Lecture 5: Neuro-fuzzy Learning and Inference Systems

DENFIS: Dynamic Evolving Neural-fuzzy Inference System and its Application for Time-series Prediction [1]

DENFIS Software in Python

[1] N. K. Kasabov and Q. Song (2002), "DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction," in IEEE Transactions on Fuzzy Systems, vol. 10, no. 2, pp. 144-154, doi:10.1109/91.995117.

1- Topic/Task/Problem Specification:

- The topic revolves around adaptive learning strategies for dynamic time series prediction in online and offline environments.
- The complexity and dynamics of real-world problems require sophisticated methods for building intelligent systems that can evolve as they operate.
- Such systems are expected to continuously update their knowledge and refine their models through ongoing interactions with their environment.
- Major requirements of these intelligent systems:
  1. fast learning,
  2. online incremental adaptive learning,
  3. open structure organization,
  4. memorising information,
  5. active interactions,
  6. knowledge acquisition and self-improvement,
  7. spatial and temporal learning.

2- Previously Published Methods for Solving the Problem:

- To address these requirements, traditional neural networks, such as multilayer perceptron (MLPs) trained with backpropagation, radial basis function networks (RBF), and self-organizing maps (SOMs), are insufficient - not designed for adaptive online learning.

- Instead, evolving connectionist systems (ECOSs) are systems that evolve their structure and functionality from a continuous input data stream in an adaptive, life-long, modular way, creating connectionist-based modules according to the input data distribution and the system's performance at a given time.

- The best manifestations of ECOS principles are the evolving fuzzy neural network (EFuNN) model, a fuzzy logic system with a five-layer structure that creates connectionist modules in an ECOS architecture, and the Dynamic Evolving Neural-Fuzzy Inference System (DENFIS) model.

3- Description of the Method in the Paper Under Discussion:

- DENFIS is a sophisticated adaptive learning system that evolves by incremental hybrid learning while accommodating new input data, including new features and classes.
- DENFIS dynamically generates and updates fuzzy rules while in operation, adjusting its output using a fuzzy inference system based on the most activated fuzzy rules selected dynamically from a set.
- DENFIS is an effective technique for complex problem-solving in dynamic environments.
- DENFIS offers dynamic adaptability to new data, real-time knowledge base refinement, and flexibility for online and offline learning scenarios.
- DENFIS is dynamic in nature, creating and adjusting fuzzy rules in response to incoming input.

4- Learning Methods in DENFIS:

Takagi-Sugeno fuzzy inference in DENFIS is a dynamic inference.

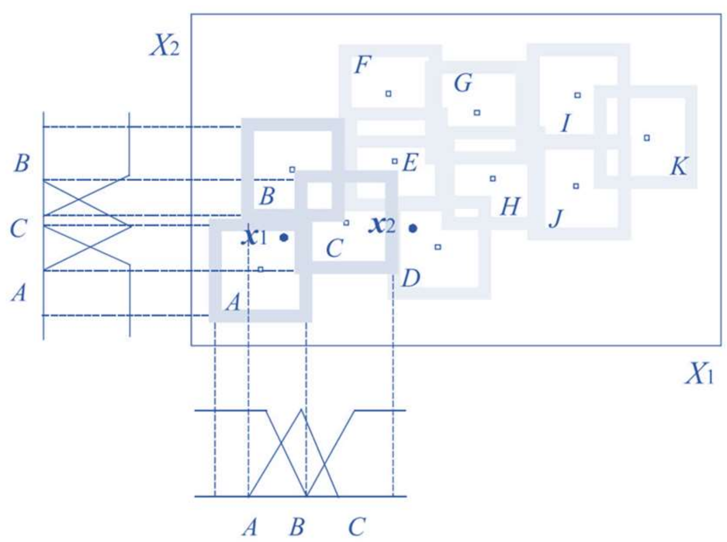
a. **Online Learning**: Directly incorporates new data into the model, adjusting rules and predictions on the fly.
   - In the DENFIS online model, if two input vectors are exactly the same, their corresponding inference systems may be different. This is because these two input vectors are presented to the system at different time moments, and the fuzzy rules used for the first input vector might have been updated before the second input vector arrived.

b. **Offline Learning**: Analysis of batches of historical data to refine and improve the model's predictive accuracy.
   - In the DENFIS offline model, if two input vectors are very close to each other, their corresponding inference systems may have the same fuzzy rule inference group.

c. **Least-Square Estimator (LSE):** A mathematical approach used to optimize the model's parameters, ensuring accurate predictions.
   - For each input vector, the DENFIS model chooses **m** fuzzy rules from the whole fuzzy rule set to form a current inference system. This operation depends on the current input vector's position in the input space.

