

Introducción al Diseño de Experimentos para el Reconocimiento de Patrones

Capítulo 5: Sistemas Modulares, Mezcla de Expertos y Sistemas Híbridos

Curso de doctorado impartido por
Dr. Quiliano Isaac Moro
Dra. Aranzazu Simón Hurtado
Marzo 2004

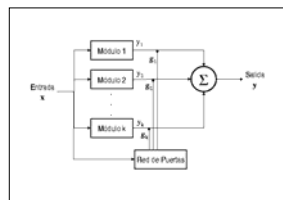
Contenido

1. Introducción: Módulos, Expertos e Hibridación
2. Sistemas Globales vs. Sistemas Locales
3. Ventajas e inconvenientes de la aproximación modular
4. Elementos a considerar
 1. Descomposición Específica vs. Sistemática del entorno
 2. Comunicación entre los módulos: Reparto de información, e integración de resultados
5. Aprendizaje – Ajuste de lo módulos
6. Ajuste del tamaño y estructura de los módulos
7. Ajuste de la arquitectura modular
8. Extracción de reglas del sistema ya ajustado
9. Ejemplos

2

Introducción

- Fruto del análisis: distinguir las partes y el todo.
 - Táctica del "divide y vencerás".
- ¿Se puede sistematizar?
- Módulos/Expertos/Hibridación
 - Cada herramienta tiene su ámbito de aplicación.
 - Se busca quedarse con lo mejor de cada una.
 - Idea de modularización: división tarea → subtareas. ¿Cómo?
 - División sistemática / específica.
 - Intuitivamente podemos identificar con no supervisada / supervisada.

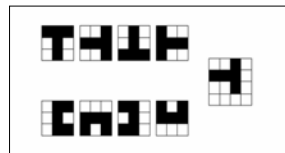


3

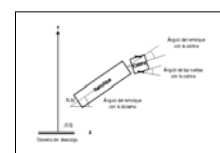
Sistemas Globales / Locales

- Sistemas globales (RNA)
 - El paradigma más usado (bp) no justifica resultados de forma razonada.
 - Problemas complejos → gran número de pesos a entrenar.
 - Riesgo de sobreentrenamiento.
 - Técnicas: early stopping, weight decay, poda....
 - **Interferencia:** durante el entrenamiento se recibe información contradictoria, ya sea de forma simultánea (interferencia espacial), o en instantes diferentes de tiempo (interferencia temporal).

• Espacial.



• Temporal.



4

Ventajas e Inconvenientes

- Aproximación local (modular)...
 - Aumenta la velocidad de aprendizaje
 - Cada módulo es menos complejo → menos parámetros a ajustar.
 - Es más fácil llegar a comprender la tarea de la que se ha hecho responsable un módulo
 - Es coherente con las limitaciones de espacio que se presentan en los modelos biológicos.
 - Relativa sencillez de los módulos constituyentes.
 - Cada subtarea se hace responsable de un subproceso elemental.
 - Cada módulo puede ser construido de manera diferente
 - sistemas heterogéneos → sistema híbridos.
 - Menos propenso al sobreentrenamiento.
 - Es más difícil de aplicar:
 - Definir módulos, estrategias de entrenamiento, de integración de resultados...

5

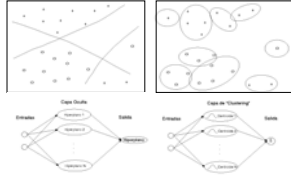
Consideraciones generales

- No es una idea nueva: La Place (1818)
- Puntos a considerar en la modularización
 1. Descomposición de la tarea principal en subtareas,
 2. Determinación de la naturaleza del módulo que se asocia a cada subtarea.
 3. Organización de los módulos conseguidos según una arquitectura apropiada, y
 4. Establecimiento de las líneas de comunicación entre dichos módulos a dos niveles:
 1. reparto de la información durante la etapa de aprendizaje, e
 2. integración de la información a la hora de generar la salida del sistema completo.

6

Descomposición en subtareas

- **Descomposición sistemática**
 - Ampliamente usado en tareas de clasificación
 - Cada módulo se encarga de reconocer una clase específica.
 - Ejemplos en RNA:
 - RBF,
 - SOM.
 - Se usa un criterio de vecindad espacial / temporal.
- **Descomposición específica "Ad Hoc".**
 - El concepto de proximidad o distancia no está bien definido.
 - Ej: conducción de un coche.



7

Comunicación

Comunicación entre módulos

- ¿Cómo hacer interaccionar los distintos módulos para que el sistema completo realice la tarea objetivo?
 - Normalmente esta tarea se lleva a cabo por medio de un elemento, módulo, o capa que permite establecer una decisión a la vista de los resultados ofrecidos por los distintos módulos.
- Se puede distinguir dos aspectos:
 - Cómo repartir la información durante la etapa de entrenamiento entre los distintos módulos.
 - Cómo integrar los distintos resultados ofrecidos por los diferentes módulos constitutivos del sistema para generar la salida final.

