

UNIVERSIDAD DE JAÉN
ESCUELA POLITÉCNICA SUPERIOR DE JAÉN



Departamento de Informática. Área de Ciencias de la
Computación e Inteligencia Artificial

Sistemas Difusos

Guión de Prácticas

Práctica:
Sistema Basado en Reglas Difusas

Curso 2008-2009

Segundo Curso de la Ingeniería Informática

Profesora: María José Gacto

Práctica 1

Sistema Basado en Reglas Difusas

1. Objetivos

El objetivo de esta práctica consiste en probar y analizar el funcionamiento de varios de los métodos de obtención de reglas difusas (Fuzzy Rule Learning) con dos dataset de regresión. Más en concreto analizaremos el funcionamiento del método WM, los métodos MOGUL Mamdani y los métodos MOGUL TSK. Para ello utilizaremos la herramienta KEEL (<http://www.keel.es>).

La práctica se califica sobre 5 puntos. Esta puntuación se distribuye de la siguiente forma:

- Algoritmo WM 2 puntos (obligatorio)
- Algoritmos Mogul Mamdani 1,25 puntos (opcional)
- Algoritmos TSK 1,25 puntos (opcional)
- Defensa de prácticas 0,5 puntos

La fecha límite de entrega será el día 28/05/2009 en formato papel durante el horario de prácticas o en formato electrónico a través de la plataforma ILIAS. La defensa se realizará el día 04/06/2009 en horario de prácticas.

2. Dataset

En la práctica se van a realizar experimentos de Regresión, utilizando particiones 10-fold cross validation. Los dataset que se deben ser utilizados en la práctica son los siguientes:

- Electrical-Length: 3 atributos y 495 casos. Este dataset esta disponible en Keel con el nombre **ELE1**

- Electrical-Maintenance: 5 atributos y 1056 casos. Este dataset no esta disponible en keel debe importarse utilizando el fichero Electrical.rar suministrado junto al gui3n de pr3cticas.

Para cada algoritmo se obtendr3n una tabla de resultados que incluir3 al menos la siguiente informaci3n:

- N3mero medio de reglas obtenidas
- Error cuadr3tico medio obtenido en entrenamiento o training de las 10 particiones (MSE)
- Desviaci3n del error cuadr3tico medio obtenido en entrenamiento (stdev)
- Error cuadr3tico medio obtenido en prueba o test de las 10 particiones.
- Desviaci3n del error cuadr3tico medio obtenido en prueba.

La herramienta Keel genera un fichero .zip para realizar las experimentaciones. Este fichero debe descomprimirse y desde la ventana del MS-DOS ejecutar usando "java -jar RunKeel.jar"

3. M3todo WM

Este es un m3todo de generaci3n de reglas difusas mediante el aprendizaje a partir de ejemplos.

En esta secci3n se analizara el comportamiento del m3todo WM [1] en los dos dataset mencionados previamente. Para ello se deben hacer las siguientes experimentaciones:

1. Experimentaci3n utilizando el *m3todo wm* en keel **Regr-Fuzzy-WM**.
2. Aplicaci3n de un m3todo de postprocesamiento (Fuzzy Rule PostProcessing), para realizar la *selecci3n de reglas* en keel **Post-Rules-Selection**.
3. Aplicaci3n de un m3todo de postprocesamiento para el *ajuste de las funciones de pertenencia* en keel **Post-G-G-Tuning-FRBSs**.
4. Para poder analizar el comportamiento de los m3todos es conveniente utilizar la herramienta de keel **Shows Results/Vis-Regr-Check**

En la siguiente gr3fica 1.1 se muestra un ejemplo de un experimento con WM

Adem3s de la *tablas de resultados*, hay que incluir un *an3lisis completo de los resultados* obtenidos que incluyan gr3ficas de resultados (tanto en entrenamiento como en prueba) y comparativas de como influye la selecci3n y el ajuste de las funciones de pertenencia sobre los resultados obtenidos.

