

## Tema-6.pdf



Blublu\_\_3



Inteligencia Artificial



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación  
Universidad de Granada

QUIERES  
CONSEGUIR  
15€??

→ TRÁENOS A TU  
CRUSH DE APUNTES ♡  
ANTES DE QUE  
LOS QUEME 🔥



WUOLAH

# TEMA 6

## 6.1 Introducción

El aprendizaje es una capacidad fundamental de la inteligencia humana, que nos permite:

- Adaptarnos a cambios de nuestro entorno
- Desarrollar una gran variedad de habilidades.
- Adquirir experiencia en nuevos dominios.

Hay que hacer aprender a una máquina para hacer que un programa sea capaz de actuar o tomar decisiones en entornos desconocidos o ante entradas que desconocemos y que además pueda actuar ante situaciones no previstas.

En realidad podemos desconocer muchos aspectos que nos hagan falta para actuar en esas circunstancias pero en general nos vamos a encontrar con situaciones en las que por un lado podemos necesitar una habilidad y tenemos que aprender a desarrollarla o podemos necesitar un conocimiento y debemos adquirirlo.

La idea fundamental de todo esto es que vamos a adquirir las habilidades o conocimientos de forma automática sin la intervención de algún humano.

Vamos a usar aprendizaje automático donde tengamos tareas difíciles de programar, en aplicaciones auto adaptables y en la minería de datos.

Un programa de ordenador se dice que aprende de la experiencia  $E$  con respecto a alguna clase de tareas  $T$  y a alguna medida de comportamiento  $P$ , si su comportamiento en tareas de  $T$ , medido a través de  $P$ , mejora con la experiencia  $E$ .

Hay varias estrategias de aprendizaje algunas de ellas son pero la más usada en aprendizaje automático es el aprendizaje a partir de ejemplos cuya idea es que a partir de un número elevado de ejemplos se aprende.

## 6.2 Tipos de aprendizaje

### 6.2.1 Aprendizaje inductivo

La idea es proporcionar un conjunto de ejemplos y con los ejemplos obtener un modelo que explica esos ejemplos u observaciones. Es decir, el modelo me tiene que decir qué está ocurriendo e incluso predecir qué es lo que va a ocurrir.

Los ejemplos son pares de valores  $(x, f(x))$  donde  $x$  puede ser un dato numérico o un vector y  $f(x)$  lo que se pretende reconocer. Lo que quiero es predecir si ante una nueva entrada desconocida se cumple  $f(x)$  (por ejemplo si un paciente tiene una enfermedad), es decir, quiero aprender a partir de los ejemplos un modelo que me explique un comportamiento general y a esto se le llama inducción. En estos problemas no podemos deducir porque no tenemos conocimiento de partida, partimos de ejemplos pero no de conocimiento. Nuestro modelo a buscar será  $f(x)$  generando varias hipótesis de qué puede ser  $f(x)$  y validando estas hipótesis con los ejemplos que me proporcionan (ejemplos de entrenamiento). Por tanto, aprender va a ser buscar en un espacio de hipótesis la hipótesis que mejor se acerca a mi función  $f$  teniendo en cuenta los ejemplos de entrenamiento.

Hemos visto que en el aprendizaje inductivo los ejemplos son pares de entrada/salida. A este tipo de aprendizaje se le denomina aprendizaje supervisado porque necesitamos un supervisor externo que proporciona información para asociar a cada entrada una salida.

Hay otros dos tipos de aprendizaje dependiendo de la retroalimentación que nos proporcione el entorno:

- Aprendizaje no supervisado: no hay retroalimentación del entorno, es decir, no hay valores de salida, solo de entrada. Y tiene como objetivo descubrir similitudes entre los datos proporcionados como entrada.
- Aprendizaje por refuerzo: se aprende a actuar a partir del refuerzo que proporciona el entorno en forma de recompensas o castigos obtenidos cuando se ejecutan acciones en situaciones determinadas.