8

Comunicación

Reparto de la Información

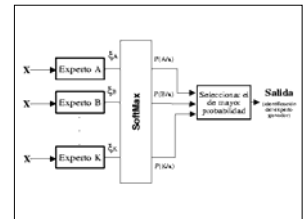
- equivalente a
 - determinar el módulo a ajustar para aprender un determinado ejemplo, y
 - de qué magnitud debe ser ese ajuste.
- Algunas soluciones:
 - Criterio de proximidad.
 - Puede reducirse al caso "el ganador se lo lleva todo" (WTA), o bien
 - puede haber varios ganadores.
 - Función lógica.
 - Caso particular: variable índice que selecciona el módulo a ajustar.
 - Automata de estados finito.
 - Lógica borrosa.
 - Técnicas estadísticas.

9

Comunicación

Integración de resultados

- Supongamos módulos ya ajustados.
 - Acorde al reparto de la información y descomposición en subtareas.
 - "WTA". Cada módulo ofrece resultados homogéneos.
 - Votación. Cada experto ofrece un grado de certeza.
 - Se suele usar SoftMax
 - La salida del sistema será la etiqueta del módulo que ofrece mayor certeza.
 - Combinación Lineal.
 - Puede incluir una etapa de ajuste para minimizar el error de esa combinación (suma ponderada)
- Cuando los resultados son heterogéneos, se suele usar un esquema en serie.
- Técnicas probabilísticas



10

Ajuste de módulos. Aprendizaje.

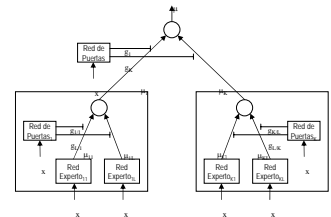
- **Descomposición "Ad Hoc"** → definido por dicha descomposición (subtarea asignada).
- **Aprendizaje progresivo.**
- **Descomposición genérica.**
 - Procedimientos basados en la homogeneidad de los módulos.
 - Es sistematizable.
 - Ejemplos
 - Aprendizaje estocástico. Sigue el esquema de Jacobs-Jordan.
 - » Mezcla de procesos estocásticos.
 - Maximización del Valor Esperado (EM).

11

Ajuste de módulos: Aprendizaje

Ej: Modelo Jakobs-Jordan.

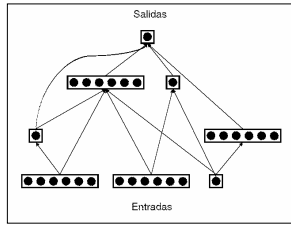
- Primer nivel: conglomerado de expertos que combinan sus salidas por medio de una "red de puertas"
- Segundo nivel: combina los resultados de todos los bloques de expertos.
 - Todos los bloques reciben el mismo vector de entrada.
- El proceso de ajuste tiende que los conglomerados se especialicen en datos de clases mutuamente excluyentes.



12

Ajuste de la Arquitectura Modular

- **Determinar para el sistema:**
 - Número y tipo de módulos.
 - En problemas de clasificación \approx nº de clases. Cada módulo construido en detectar cada clase.
 - Arquitectura que los relaciona.
- **Con información a priori**
 - Se tiende a sistemas supervisados.
- **Sin información a priori.**
 - Se tiende a sistemas no supervisados.
 - Se tiene que hacer una búsqueda, p.ej. con
 - Método de Monte Carlo.
 - Algoritmos genéticos.
 - Ejemplo: "TC" con RNA y algoritmos genéticos.



13

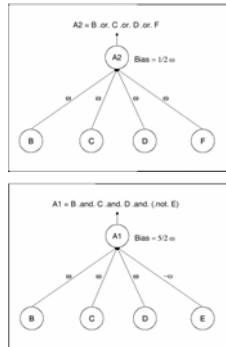
Sistemas Híbridos

- **Algunos modelos de RNA son por su propia definición sistemas híbridos:**
 - **Redes Contra-propagación.**
 - Una primera capa realiza un aprendizaje no supervisado competitivo.
 - Una segunda capa realiza un aprendizaje supervisado para asignar etiquetas.
 - **RBF.**
 - Una primera capa con funciones de base radial para calcular distancias (entrada \rightarrow centroide).
 - Segunda capa que aprende de forma supervisada a generar la salida deseada.
 - **ARTMAP**
 - Aprendizaje no supervisado en la entrada y salida, con lo que se hacen clusters en ambos espacios.
 - En la capa intermedia se aprende de forma supervisada a asociar un cluster de entrada con otro de salida.

14

Extracción de reglas de las RNA

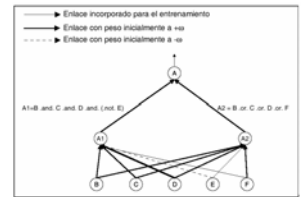
- **Ej: KBNN.**
 - Entradas y salidas binarias {0,1}
 - Se parte de un conocimiento (parcial) en forma de reglas del dominio.
 - Cada regla se implementa como una subred neuronal.



