



Computación Inteligente

Máster en Ingeniería en Informática

Área de Ciencias de la Computación

e Inteligencia Artificial

Departamento de Tecnologías de la Información

## PRÁCTICA 1 (Versión 1 – Noviembre de 2021)

### Soft Computing

#### Objetivos.

El objetivo de esta práctica es implementar sistemas difusos completos, específicamente, un Sistema Basado en Reglas Difusas para Regresión (o Control) (SBRD), y un Sistema de Clasificación Basado en Reglas Difusas (SCBRDs). La práctica comprenderá todas las etapas de los mismos, desde la generación de reglas a partir de un conjunto de datos (aprendizaje de reglas), la realización del código del algoritmo de regresión y clasificación, y opcionalmente, para el caso de la regresión, un mecanismo de post-procesamiento para reducir el número de reglas basado en algoritmos evolutivos.

#### Enunciado de la práctica

Los SBRDs y SCBRDs son aplicaciones típicas de los sistemas difusos. Ambas son bien conocidas por su capacidad expresiva o de interpretabilidad por poder mezclar el conocimiento aportado por el experto con el obtenido o aprendido por métodos automáticos, y por su capacidad interpolativa, es decir, por hacer participar a las diferentes reglas de la decisión final y no necesariamente a una única regla concreta de su Base de Conocimiento.

Trabajaremos con conjuntos de datos o ejemplos (*datasets*) típicos de un repositorio reconocido, aprenderemos el conjunto de reglas mediante la adaptación de un algoritmo derivado de uno de los clásicos, implementaremos la regresión y la clasificación, y verificaremos su comportamiento, además de aplicar un post-procesamiento evolutivo de selección de reglas para reducir el conjunto de ellas en el caso de la regresión.

## ***Implementación de un Algoritmo de Regresión o Control Difuso***

Se trata de implementar el algoritmo de regresión o control *fuzzy* completo, es decir, con capacidad de predecir instancias nuevas empleando su Base de Conocimiento previamente aprendida mediante los algoritmos estudiados en las sesiones teóricas.

Los operadores y mecanismos que se deben implementar son:

- el *producto* (i.e. *t-norma* del *producto algebraico*) para el *matching*,
- el *mínimo* (i.e. *t-norma* del *producto lógico*) para la inferencia,
- el *Punto de Máximo Valor ponderado por el matching* para la desborrosificación.

## ***Aprendizaje de la Base de Conocimiento***

El conjunto de ejemplos que debe emplear es el *delta ail*, con 5 variables, y sobre 7000 datos, que puede descargar desde <https://sci2s.ugr.es/keel/category.php?cat=reg>

El universo de las variables lo particionará en 3 y 5 etiquetas, para comparar los resultados.

La base de reglas debe aprenderse con el algoritmo estudiado en las sesiones de teoría, el archiconocido *Método de Wang y Mendel*, que encuentra las reglas por un criterio de cubrimiento y calidad individual de cada regla.

El conjunto de datos se dividirá en partes, como se explicará en la sección sobre la experimentación.

## ***Evaluación del Sistema de Regresión o Control***

Se empleará el error cuadrático medio entre la solución ideal y la obtenida, es decir, consistirá en medir la precisión (ver medida en la teoría) con todo el conjunto de prueba (ver sección de experimentación posterior).

## ***Algoritmo Evolutivo (opcional) para la Selección de Reglas***

Se implementará un algoritmo genético generacional para la selección de reglas, es decir, para obtener un conjunto menor y más compacto de ellas, al tiempo que más preciso. Empleará codificación binaria, y los siguientes operadores genéticos y otras características:

- Operador de cruce en un punto. Probabilidad de cruce 0.9.
- Operador de mutación aleatorio uniforme sobre el cromosoma de 0.01 (una vez el cromosoma debe mutar con probabilidad 0.01, aleatoriamente se escogerá el cromosoma que muta).
- 21 cromosomas como longitud de población.
- La inicialización de los cromosomas será: un cromosoma con todas las reglas activas (esto es, genes a 1), y el resto de cromosomas aleatoriamente.
- Número de generaciones: 50 sin mejora en el mejor cromosoma.

