
- La representación de las hipótesis puede ser de lo más variada:

- Una combinación lineal de n características

$$h(x) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n$$

- Árboles de decisión.
- Redes Neuronales.
- En cada caso, los algoritmos buscan en espacios de búsqueda completamente distintos.

Sesgos inductivos

- Para poder **generalizar** más allá de los ejemplos de entrenamiento, todo sistema de aprendizaje inductivo utiliza alguna forma de **sesgo (bias)**.

Sesgos inductivos

- Para poder **generalizar** más allá de los ejemplos de entrenamiento, todo sistema de aprendizaje inductivo utiliza alguna forma de **sesgo (bias)**.
- Tipos de sesgo:

Sesgos inductivos

- Para poder **generalizar** más allá de los ejemplos de entrenamiento, todo sistema de aprendizaje inductivo utiliza alguna forma de **sesgo (bias)**.
- Tipos de sesgo:
 - Sesgo de **lenguaje**: es el que se produce al elegir el lenguaje para expresar los conceptos.

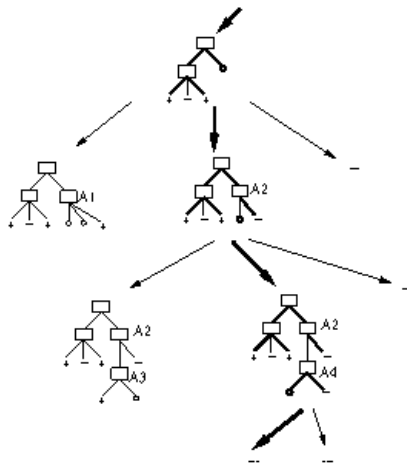
Sesgos inductivos

- Para poder **generalizar** más allá de los ejemplos de entrenamiento, todo sistema de aprendizaje inductivo utiliza alguna forma de **sesgo (bias)**.
- Tipos de sesgo:
 - Sesgo de **lenguaje**: es el que se produce al elegir el lenguaje para expresar los conceptos.
 - Sesgo de **búsqueda**: es el que determina **el orden** en el cual las hipótesis del espacio de búsqueda son consideradas.

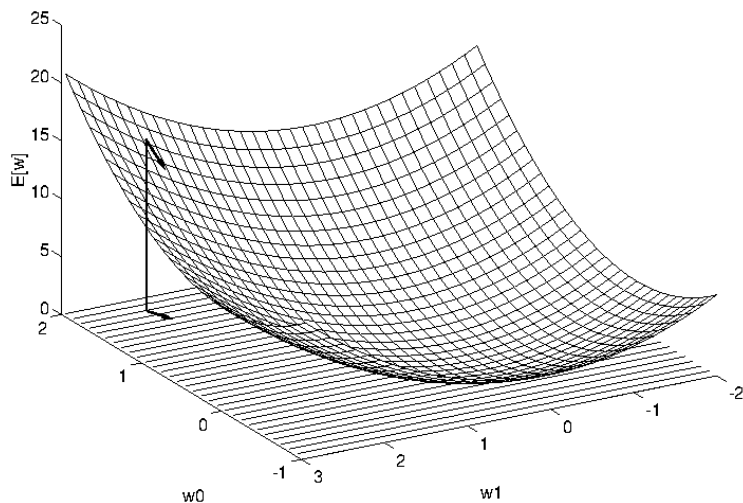
Sesgos inductivos

- Para poder **generalizar** más allá de los ejemplos de entrenamiento, todo sistema de aprendizaje inductivo utiliza alguna forma de **sesgo (bias)**.
- Tipos de sesgo:
 - Sesgo de **lenguaje**: es el que se produce al elegir el lenguaje para expresar los conceptos.
 - Sesgo de **búsqueda**: es el que determina **el orden** en el cual las hipótesis del espacio de búsqueda son consideradas.
 - Sesgo para **evitar el “sobre-ajuste”**: es el que busca descripciones de conceptos **más simples** (por ejemplo, la **poda** de árboles de decisión).