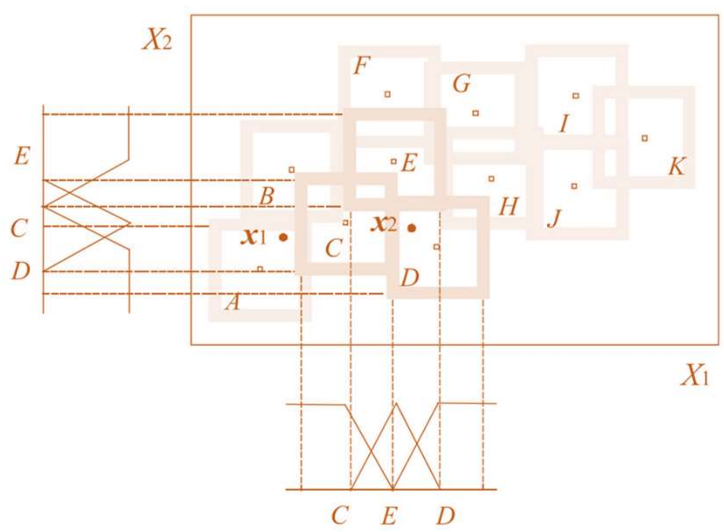
4- Learning Methods in DENFIS (cont'd):

(a) Fuzzy rule group 1 for a DENFIS (m = 3)



- In this example, two different groups of fuzzy inference rules are formed depending on two input vectors, x1 and x2, respectively, in a 2D input space.
- Region **C** has a linguistic meaning of '**large**' in the x1 direction in (a) but a linguistic meaning of '**small**' in the same direction of x1 in group (b).
- This denotes that region **C** is defined by different membership functions respectively in each of the two groups of rules.

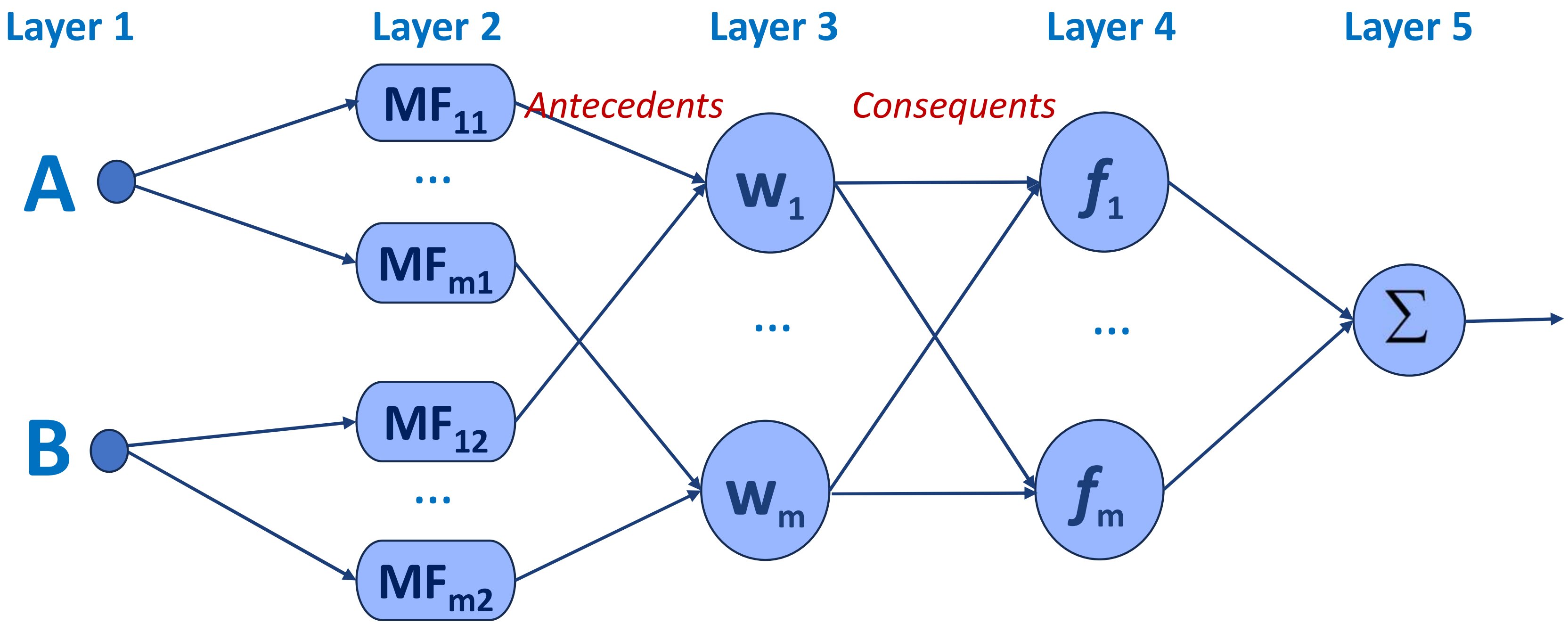
(b) Fuzzy rule group 2 for a DENFIS (m = 3)