15

Extracción de reglas de las RNA

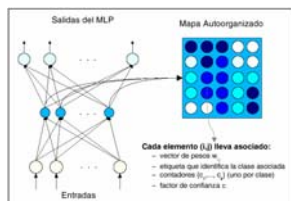
- **Ej: KBNN.**
 - Se construye el sistema completo (con todas sus reglas) y se completa con las conexiones no existentes inicializadas con pesos aleatorios.
 - Se entrena.
 - Se observan los pesos de las conexiones de los submódulos y se interpretan como nuevas reglas.



16

Ejemplos de Sistemas Modulares / Híbridos

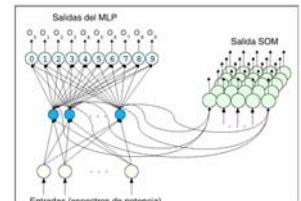
- **BP-SOM**
 - Idea: forzar que la representación en la capa oculta de entradas parecidas (próximas) también estén próximas en el espacio de la capa oculta del MLP.
 - Sistema formado por un MLP y un SOM.
 - Coexisten sólo durante la etapa de aprendizaje.
 - Otras ventajas
 - ¿Extraer reglas de la red?
 - Optimización del tamaño de la capa oculta.



17

Ejemplos de Sistemas Modulares / Híbridos

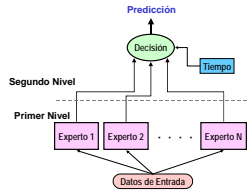
- **Reconocimiento automático del habla.**
 - El MLP es entrenado para identificar uno de los 10 dígitos en las muestras de voz.
 - Aquellos casos dudosos en los que la activación de los 10 elementos de proceso de salida no nos permita afirmar con certeza cuál es el dígito, se consulta el resultado del SOM.



18

Ejemplos de Sistemas Modulares / Híbridos

- Predicción meteorológica: se establece una jerarquía de módulos a dos niveles:
 - Primer nivel: nodos expertos en el pronóstico sobre un determinado intervalo del año.
 - Garantiza la cuasi-estacionariedad de la serie.
 - Segundo nivel: un único nodo que decide qué valor ofrecer como predicción.
 - Todos los nodos son MLP's.
 - Una vez entrenado el sistema, se utilizará para ofrecer predicciones para cualquier época del año.
 - Hipótesis: los resultados individuales se compensan y complementan al combinarlos en el segundo nivel.



19

Ejemplos de Sistemas Modulares / Híbridos

Resultados:

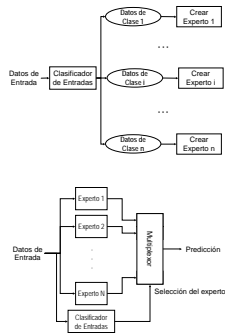
- Parámetros variados
 - Ancho de ventana de datos
 - Número de elementos de proceso en la capa oculta de los MLP.
 - Número de expertos en el primer nivel (=intervalos en los que se se divide el año).
- Mejor resultado en validación: error medio= $\pm 1.651^{\circ}\text{C}$
 - Ancho de ventana de datos 6.
 - 9 expertos (cada uno abarca unos 40 días)

(la variación media de la temperatura es de $\pm 2.3^{\circ}\text{C}$)

20

Ejemplos de Sistemas Modulares / Híbridos

- Otro ej. de predicción meteorológica:
 - Existen 2 niveles; el segundo contiene los predictores expertos (MLP).
 - La asignación de datos a cada experto se hace de manera no supervisada por medio de un mapa autoorganizado de Kohonen en el primer nivel.
 - Una vez entrenado, el sistema se utiliza para efectuar predicciones en cualquier momento del año.
 - Tiene diferentes esquemas de procesamiento en
 - Aprendizaje
 - Funcionamiento



21

Ejemplos de Sistemas Modulares / Híbridos

Resultados

- Parámetros variados:
 - Dimensión del SOM (de 1x2 hasta 8x8)
 - Ancho ventana de datos
 - Elementos de proceso de la capa oculta del MLP
- Mejor resultado en validación: error medio= $\pm 1.652^{\circ}\text{C}$
 - Tamaño del SOM: 5 x 2
 - Ancho ventana datos: 1

Orden	Tamaño Mapa	Ancho Ventana	Error Medio Validación
1	10	1	1.652
2	20	1	1.652
3	28	2	1.660
4	3	5	1.663
5	6	2	1.668
6	20	2	1.679
7	18	1	1.681
8	24	2	1.686
9	14	2	1.687
10	8	1	1.687

22

Principio de "La Navaja de Occan"

"El Modelo que con más probabilidad haya generado un conjunto de datos experimentales seguramente es el que depende del menor número de parámetros"

- Son preferibles los modelos con pocos parámetros.
 - Son menos propensos al sobreentrenamiento.
 - Son más fáciles de entrenar.
 - Generalizan mejor.

23