Búsqueda en el espacio de hipótesis en ID3



Búsqueda en el espacio de hipótesis en NN

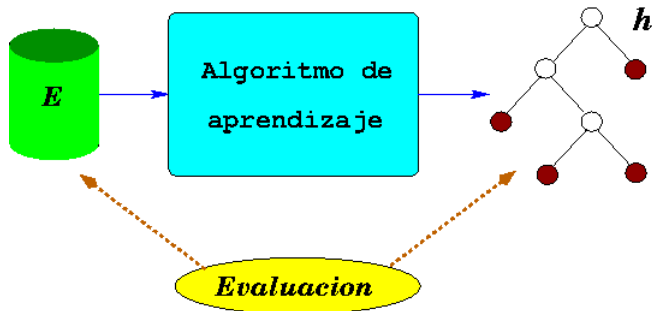


Evaluación de una hipótesis

Alternativas para la evaluación de una hipótesis (clasificador):

- El conjunto E se usa para entrenamiento y evaluación
- Separar la evidencia en un **conjunto de entrenamiento** y un **conjunto de test (prueba)**.
- Validación cruzada

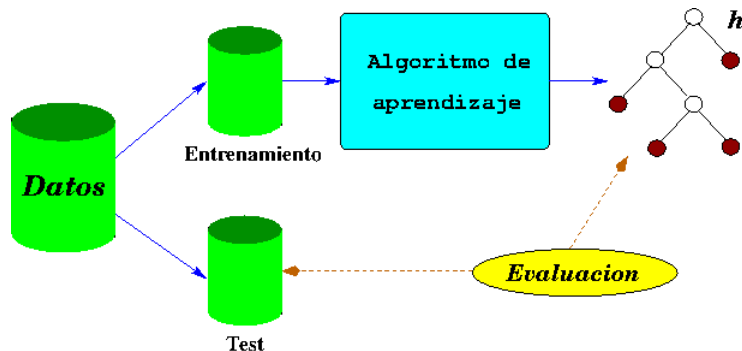
Entrenamiento y evaluación sobre el mismo conjunto



Problemas:

- sobreajuste (**overfitting**)
- subajuste (**underfitting**)

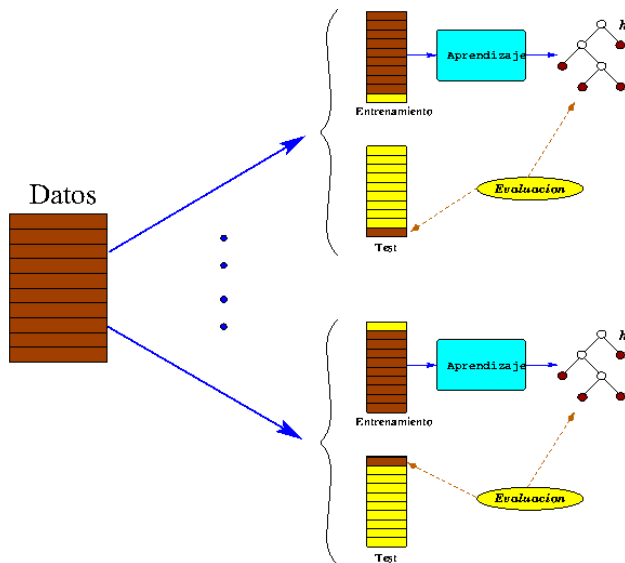
Entrenamiento y evaluación sobre conjuntos separados



Permite detectar el sobreajuste cuando la hipótesis arroja resultados mucho mejores para el conjunto de entrenamiento que el de test. **Problemas:**

- Resultados muy dependientes de la partición
- Escasez de datos

Evaluación mediante validación cruzada (*cross validation*)



Medidas de evaluación de clasificadores

Un método usual para medir las bondades de un clasificador, es considerar la **exactitud (accuracy)** del modelo, que mide esencialmente el **porcentaje de aciertos** de la hipótesis aprendida.