5- The Architecture of DENFIS - *Fuzzy rules are at the heart of the DENFIS model*:

Layer 1:

- This is the input layer that captures input vectors and processes raw data points.
- Input data is normalized to ensure consistency for better processing.

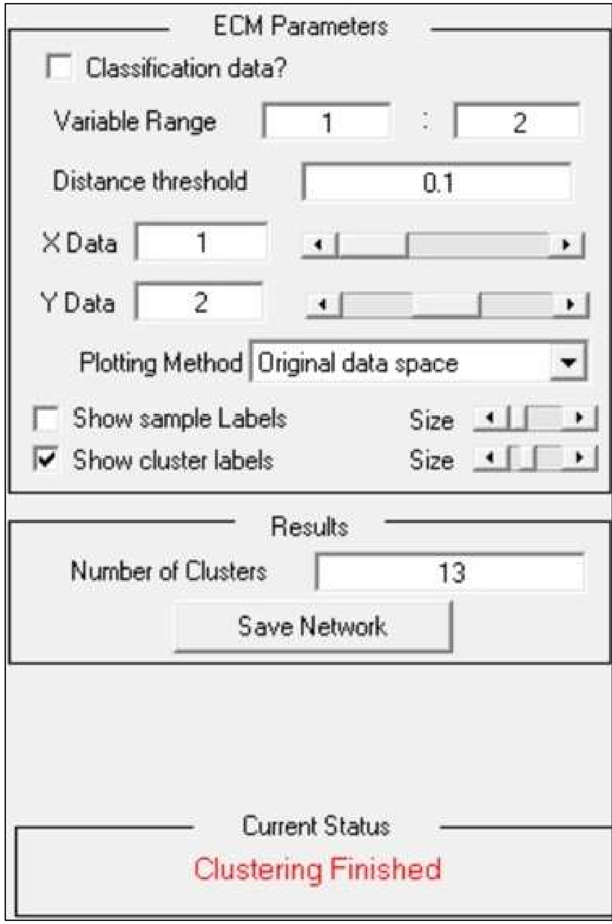


5- The Architecture of DENFIS - *Fuzzy rules are at the heart of the DENFIS model* (cont'd):

Layer 2: **Fuzzy Layer**

- Fuzzy logic principles are applied to the input data, transforming quantitative data points (crisp numerical values) into qualitative (fuzzy values).
- This layer clusters input vectors and determines the clusters' centres.
- Cluster centre and the distance threshold determine the three parameters {a, b, c} for the calculation of membership degrees.
- Membership functions defined within this layer assign degrees of membership to each input value, categorizing them into fuzzy sets like "Low", "Medium", or "High" -> mimics human reasoning to deal with imprecise data.
- The triangular membership function is given below, where **x** is the input to be fuzzified, and **a**, **b**, and **c** are its parameters,

