# Neuro-Fuzzification

Group Fuzzification: Lyndon, Jaypal, Samyak, Anik, Vishnu

# Introduction

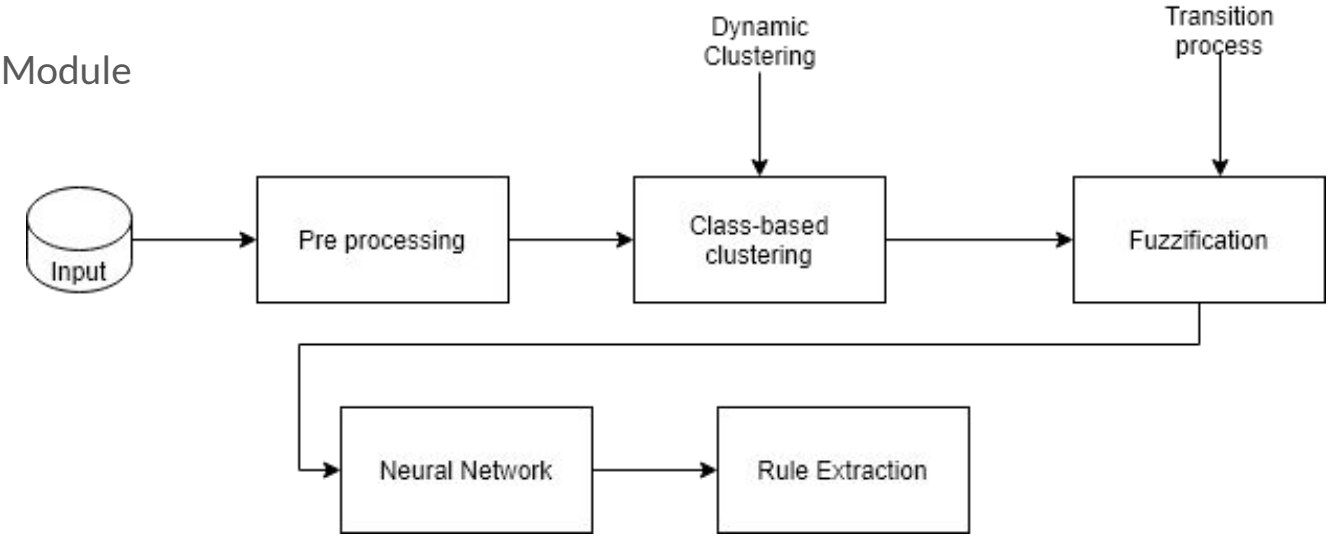
1. The presented problem, is to precisely and accurately classify data in a quick and effective way.
2. We will show, that our solution provides a quick and reusable method of classifying data.
3. Contribution: Providing additional validity to rule based classification.
4. Showed rule extractraction validity for binary classification.

# The Problem

1. Other methods, that accomplish the same goal. Standard Binary Classifiers. Standard Neural Networks. Standard Expectation Maximization.
2. Our approach likely is overly complex in implementation for a binary classification.
3. We are trying to improve on unsupervised learning accuracy as well as create more competition for supervised learning methods.

# Our Idea

1. Dynamic Clustering Module
2. Fuzzification Module
3. Neural Network Module
4. Create a Rule Extraction Module
5. Run Them All Together
6. ...?
7. Profit



# Dynamic Clustering

- Dynamic clustering is a form of incremental machine learning.
- The cluster formation does not require a predefined value.
- It embraces different scenarios: dynamic features, dynamic clusters.

# Dynamic Clustering

- We took the data,sort it and formed clusters for every feature.
- Centroid calculation for each cluster.
- Threshold was calculated for each feature.
- Appending of clusters on basis on threshold.

  $d(c_j,c_k)=|c_j-c_k|$

  $J=\min\{d(c_j,c_k)\}$; $k=1,\ldots,(N-1)$

$$F_{th} = \frac{\sum_{j=1}^{N} M_j}{N}.$$

Threshold

- Mean and standard deviation was calculated.

# Example of Cluster Formation

step :1

| $x_i$ | Class | |
|---|---|---|
| 4.40 | 1 | Cluster 1 |
| 4.75 | 1 | |
| 4.83 | 1 | |
| 4.98 | 2 | Cluster 2 |
| 5.01 | 1 | Cluster 3 |
| 5.06 | 2 | Cluster 4 |
| 5.13 | 2 | |
| 5.18 | 2 | |
| 5.24 | 3 | Cluster 5 |
| 5.29 | 2 | Cluster 6 |
| 5.32 | 2 | |
| 5.41 | 3 | Cluster 7 |
| 5.43 | 3 | |
| 5.53 | 3 | |
| 5.60 | 3 | |

step :2

| Group | Centroid | *n* member |
|---|---|---|
| 1 | 4.66 | 3 |
| 2 | 4.98 | 1 |
| 3 | 5.01 | 1 |
| 4 | 5.12 | 3 |
| 5 | 5.24 | 1 |
| 6 | 5.31 | 2 |
| 7 | 5.49 | 4 |
| *threshold* | | 2.125 |

step :3

| Group | Centroid | *n* member |
|---|---|---|
| 1 | 4.66 | 3 |
| 2,3,4 | 5.07 | 5 |
| 5,6 | 5.28 | 3 |
| 7 | 5.49 | 3 |

step :4

| Cluster | $c$ | $\sigma$ |
|---|---|---|
| 1 | 4.66 | 0.186 |
| 2,3,4 | 5.07 | 0.074 |
| 5,6 | 5.28 | 0.032 |
| 7 | 5.49 | 0.076 |