- Puede emplear elitismo, de parámetro 3.
- La función objetivo mediará-el error cuadrático medio.

### ***Implementación de un Algoritmo de Clasificación Difuso***

Se trata de implementar el algoritmo de clasificación *fuzzy* completo, es decir, con capacidad de clasificar instancias nuevas empleando su Base de Conocimiento previamente aprendida mediante los algoritmos estudiados en las sesiones teóricas.

Los operadores y mecanismos que se deben implementar son

- el *mínimo* (i.e. *t-norma* del *producto lógico*) para el *matching*,
- el *producto algebraico* para la *función de ponderación*,
- el método de *Regla Ganadora*.

### ***Aprendizaje del Clasificador***

El conjunto de ejemplos que debe emplear es el Glass1, con 9 variables, 7 clases, y 214 ejemplos, que puede descargar desde <https://sci2s.ugr.es/keel/category.php?cat=clas>

El universo de las variables lo particionará en 3 y 5 etiquetas, para comparar los resultados.

La base de reglas debe aprenderse con el algoritmo estudiado en las sesiones de teoría, conocido como método *Chi*, derivado del archiconocido *Método de Wang y Mendel*, que encuentra las reglas por un criterio de cubrimiento y calidad individual de cada regla.

El conjunto de datos se dividirá en partes, como se explicará en la sección sobre la experimentación.

### ***Evaluación del Clasificador***

La precisión del clasificador con un conjunto de ejemplos, la cual se llevará a cabo mediante esta ecuación clásica empleada habitualmente en la medida de error de clasificadores:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

donde *TP*, *TN*, *FP* y *FN* son respectivamente los contadores de *True Positive*, *True Negative*, *False Positive* y *False Negative*, (*Positivo Verdadero*, *Negativo Verdadero*, *Positivo Falso* y *Negativo Falso*), es decir, se divide el número de predicciones correctas entre el número total de predicciones (correctas más erróneas).

## ***Experimentación y Presentación de los Resultados***

Se trabajará con un modelo de validación cruzada (de orden 5) con 5 particiones de datos, esto es, 5 particiones aleatorias al 20%, y se crearán 5 particiones combinando grupos de 4 al 80%, y una restante del 20% como prueba. Los datasets que descarga del repositorio, vienen ya particionados: si no lo estuviesen hagan la partición anterior por vd. Mismo.

Se presentarán los resultados en dos tablas, una para el problema de regresión y otra para el de clasificación, en las que se mostrarán los resultados con el conjunto de entrenamiento y prueba en columnas diferentes. Dado que empleará particiones de 3 y 5 etiquetas en cada caso, debe incluir ambos resultados.

Si realiza la parte opcional de regresión (el algoritmos evolutivo) incluya la tabla de resultados del mismo, es decir, indique si usó 3 ó 5 etiquetas, y cómo mejora con respecto a la versión no ajustada (en número de reglas y en precisión).

Los resultados obtenidos con 3 y 5 etiquetas deben compararse, es decir, hay que hacer un estudio de ellos y deducir conclusiones entre ambas situaciones, más que implementar y quedarse sólo en los resultados obtenidos: lo relevante es poder llegar a hacer el análisis y hacerlo correctamente.

### **Fecha y Método de Entrega:**

Las prácticas se realizarán individualmente. El día 20 de Diciembre debe estar subido a Moodle antes de las 23:55 sin retrasos en un único fichero comprimido ZIP, que contenga:

- Documento DOC (MS Word) ó PDF con las tablas de resultados y análisis.
- Ficheros de código fuente completo ejecutable utilizado.
- *Scripts*, si los ha utilizado.

Permanezca atento/a a posibles versiones mejoradas de este documento.