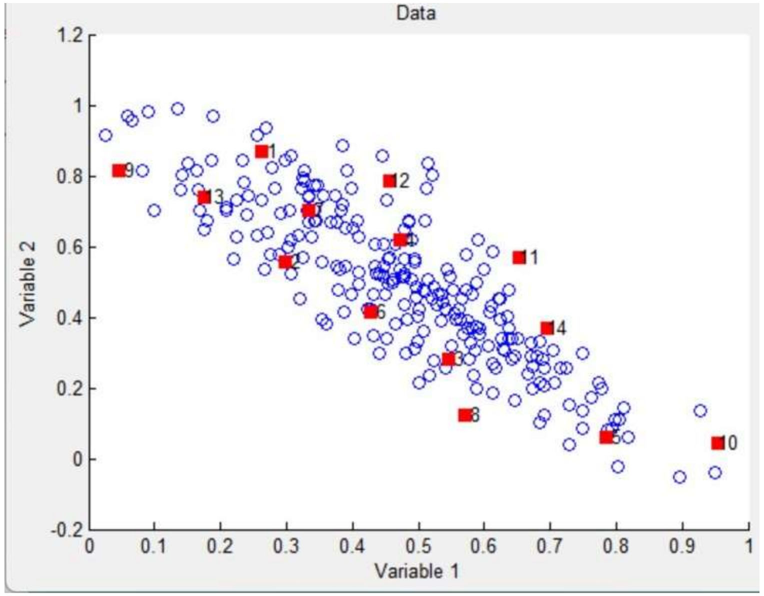
$$\mu(x) = \mathrm{mf}(x; a, b, c) = \begin{Bmatrix} 0 & \quad if\ x \le a \\ \frac{x-a}{b-a} & \quad if\ a < x \le b \\ \frac{c-x}{c-b} & \quad if\ b < x < c \\ 0 & \quad if\ x \ge c \end{Bmatrix}$$

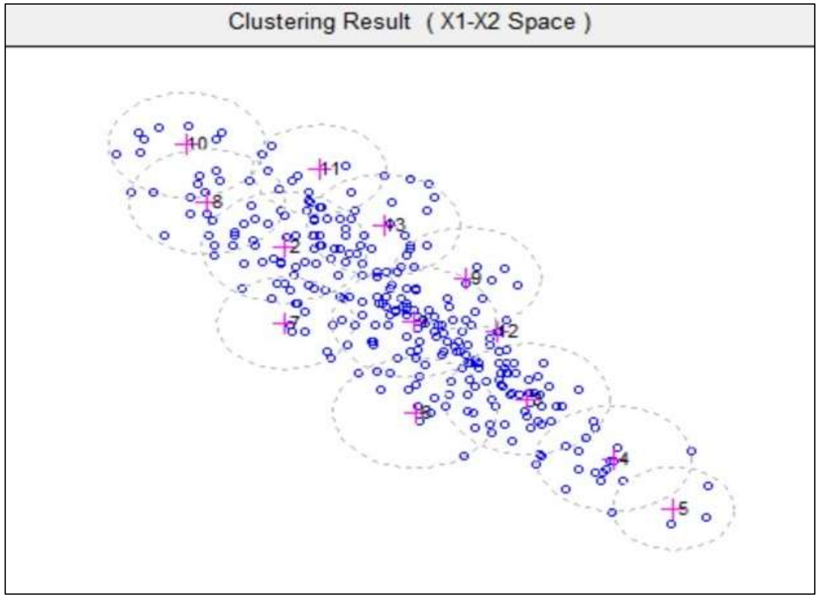


5- The Architecture of DENFIS - *Fuzzy rules are at the heart of the DENFIS model* (cont'd):

Layer 2: **Fuzzy Layer**

- **b** is the centre for cluster along x dimension,
- $\mathbf{a} = \mathbf{b} - \mathbf{d} * \mathbf{d}_{thr}$ & $\mathbf{c} = \mathbf{b} + \mathbf{d} * \mathbf{d}_{thr}$
- **d** varies from -0.2 to 1.2.
- $\mathbf{d}_{thr}$ is the threshold value of the clustering parameter set at 0.1.