Esta medida se obtiene fácilmente a partir de la **matriz de confusión**.

Si se deben categorizar objetos en n clases, corresponderá una matriz de confusión M de $n \times n$.

Matriz de confusión

Cada componente $M_{i,j}$ es el número de casos en que la hipótesis h predijo el valor i y el valor real era j .

Ejemplo: Identificación de Autoría

<i>Estimado ($h(x)$)</i>	<i>Real ($f(x)$)</i>			
		Borges	Cortázar	Arlt
	Borges	71	3	1
	Cortázar	8	7	1
	Arlt	4	2	3

La exactitud se calcula dividiendo el número de casos en la diagonal (**aciertos**) por el número total de casos testeados:

$$acc_T(h) = \frac{71 + 7 + 3}{71 + 3 + 1 + 8 + 7 + 1 + 4 + 2 + 3} = \frac{81}{100} = 0,81$$

Otras medidas de evaluación

Precisión (precision) y alcance (recall)

<i>Estimado ($h(x)$)</i>	<i>Real ($f(x)$)</i>			
		Borges	Cortázar	Arlt
	Borges	71	3	1
	Cortázar	8	7	1
	Arlt	4	2	3

$$\pi_{Borges} = \frac{71}{71 + 3 + 1} = 0,947$$

$$\rho_{Borges} = \frac{71}{71 + 8 + 4} = 0,855$$

Combinando π y ρ

- Rara vez precision y recall son consideradas en forma aislada

Combinando π y ρ

- Rara vez precision y recall son consideradas en forma aislada
- Alternativas: medidas combinadas como la “ **F -measure**” (medida F):

$$F = \frac{2\pi\rho}{\pi + \rho}$$

Combinando π y ρ

- Rara vez precision y recall son consideradas en forma aislada
- Alternativas: medidas combinadas como la “ **F -measure**” (medida F):

$$F = \frac{2\pi\rho}{\pi + \rho}$$

- La medida previa es un caso particular (F_1) de la función **F_β** :

$$F_\beta = \frac{(\beta^2 + 1)\pi\rho}{\beta^2\pi + \rho}$$

para algún $0 \leq \beta \leq +\infty$

Combinando π y ρ

- Rara vez precision y recall son consideradas en forma aislada
- Alternativas: medidas combinadas como la “ **F -measure**” (medida F):

$$F = \frac{2\pi\rho}{\pi + \rho}$$

- La medida previa es un caso particular (F_1) de la función **F_β** :

$$F_\beta = \frac{(\beta^2 + 1)\pi\rho}{\beta^2\pi + \rho}$$

para algún $0 \leq \beta \leq +\infty$

- Usualmente $\beta = 1$ (igual peso a π y ρ)

Hiperparámetros y validación del modelo

- **Hiperparámetros**: parámetros relacionados a distintas **elecciones** realizadas al **instanciar** un modelo que deben ser especificados **antes** de entrenar el modelo con los datos.

Hiperparámetros y validación del modelo

- **Hiperparámetros**: parámetros relacionados a distintas **elecciones** realizadas al **instanciar** un modelo que deben ser especificados **antes** de entrenar el modelo con los datos.
- **Ejemplos**: número de vecinos y medida de distancia en **k-NN**, número de capas ocultas en **redes neuronales**, etc.

Hiperparámetros y validación del modelo

- **Hiperparámetros**: parámetros relacionados a distintas **elecciones** realizadas al **instanciar** un modelo que deben ser especificados **antes** de entrenar el modelo con los datos.
- **Ejemplos**: número de vecinos y medida de distancia en **k-NN**, número de capas ocultas en **redes neuronales**, etc.
- Para saber si el **modelo elegido** y sus **hiperparámetros** son adecuados, necesitamos una manera de **validarlos**

Hiperparámetros y validación del modelo

- **Hiperparámetros**: parámetros relacionados a distintas **elecciones** realizadas al **instanciar** un modelo que deben ser especificados **antes** de entrenar el modelo con los datos.
- **Ejemplos**: número de vecinos y medida de distancia en **k-NN**, número de capas ocultas en **redes neuronales**, etc.
- Para saber si el **modelo elegido** y sus **hiperparámetros** son adecuados, necesitamos una manera de **validarlos**
- Previamente vimos tres enfoques generales para validar un modelo, ahora describiremos algunos aspectos adicionales

Pensando sobre validación de modelos ...

