

WUOLAH



euge4

www.wuolah.com/student/euge4



1522

resumenTema5.pdf

Resumen Tema4+5 Diapos+Libro



2º Inteligencia Artificial



Grado en Ingeniería Informática



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

WUOLAH + #QuédateEnCasa

#KeepCalm #EstudiaUnPoquito

Enhorabuena, por ponerte a estudiar te **regalamos un cartel** incluido entre estos apuntes para estos días.

INTELIGENCIA ARTIFICIAL

Tema 5

INTRODUCCIÓN AL APRENDIZAJE AUTOMÁTICO

Eugenia Castilla Fragoso

ÍNDICE

1. Distintos tipos de aprendizaje

- 1.1. Aprendizaje automático
- 1.2. Estrategias de aprendizaje
- 1.3. Aprendizaje Inductivo
- 1.4. Tipos de aprendizaje automático

2. Modelos inductivos sobre árboles de decisión

- 2.1. Árboles de decisión como herramientas de desarrollo
- 2.2. Expresividad de los árboles de decisión
- 2.3. Inducción de árboles de decisión
 - 2.3.1. Elección de los atributos
 - 2.3.2. Elección de los atributos de test
 - 2.3.3. Ganancia de información
 - 2.3.4. Valoración de la calidad del algoritmo de aprendizaje
 - 2.3.5. Metodología
 - 2.3.6. Ruido y sobreajuste
 - 2.3.7. Valores perdidos

1. Distintos tipos de aprendizaje

1.1. Aprendizaje Automático

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

El aprendizaje sirve para:

- El aprendizaje automático cubre una amplia gama de fenómenos como:
 - el perfeccionamiento de la habilidad
 - la adquisición del conocimiento
- El aprendizaje es esencial en entornos desconocidos
- El aprendizaje modifica el mecanismo de decisión del agente para mejorar su comportamiento
- Aprendizaje automático: programas que mejoran su comportamiento con la experiencia

Definición de aprendizaje automático definitiva: un programa de ordenador se dice que aprende de la experiencia E con respecto a una tarea T y a alguna medida de comportamiento P , si su comportamiento en tareas T , medido a través de P , mejora con la experiencia E .

1.2. Estrategias de aprendizaje

- **Aprendizaje memorístico** → es el más básico
- **Aprendizaje a través de consejos** → se dan consejos que ayudan a la resolución de los programas
- **Aprendizaje en la resolución de problemas** → resuelven muchos problemas y aprendes de la experiencia. No hay intervención.
- **Aprendizaje por refuerzo** → haces una tarea, si la resuelves, vas bien y si no pues vas de culo. Es la técnica mas usada en juegos
- **Aprendizaje a partir de ejemplos: Inducción** → se usan ejemplos y casos de los que se extraen conclusiones. Es el más popular en la minería de datos y en las redes neuronales
- **Aprendizaje basado en explicaciones** → aprende en base a ejemplos, pero necesita menos ejemplos ya que tiene algo de conocimiento previo
- **Aprendizaje a través de descubrimiento** → se resuelven muchos ejemplos y casos parecidos al que se quiere resolver, hasta que finalmente, se resuelve el que se quiere resolver.
- **Aprendizaje por analogía** → se resuelve el problema en base a otros problemas parecidos que ya se han resuelto con anterioridad

1.3. Aprendizaje Inductivo

Aprende a partir de ejemplos.

- Se tiene un conjunto de ejemplos, en base a los cuales se define una función que generaliza a todos los demás.
- Los ejemplos elegidos son muy importantes
- El objetivo es definir la función f tal que dada la hipótesis $h \Rightarrow (h = f)$ sobre los conjuntos de entrenamiento
- Esto se hace con pares de valores $(x, f(x))$.
- Una hipótesis estará bien generalizada si se puede predecir ejemplos que no se conocen

1.4. Tipos de aprendizaje automático

Uno de los puntos clave para el aprendizaje automático es el tipo de realimentación disponible en el proceso:

- **Aprendizaje supervisado:** aprender una función a partir de ejemplos de sus entradas y salidas
 - Método basado en modelos: representan el conocimiento aprendido en algún lenguaje de representación
 - Métodos basados en instancias: representan el conocimiento aprendido como un conjunto de prototipos descritos en el mismo lenguaje usados para representar la evidencia.
- **Aprendizaje no supervisado:** aprender a partir de patrones de entradas para los que no se especifican los valores de sus salidas
- **Aprendizaje inductivo:** una hipótesis estará bien formada si se puede predecir ejemplos que no se conocen. La hipótesis se dice consistente si satisface a todos los datos. Pero, **¿como elegir entre múltiples hipótesis consistentes?** Es preferible la hipótesis consistente con los datos que sea más sencilla.
 - Espacio de hipótesis → las hipótesis se pueden expresar de diversas formas:
 - Árboles de decisión
 - Reglas
 - Redes neuronales
 - Modelos probabilísticos y bayesianos
 - *los árboles de decisión y las reglas son algunos de los modelos más usados en el aprendizaje automático
 - Problema de aprendizaje realizable: se dice que un problema de aprendizaje es realizable si el espacio de hipótesis contiene a la función verdadera, en otro caso se dice que es irrealizable. No siempre se puede determinar si un problema de aprendizaje es realizable, ya que la función verdadera no se conoce.
 - Algoritmos más ampliamente utilizados →
 - Algoritmos basados en Divide y Vencerás (splitting): consisten en ir partiendo sucesivamente los datos en función del valor de un atributo seleccionado cada vez (aprendizaje de árboles de decisión)
 - Algoritmos basados en el “separa y vencerás” (covering): consisten en encontrar condiciones de las reglas que cubran la mayor cantidad de ejemplos de una clase y la menor en el resto de la clase (aprendizaje de reglas)