5- The Architecture of DENFIS - *Fuzzy rules are at the heart of the DENFIS model* (cont'd):

Layer 3: **Rule Layer**

- This is the core logic of the DENFIS model, with evolving rules that govern the predictions as new data is introduced.
- This layer calculates the product (**Π**) of the membership degrees of input variables in the previous layer.
- Each neuron represents a fuzzy rule. For example, "*if temperature is high and humidity is low, then risk is high*."
- As new data arrives, the system either adjusts existing rules or creates new ones, ensuring the model's adaptability.
- The connection weights between the fuzzy layer and the rule layer represent the **antecedent** part of the fuzzy rules, while the connection weights between the rule layer and the output layer represent the **consequent** part.

Layer 4: Output Layer

- In DENFIS, the first-order Takagi Sugeno type inference systems are used. This layer calculates each rule output as a linear combination of input variables weighted by coefficients defined in the rule's consequent part:
  $\mathbf{y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n}$, where β's are the rule consequent and calculated using Least Square Estimator (LSE).

Layer 5: Regression Output

- After calculating the individual outputs, they are aggregated to form a single fuzzy output.
- This layer calculates the final output, which is the weighted sum of each rule's output.

6- Application of DENFIS on a benchmark dataset:

a. Dataset:

- A gas furnace dataset with time-sequenced records of methane ($CH4$) at t-4 and carbon dioxide ($CO2$) at t-1.
- DENFIS system will be modelled to predict $CO2$ emissions at time t.

b. Data pre-processing:

- Data is uploaded and normalised to ensure that the inputs contribute equally to the model's learning process.
- The normalized dataset is split into training (20%) and testing (80%) sets.

c. Fuzzification:

- We define 3 membership functions (MF) for both input features to categorize the data into:
  Low: Significantly below the average level,
  Medium: Around the average level, and
  High: Significantly above the average level.
- A threshold ($D_{thr} = 0.1$) is set to determine the boundaries of these fuzzy sets and fine-tune the overlap between these membership functions.
- Convert both input features, $CH4_{t-1}$ and $CO2_{t-1}$, into fuzzy values based on the defined 3 MFs.

6- Application of DENFIS on a benchmark dataset (cont'd):

d. **Create the Rules** to reflect the system's dynamics:

Rule 1: If $CH4_{t-1}$ is Low and $CO2_{t-1}$ is Low, then $CO2_t$ is calculated with a specific set of coefficients.
Rule 2: If $CH4_{t-1}$ is Medium and $CO2_{t-1}$ is Medium, then $CO2_t$ is calculated with another set of coefficients.
…
Rule n: If $CH4_{t-1}$ is High and $CO2_{t-1}$ is High, then $CO2_t$ is calculated with a different set of coefficients.

e. **Rules' equations**: Each rule will have a linear equation associated with it for the output: $CO2_t = \mathbf{a_r} * CH4_{t-4} + \mathbf{b_r} * CO2_{t-1} + \mathbf{c_r}$ *where r is the rule no.*

f. **Partition and segment** the dataset based on generated fuzzy rules (13 rules).