- 1 Validarlo sobre **el mismo** conjunto de entrenamiento (la forma **errónea**)

Pensando sobre validación de modelos ...

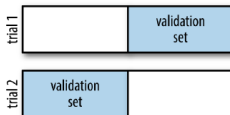
- 1 Validarlo sobre **el mismo** conjunto de entrenamiento (la forma **errónea**)
- 2 Validarlo sobre un **conjunto separado** (*holdout set*), vamos mejorando

Pensando sobre validación de modelos ...

- 1 Validarlo sobre **el mismo** conjunto de entrenamiento (la forma **errónea**)
- 2 Validarlo sobre un **conjunto separado** (*holdout set*), vamos mejorando
- 3 Validarlo mediante **validación cruzada** (*cross-validation*), el método normalmente usado.

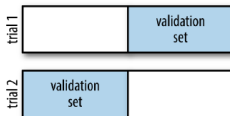
Validación cruzada en k -folds

$k = 2$

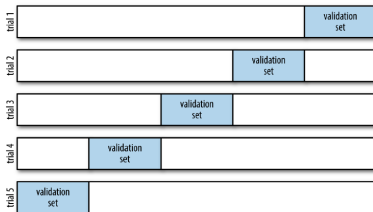


Validación cruzada en k -folds

$k = 2$

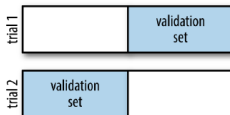


$k = 5$

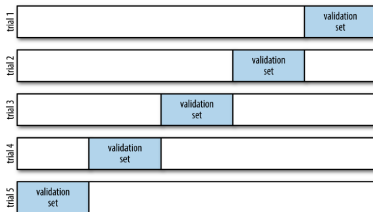


Validación cruzada en k -folds

$k = 2$



$k = 5$



k “extremo” = nro. de puntos de datos (leave-one-out)

Seleccionando el mejor modelo

Pregunta crucial: *¿si nuestro estimador (modelo) no está funcionando tan bien, como lo podría mejorar?*

Seleccionando el mejor modelo

Pregunta crucial: *¿si nuestro estimador (modelo) no está funcionando tan bien, como lo podría mejorar?*

- 1 Usar un modelo **más complicado/flexible**.

Seleccionando el mejor modelo

Pregunta crucial: *¿si nuestro estimador (modelo) no está funcionando tan bien, como lo podría mejorar?*

- 1 Usar un modelo **más complicado/flexible**.
- 2 Usar un modelo **menos complicado/flexible**.

Seleccionando el mejor modelo

Pregunta crucial: *¿si nuestro estimador (modelo) no está funcionando tan bien, como lo podría mejorar?*

- 1 Usar un modelo **más complicado/flexible**.
- 2 Usar un modelo **menos complicado/flexible**.
- 3 Conseguir **más ejemplos de entrenamiento**.

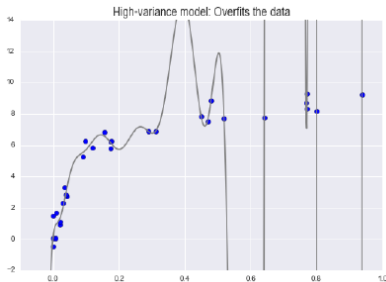
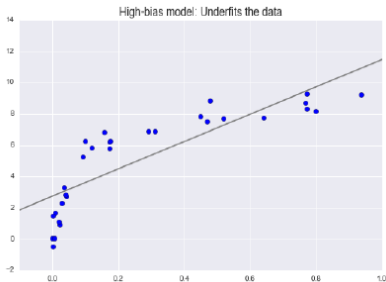
Seleccionando el mejor modelo

Pregunta crucial: *¿si nuestro estimador (modelo) no está funcionando tan bien, como lo podría mejorar?*

- 1 Usar un modelo **más complicado/flexible**.
- 2 Usar un modelo **menos complicado/flexible**.
- 3 Conseguir **más ejemplos de entrenamiento**.
- 4 Conseguir **más datos** para **agregar features** a cada ejemplo.