### 6.2.2 Aprendizaje Supervisado

Hay dos grandes clases de métodos para llevarlo a cabo:

- Métodos basados en modelos: (es el más común) el resultado del aprendizaje es un conocimiento representado en algún lenguaje.
- Métodos basados en instancias: no se crea un modelo sino que se utiliza la propia representación utilizada para representar los ejemplos para poder dar respuesta a nuevas entradas. El caso más claro de este método es usar un diccionario que asocia entradas con salidas y que se va actualizando conforme se encuentran nuevos ejemplos.

## 6.3 Conceptos esenciales en aprendizaje inductivo.

Vamos a asumir que la función es de variable numérica y que los ejemplos son un conjunto de pares numéricos  $(x, f(x))$ .

-Generalización: Una hipótesis estará bien generalizada si puede predecir ejemplos que no se conocen. Es decir si la función que hemos encontrado en el proceso de aprendizaje devuelve valores esperados o adecuados cuando le paso ejemplos desconocidos.

-Consistencia: una hipótesis es consistente si se ajusta completamente a los datos de partida. Esto no siempre puede ser deseable que una hipótesis se ajuste a todos los datos de partida o entrenamiento. Para elegir entre múltiples hipótesis consistentes vamos a acudir a la Navaja de Ockham que establece que ante dos explicaciones de un mismo fenómeno la más simple prevalece.

-Simplicidad: La definición de simplicidad va a depender del problema en el que nos encontremos.

En un proceso de aprendizaje hay que encontrar un equilibrio entre hipótesis que se ajustan a los datos de entrenamiento pero que son más complejas o hipótesis que son más sencillas que no se ajustan a todos los ejemplos pero que son más simples según la Navaja de Ockham.

Además de estos conceptos, otro aspecto del proceso de aprendizaje está relacionado con el esquema que vamos a usar para representar las hipótesis de nuestro espacio de búsqueda. Hay varios modelos para representar hipótesis como árboles de decisión, reglas etc.. los nombrados son los más utilizados. En todo caso la elección del modelo de representación que yo use no es inmediato ni simple y esta elección va a determinar si mi problema de aprendizaje es realizable o no.

Se dice que un problema de aprendizaje es realizable si el espacio de hipótesis contiene la función verdadera. Pero si mi espacio de hipótesis está representado por árboles de decisión mi problema será

QUIERES  
CONSEGUIR  
15€??



TRÁENOS A TU  
CRUSH DE APUNTES  
ANTES DE QUE  
LOS QUEME



si consigues que suba apuntes, te llevas 15€ + 5  
Wuolah Coins para los sorteos

**WUOLAH**

realizable o no si hay un árbol de decisión que represente la función verdadera pero esto es un concepto teórico que en la práctica es difícil de llevar a cabo. Si mi espacio de hipótesis son funciones lineales y la función verdadera está expresada a partir de senos y cosenos no voy a poder encontrar una solución óptima a mi problema de aprendizaje pero a lo mejor la solución que encuentre puede ser aceptable de acuerdo a las medidas que se hayan establecido.

## 6.4 Árboles de decisión conceptos básicos

Un árbol de decisión toma como entrada un objeto o una situación descrita a través de un conjunto de atributos y devuelve una "decisión", el valor previsto de la salida dada la entrada. A esta solución también se la denomina clase. Estos árboles pueden recibir de entrada tanto valores discretos (booleans o enumerados) como continuos y a la salida también y si los valores de las etiquetas (clases) son discretos estamos hablando de un problema de clasificación (en los que nos vamos a centrar), por otro lado si los valores de la salida son continuos estamos ante un problema de regresión.

Un árbol de decisión tiene una estructura de árbol normal pero sus nodos no terminales representan una pregunta o una condición sobre un atributo de la tabla, los arcos representan posibles valores de ese atributo y los nodos hoja representan un posible valor de la salida (cierto o falso solamente para nosotros).

En conclusion, podemos usar un árbol de decisión como una función que a partir de una entrada lo que va a hacer va a ser recorrer el árbol haciendo preguntas teniendo en cuenta los nodos del árbol hasta que encuentre un nodo hoja y devolverá el valor del nodo hoja.