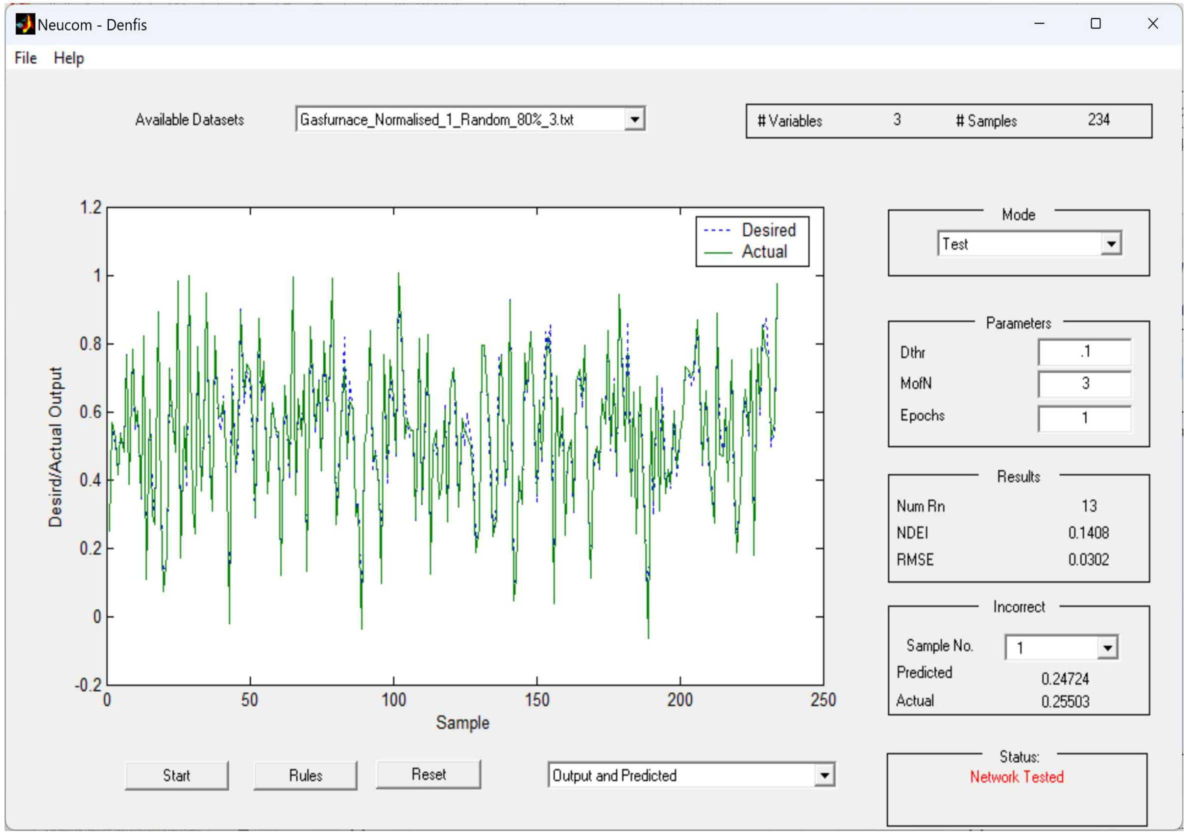
g. **Calculate the Coefficients:**
- Apply linear regression to each cluster to calculate its rule coefficients.
- For each rule, the linear regression on the training data provides different coefficients.
- Linear regression minimizes the sum of squared differences between the predicted and actual output values, providing the best-fit line in each cluster.

	$CH4_{t-4}$	$CO2_{t-1}$	a	b	c
Rule 01	[0.50 , 0.14]	[0.50 , 0.89]	-0.39	0.60	1.43
Rule 02	[0.50 , 0.75]	[0.50 , 0.30]	-0.60	0.60	1.54
Rule 03	[0.50 , 0.70]	[0.50 , 0.63]	-0.40	0.69	1.37
Rule 04	[0.50 , 0.43]	[0.50 , 0.73]	-0.38	0.62	1.41
Rule 05	[0.50 , 0.76]	[0.50 , 0.48]	-0.50	0.67	1.44
Rule 06	[0.50 , 0.95]	[0.50 , 0.05]	-0.55	0.56	1.52
Rule 07	[0.50 , 0.63]	[0.50 , 0.39]	-0.55	0.58	1.51
Rule 08	[0.50 , 0.60]	[0.50 , 0.21]	-0.58	0.60	1.53
Rule 09	[0.50 , 0.27]	[0.50 , 0.95]	-0.39	0.60	1.43
Rule 10	[0.50 , 0.30]	[0.50 , 0.64]	-0.47	0.54	1.50
Rule 11	[0.50 , 0.61]	[0.50 , 0.83]	-0.54	0.67	1.36
Rule 12	[0.50 , 0.84]	[0.50 , 0.19]	-0.55	0.56	1.52
Rule 13	[0.50 , 0.48]	[0.50 , 0.42]	-0.51	0.52	1.51

6- Application of DENFIS on a benchmark dataset (cont'd):

h. **Predicted output**:
$CO2_t = -0.49 * CH4_{t-4} + 0.60 * CO2_{t-1} + 1.47$

i. **Model's accuracy**: The predictive accuracy is evaluated by using the Root Mean Square Error (RMSE) which is obtained at 3%.



7- Further Applications of DENFIS in clinical settings to enhance diagnostic accuracy.