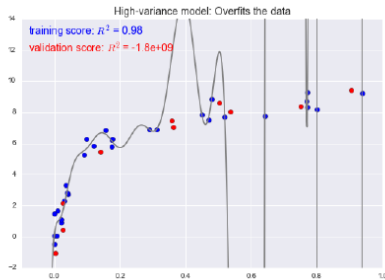
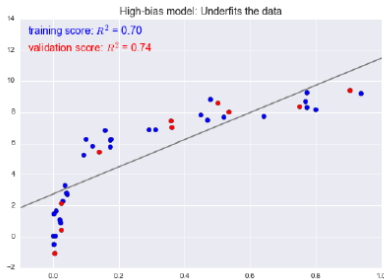
Complejidad del modelo y el compromiso “sesgo-varianza”

Alto sesgo y alta varianza en modelos de regresión



Complejidad del modelo y el compromiso “sesgo-varianza”

Desempeño en entrenamiento y validación

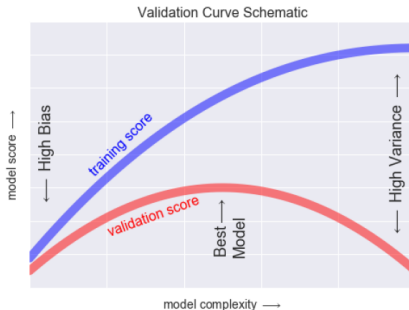


Curva de validación

- Desempeño en entrenamiento es **más alto** que en validación.
- Modelos de **baja complejidad subajustan** puntos de entrenamiento.
- Modelos de **alta complejidad** los **sobreajustan**.
- En un **valor intermedio**, la curva tiene un **máximo**.

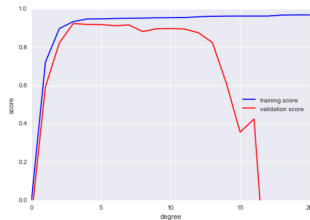
Curva de validación

- Desempeño en entrenamiento es **más alto** que en validación.
- Modelos de **baja complejidad subajustan** puntos de entrenamiento.
- Modelos de **alta complejidad los sobreajustan**.
- En un **valor intermedio**, la curva tiene un **máximo**.



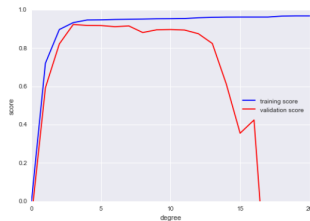
Ejemplo: encontrando el grado de un regresor polinomial

Curvas de validación para los datos del ejemplo previo

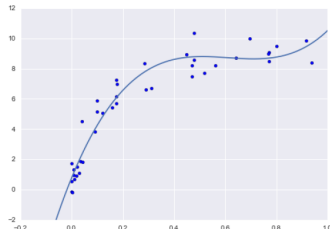


Ejemplo: encontrando el grado de un regresor polinomial

Curvas de validación para los datos del ejemplo previo



Modelo **óptimo** (grado 3) para esos datos



Introducción
ooooooo

Aprendizaje automático (supervisado)
ooooooooo

Aprendizaje de una hipótesis: esquema general
ooooooooooooooooo

Evaluación de una hipótesis
oooooooooooooooooooo●o

A continuación

A continuación

En lo que resta de la clase, veremos:

- Cómo llevar a cabo estas tareas en scikit-learn

A continuación

En lo que resta de la clase, veremos:

- Cómo llevar a cabo estas tareas en scikit-learn
- Cómo realizarlas en Weka