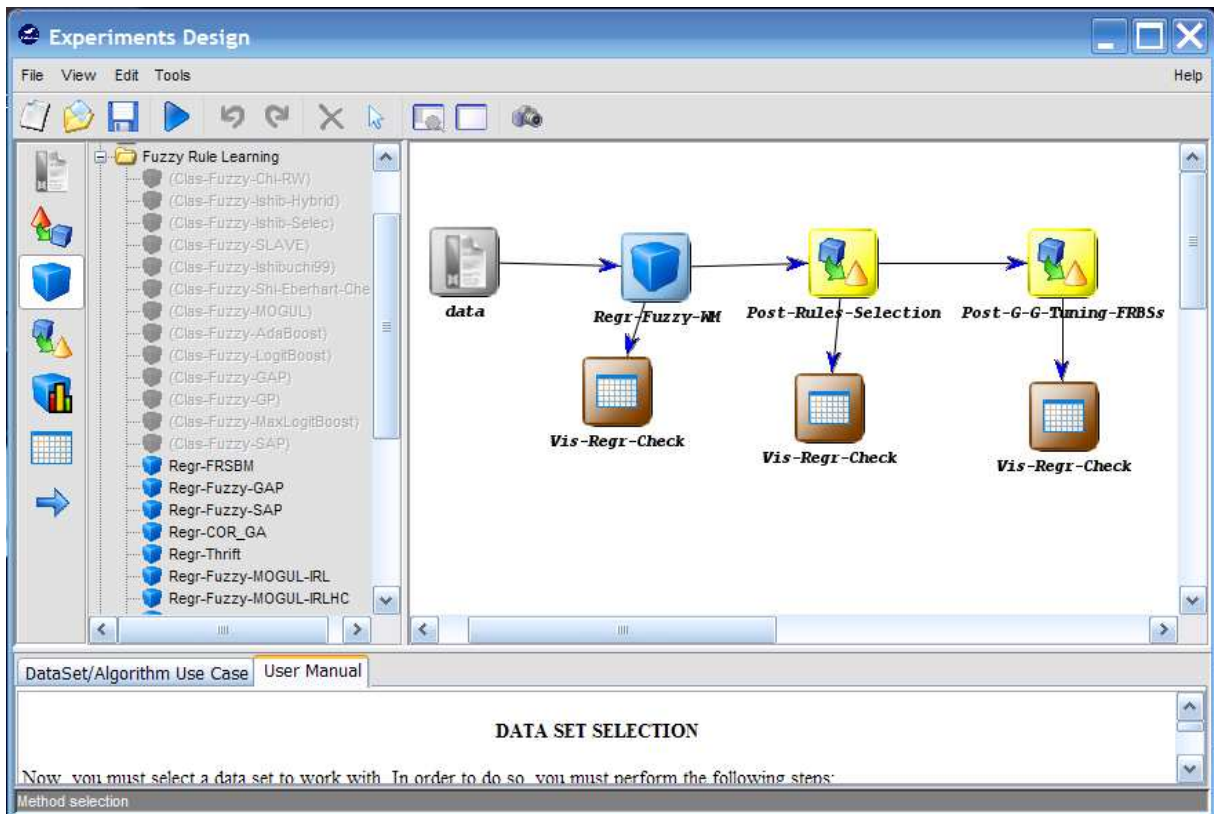


Figura 1.1: Ejemplo de experimento con WM

Por último, realizar un **análisis** comprobando cuál/es de las siguientes **propiedades cumple la base de reglas obtenida con WM** (antes de la etapa de selección y ajuste) y por qué (consistencia, completitud y continuidad). Hay que realizar dos análisis (uno para cada uno de los dataset con una sola de las particiones). Para realizar este análisis se puede utilizar (los resultados obtenidos dentro del directorio results/Regr-Fuzzy-WM) en dos ficheros:

- result0e0.txt que contiene la base de reglas
- result0e1.txt que contiene la base de datos

Gracias a estos ficheros se puede realizar el análisis de la base de reglas.

4. Métodos MOGUL Mamdani

En esta sección realizaremos experimentación y análisis de métodos de aprendizaje descriptivo y aproximativo de reglas mamdani. Para ello se van a utilizar los métodos siguientes:

- Aprendizaje de reglas descriptivo [2] en keel **Regr-Fuzzy-MOGUL-IRL** (3 etapas: obtención de reglas, ajuste y selección)
- Aprendizaje de reglas aproximativo [2] en keel **Regr-Fuzzy-MOGUL-IRLSC** (3 etapas: obtención de reglas, ajuste y selección)

En la siguiente gráfica 1.2 se muestra un ejemplo de un experimento con los métodos Mogul Mamdani.