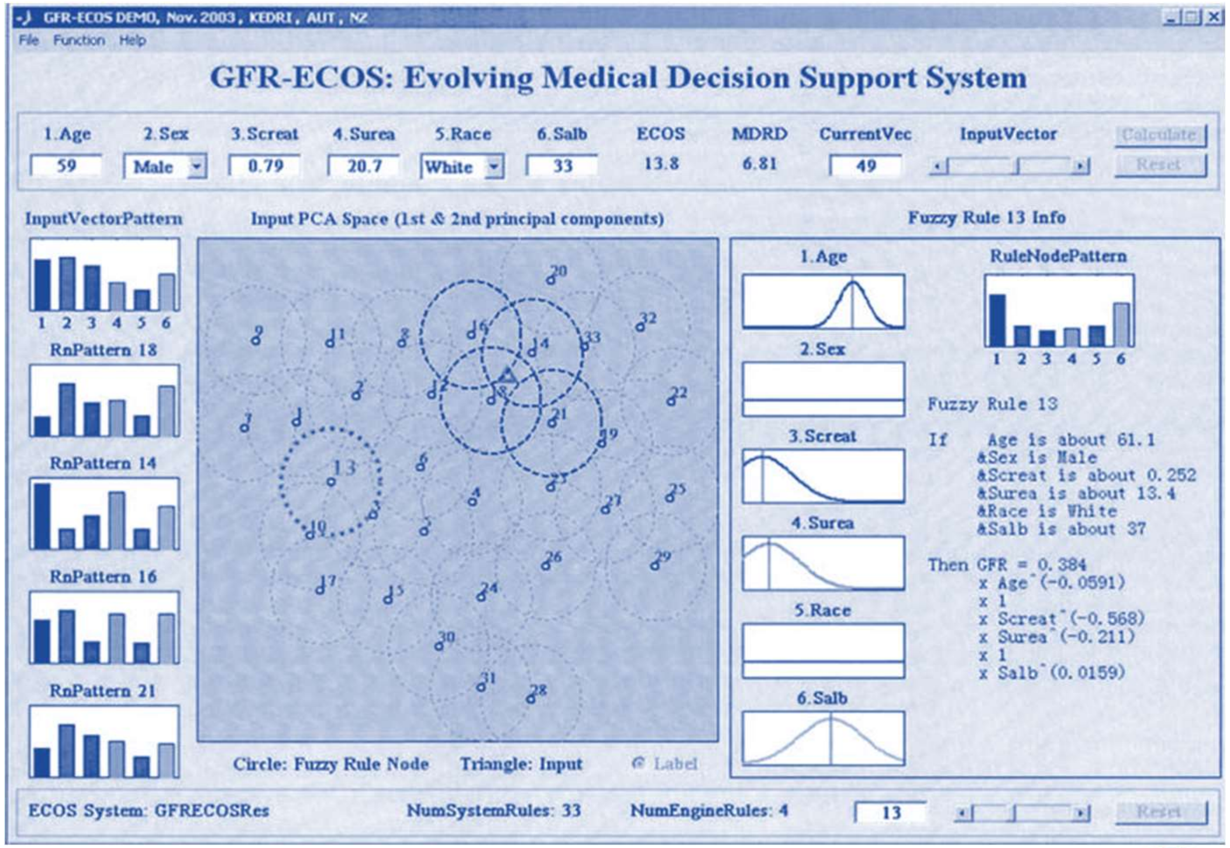
a. Utilizing the ECOS-DENFIS model in clinical settings for more accurate and adaptive predictions of kidney function.

b. In this example, six input variables are used to train a DENFIS system for medical decision support on a renal function GFR.

c. The evolved hidden nodes represent clusters of input data.

d. The data in each of these clusters is approximated by a regression function.

e. Local fuzzy rule 13 is extracted that represents/approximates the data in cluster 13 using the shown membership functions.



8- The Advantages of DENFIS:

- DENFIS is notable for its ability to adapt and excel in both online and offline learning environments.
- DENFIS demonstrates exceptional accuracy when dealing with complex nonlinear time-series data.
- DENFIS outperforms traditional models in its ability to dynamically learn sequences across time.
- DENFIS continuously updates its knowledge base, making it ideal for real-time analysis and decision-making scenarios.
- However, DENFIS needs to enhance its ability to analyze and model spatio-temporal data.