# Fuzzification

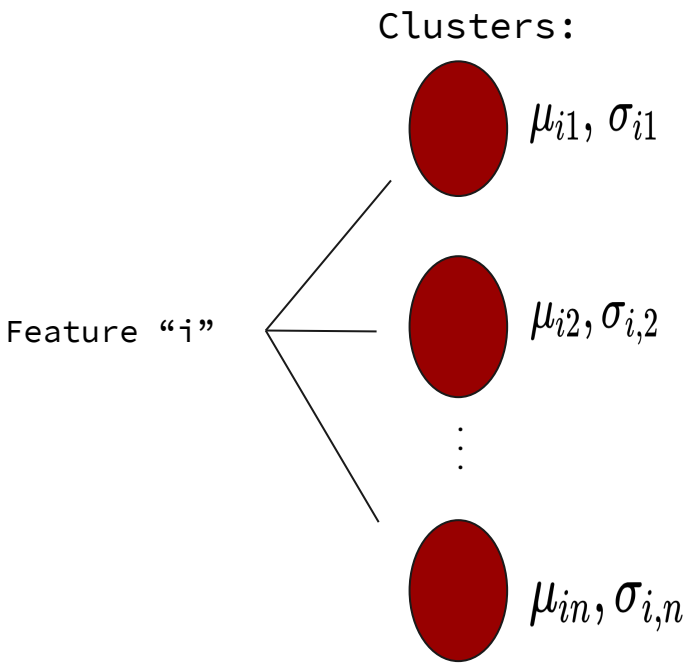
We take the inputs from the Dynamic clustering part.

We create membership function.

Membership function identifies the probability of each datapoint being member of a cluster based on mean and standard deviation of the cluster.

It then creates a binary output that assigns the datapoint to the cluster with highest membership value.

# Fuzzification




Membership function:

$$m_{ij}(x) = \begin{cases} 0, & if\ \sigma_{ij} = 0\ and\ x \neq \mu_{ij} \\ e^{-\left((x-\mu_{ij})^2/2\sigma_{ij}^2\right)}, & if\ \sigma_{ij} \neq 0 \\ 1, & if\ \sigma_{ij} = 0\ and\ x = \mu_{ij} \end{cases}$$

Binary output: This is fed into the Neural network

$$b_{ij} = \begin{cases} 1, & \text{if } m_{ij} \text{ is maximum among all } 1 < j < n \\ 0, & otherewise \end{cases}$$

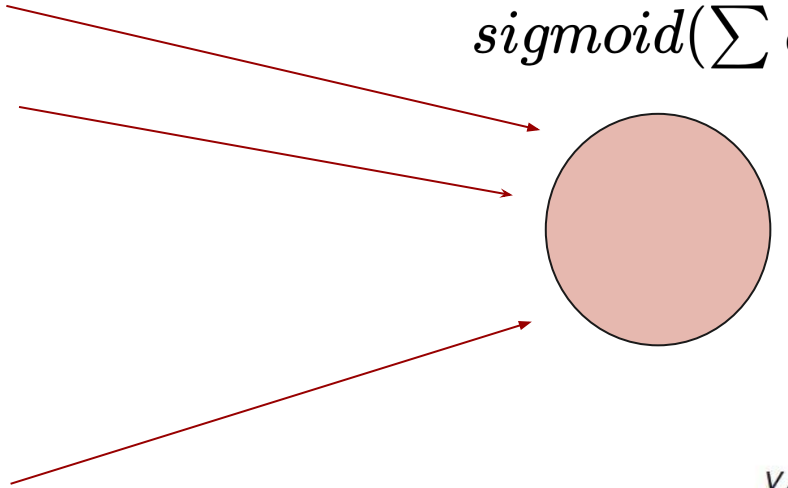
# Neural Network

Binary Input:

$$[a_1]$$
$$[a_2]$$
$$\cdot$$
$$\cdot$$
$$\cdot$$
$$[a_n]$$

Output: With Sigmoid activation

$$sigmoid(\sum a_i w_i)$$



$$[x \,|\, 0 \leq x \leq 1]$$

Sigmoid Definition

$$y_o = \frac{1}{(1 + e^{-V_o})}$$

$$V_o = \sum_{i=1}^{n} \sum_{j=1}^{k} b_{ij} w_{ijo}$$

Weights:

$$[w_1, w_2, \cdots, w_n]$$

# Neural Network

Training the network:

Output vs Expected Error

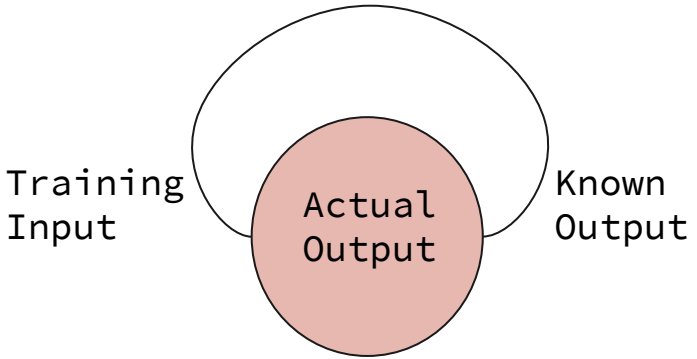
$$e_o(l) = d_o(l) - y_o(l)$$

Error * Sigmoid Derivative of Dot Product

$$\delta_o(l) = e_o(l)\varphi'(V_o(l))$$

Weighted Adjustment with Input

$$w_{ijo}(l+1) = w_{ijo}(l) - \eta\delta_o(l)b_{ij}(l)$$



# Neural Network - Rule Extraction

Post Training, Weights Can Be Removed/Sorted

Weights:

$$[w_1, w_2, \cdots, w_n]$$



Sorted Weights:

$$[w_{s_1}, w_{s_2}, \cdots, w_{s_n}]$$

Once extracted, it becomes possible to select the ones of importance in the rule formation methods.

# Rule Extraction

The top ‘n’ features are selected based on the training of weights from the neural network and then a easy classification rule is formed.

A simple ‘OR’ rule is created as follows;

*If* ***f1*** *is* ***lv 6*** *‘or’* ***f5*** *is* ***lv 2*** *‘or’* ***f6*** *is* ***lv 9****:*

*then it is in class ‘a’*

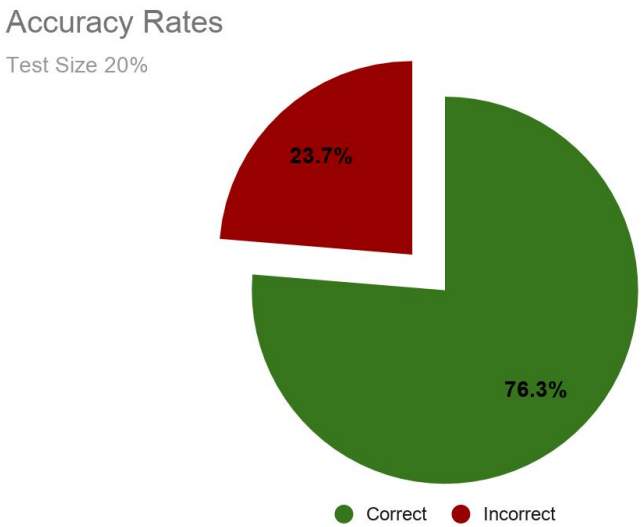
# Breast Cancer Detection Results

Picked Top 8 Features for Rule Formation

Compared Computed Value

V.S. Expected Value

Trained on 30 Features



# Conclusion

We have successfully implemented the algorithm to come up with a rule based classification.  Achieved satisfactory accuracy.

Have future plans in this current work to test and gather more information from other datasets.

And for a goal in a future work, we would implement a complex rule extraction method.

# References And Related Works


1. Phichit Napook and Narissara Eiamkanitcaht. 2015. Forecasting algorithm developed from the neuro: fuzzy system. In Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers(UbiComp/ISWC'15 Adjunct). Association for Computing Machinery, New York, NY, USA, 1189–1196. DOI:https://doi.org/10.1145/2800835.2800983
2. N. Eiamkanitchat, N. Theera-Umpon and S. Auephanwiriyakul, "A novel neuro-fuzzy method for linguistic feature selection and rule-based classification," *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, 2010, pp. 247-252, doi: 10.1109/ICCAE.2010.5451487.
3. P. Wongchomphu and N. Eiamkanitchat, "Enhance Neuro-fuzzy system for classification using dynamic clustering," *The 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (JICTEE)*, 2014, pp. 1-6, doi: 10.1109/JICTEE.2014.6804071.
4. Heisnam Rohen Singh, Saroj Kr. Biswas, Biswajit Purkayastha, A neuro-fuzzy classification technique using dynamic clustering and GSS rule generation, Journal of Computational and Applied Mathematics, Volume 309, 2017, Pages 683-694, ISSN 0377-0427, https://doi.org/10.1016/j.cam.2016.04.023.

# Questions?