Figura 1.2: Ejemplo de experimento con MOGUL

Para estos experimentos se debe utilizar una semilla concreta que será vuestro número de DNI(sin letra).

Como resultado se deben incluir gráficas y tablas (tanto en entrenamiento como en prueba) analizando los resultados obtenidos por los dos métodos propuestos. Por último, en la documentación hay que incluir un análisis comparando los resultados obtenidos con el método WM contra los resultados obtenidos con los métodos mogul.

5. Métodos TST

En esta sección realizaremos experimentación y análisis de métodos de aprendizaje descriptivo de reglas TSK. Para ello se van a utilizar los métodos siguientes:

- Aprendizaje descriptivo de reglas TSK [3] en keel **Regr-Fuzzy-TSK-IRL** (2 etapas: obtención de reglas y ajuste)
- Aprendizaje descriptivo de reglas TSK [4] en keel **Regr-Fuzzy-MOGUL-TSK** (3 etapas: obtención de reglas, ajuste y selección)

Para estos experimentos se debe utilizar como semilla vuestro número de DNI.

Como resultado se deben incluir gráficas y tablas (tanto en entrenamiento como en prueba) analizando los resultados obtenidos por los dos métodos propuestos. Por último, en la documentación hay que incluir un análisis comparando los resultados obtenidos con el método WM y con los métodos mogul mamdani contra los resultados obtenidos con los métodos TSK.

6. Documentación a Entregar

La documentación de la práctica deberá incluir, al menos, el siguiente contenido:

1. Portada con el título de la práctica, curso académico, nombre, DNI y dirección e-mail del alumno.
2. Índice de la documentación incluyendo la numeración de las páginas.
3. **Breve** descripción del algoritmo WM (**máximo 1 página**): ¿en qué consiste, para qué sirve?
4. Experimentos y análisis de resultados del método WM.
5. **Breve** descripción de los algoritmos MOGUL Mamdani (**máximo 2 página**): ¿en qué consisten, para qué sirven?
6. Experimentos y análisis de resultados de los métodos MOGUL Mamdani.
7. **Breve** descripción de los algoritmos TSK (**máximo 2 página**): ¿en qué consisten, para qué sirven?
8. Experimentos y análisis de resultados de los métodos TSK.
9. Conclusión final. Debeis incluir una conclusión general de cual de los métodos es más adecuado teniendo en cuenta consideraciones como resultados en entrenamiento-prueba, tiempo de ejecución, número de reglas, etc...
10. Referencias bibliográficas, si procede.

Es muy importante en esta práctica tener un buen **Análisis de resultados** y deberá estar orientado a justificar (según el comportamiento de cada algoritmo) los resultados obtenidos en lugar de realizar una mera "lectura" de las tablas. Se valorará la inclusión de otros elementos de comparación tales como gráficas, análisis comparativo de las soluciones obtenidas, etc.

Para mayor claridad de la documentación, se recomienda comenzar cada sección o descripción en una página nueva. Emplear un formato ilegible podrá influir negativamente (hasta en un 50 %) en la evaluación final de la práctica.

Además se debe incluir en el ILIAS (o en CD) un fichero .wml que incluya los experimentos y con los directorios con los resultados de las ejecuciones realizadas para todos de los métodos y dataset.

Bibliografía

- [1] L.-X. Wang y J.M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man, and Cybernetics* 22:6 (1992) 1414-1427.
- [2] O. Cordón, F. Herrera, A three-stage evolutionary process for learning descriptive and approximate fuzzy logic controller knowledge bases from examples. *International Journal of Approximate Reasoning* 17:4 (1997) 369-407.
- [3] O. Cordón, F. Herrera, A Two-Stage Evolutionary Process for Designing TSK Fuzzy Rule-Based Systems, *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics* 29:6 (1999) 703-715.
- [4] R. Alcalá, J. Alcalá-Fdez, J. Casillas, O. Cordón, F. Herrera, Local Identification of Prototypes for Genetic Learning of Accurate TSK Fuzzy Rule-Based Systems. *International Journal of Intelligent Systems* 22:9 (2007) 909-941