2. Modelos inductivos sobre árboles de decisión

La inducción de árboles de decisión es uno de los métodos más sencillos y con más éxito para construir algoritmos de aprendizaje

- 2.1. Árboles de decisión como herramienta de desarrollo
- 2.2. Expresividad de los árboles de decisión
- 2.3. Inducción de árboles de decisión

2.1. Árboles de decisión como herramientas de desarrollo

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.

- Atributos → discretos o continuos
- Salida → discreta (clasificación) o continua (regresión)

2.2. Expresividad de los árboles de decisión

Los árboles de decisión pueden expresar cualquier función a partir de los atributos de entrada. De forma trivial, hay un árbol de decisión consistente para cualquier conjunto de entrenamiento con un camino asociado a cada ejemplo, pero seguramente no será bueno para generalizar nuevos ejemplos. Es preferible encontrar árboles de decisión más compactos.

2.3. Inducción de árboles de decisión

Hay muchas formas e inferir el árbol:

- Trivial → se crea una ruta del árbol por cada instancia de entrenamiento
 - árboles excesivamente grandes
 - No funciona bien con instancias nuevas
- Optimo → el árbol mas pequeño posible compatible con todas las instancias (navaja de Ockham)
 - inviable computacionalmente
- Pseudo-optimo (heurístico) → selección del atributo en cada nivel del árbol en función de la calidad de la división que se produce
 - los principales programas de generación de árboles utilizan procedimientos similares

2.3.1. Elección de atributos

Idea = un buen atributo debería dividir el conjunto de ejemplos en subconjuntos que sean o “todos positivos” o “todos negativos”.

Con la función DTL →

- no quedan ejemplos: valor por defecto calculado a partir de la mayoría en el nodo padre
- todos los ejemplos son positivos o negativos
- no quedan atributos: voto de la mayoría de los ejemplos que quedan
- quedan ejemplo positivos y negativos

2.3.2. Elección de los atributos de test

- un atributo perfecto divide los ejemplos en conjuntos que contienen solo ejemplos positivos y negativos.
- Definir una medida de atributo bastante adecuado o inadecuado

$$I(P(v_1), \dots, P(v_n)) = \sum_{i=1} -P(v_i) \log_2 P(v_i)$$

- Para un conjunto de entrenamiento que contenga p ejemplos positivos y n ejemplos negativos

$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$

- Intuición → mide la ausencia de homogeneidad de la clasificación
- Teoría de la Información → cantidad media de información (en bits) necesaria para codificar la clasificación de un ejemplo

2.3.3. Ganancia de información

- Entropía esperada después de usar un atributo A en el árbol

$$resto(A) = \sum_{i=1}^v \frac{p_i + n_i}{p+n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right)$$

- Ganancia de información esperada después de usar un atributo

$$Ganancia(A) = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) - resto(A)$$

- Se elige el atributo con mayor valor de G

2.3.4. Valoración de la ganancia

Un algoritmo de aprendizaje es bueno si produce hipótesis que hacen un buen trabajo al predecir clasificaciones de ejemplos que no han sido observados

2.3.5. Metodología

- recolectar un conjunto de ejemplos grande
- dividir el conjunto de ejemplos en dos conjuntos: el conjunto de entrenamiento y el conjunto de test
- aplicar el algoritmo de aprendizaje al conjunto de entrenamiento, generando la hipótesis h
- medir el porcentaje de ejemplos del conjunto de test que h clasifica correctamente
- repetir los pasos del 1 al 4 para conjuntos de entrenamiento seleccionados aleatoriamente para cada tamaño.

2.3.6. Ruido y sobreajuste

- Ruido → dos o mas ejemplos con la misma descripción pero diferentes clasificaciones
- Sobreajuste → encontrar regularidades poco significativas en los datos

Se dice que una hipótesis h se sobreajusta al conjunto de entrenamiento si existe otra hipótesis h' tal que el error de h es menor que el de h' sobre el conjunto de entrenamiento, pero es mayor sobre la distribución completa de ejemplos del problema(entrenamiento + test)

2.3.7. Valores perdidos

- asignar el valor más común entre todos los ejemplos de entrenamiento pertenecientes al nodo
- asignar una probabilidad a cada uno de los posibles valores del atributo basada en la frecuencia observada en los ejemplos pertenecientes al nodo. Finalmente, distribuir de acuerdo a dicha probabilidad.