### 6.4.1 Construcción de un árbol de decisión

Consiste en un proceso en el que vamos seleccionando atributos y los vamos añadiendo al árbol hasta que todos los ejemplos estén cubiertos y en este proceso hay dos aspectos a los que le debemos prestar atención.

Por un lado el número de preguntas que se tiene que hacer el árbol para llegar a una clasificación, es decir, el número de nodos intermedios del árbol, cuantos menos nodos tenga mejor será.

Por otro lado tenemos que decidir qué criterio usamos para seleccionar unos atributos u otros. Para ello tenemos que observar que cada vez que seleccionamos un atributo creamos una división o partición en el conjunto de ejemplos.

### 6.4.2 Inducción

Para cualquier conjunto de entrenamiento vamos a poder construir un árbol de decisión consistente de forma trivial. Pero seguramente no será bueno para generalizar.

La idea para la construcción de árboles de decisión es que manteniendo la consistencia los árboles sean más compactos que el árbol trivial. Es decir queremos un árbol consistente y simple. Podemos construirlo de varias formas:

- Trivial: crear una ruta del árbol por cada instancia de entrenamiento, esto nos lleva a árboles grandes y poco generalizados.
- Óptimo: el árbol más pequeño posible compatible con todas las instancias. Siguiendo la filosofía de la navaja de Ockham su simplicidad dependerá del número de nodos. En este caso encontrar un árbol lo más pequeño posible y consistente es un problema de optimización que para datos grandes es inviable computacionalmente.
- Pseudo-óptimo: vamos a construir el árbol por pasos, en cada paso vamos a seleccionar un atributo y lo vamos a incluir en el árbol y el criterio de selección va a depender de la calidad de la partición o división que produce el atributo en el conjunto de ejemplos.

Este concepto de calidad se implementará mediante una valoración heurística de atributos.

#### 6.4.2.1 Elección de atributos

Un buen atributo debe dividir el conjunto de ejemplos en subconjuntos que sean o todos positivos o todos negativos.

Con este criterio de selección de atributos basado en información heurística vamos a ver un algoritmo general para aprender árboles de decisión.

#### 6.4.2.2 DTL (Decision tree learning)

Este algoritmo lleva a cabo una estrategia DYV de manera que en cada paso selecciona un atributo de forma greedy (el mejor entre todas las alternativas) y con este atributo seleccionado va a crear una partición de varios subconjuntos de elementos que recursivamente va a aplicar el mismo algoritmo con cada uno de los subconjuntos de la partición.

De manera que este algoritmo recursivo tiene 4 puntos esenciales de decisión:

1. Caso base (no quedan ejemplos) en este caso se va a devolver un valor por defecto asociado a la mayoría de ejemplos en el nodo padre.
2. Caso base: se llega a un nodo hoja, es decir, todos los ejemplos son positivos o negativos, en este caso lo que se devuelve es la etiqueta que se corresponde o bien con los datos positivos o con los negativos.
3. Caso base: no quedan atributos y aún quedan ejemplos se devuelve el valor de la etiqueta de la mayoría de ejemplos.
4. Cuando quedan ejemplos positivos y negativos y todavía no se han agotado los ejemplos ni hemos llegado a un nodo hoja se va a continuar.

```
function DTL(examples, attributes, default) returns a decision tree
    if examples is empty then return default
    else if all examples have the same classification then return the classification
    else if attributes is empty then return MODE(examples)
    else
        best ← CHOOSE-ATTRIBUTE(attributes, examples)
        tree ← a new decision tree with root test best
        for each value  $v_i$  of best do
             $examples_i \leftarrow \{\text{elements of } examples \text{ with } best = v_i\}$ 
            subtree ← DTL( $examples_i$ , attributes − best, MODE(examples))
            add a branch to tree with label  $v_i$  and subtree subtree
        return tree
```

1. No quedan ejemplos: valor por defecto calculado a partir de la mayoría en el nodo padre.
2. Todos los ejemplos son positivos o negativos.
3. No quedan atributos: voto de la mayoría de los ejemplos que quedan.
4. Quedan ejemplos positivos y negativos.

Me han encerrado aquí ¿alguien puede leer esto?

WUOLAH

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.