Lecture 8. Integrated Learning Systems

I. AbouHassan and N. Kasabov (2024)
"**NeuDen**: A Framework for the Integration of
Neuromorphic Evolving Spiking Neural Networks with
Dynamic Evolving Neuro-Fuzzy Systems for
Predictive and Explainable Modelling of Streaming Data,"
Preprint, LicenseCC BY 4.0,
doi:10.21203/rs.3.rs-4119202/v1.

Cognitive systems (CogSys) are software-hardware systems that have their structure and functionality based on principles of information processing in the human brain. They are part of AI, but AI includes also other systems that manifest cognitive behaviour, such as speech and image recognition, learning and reasoning, but using other methods, such as statistical, empirical, abstract logic, etc.

The course is by research papers.
Every topic will include:

1. Topic/task/problem specification
2. Previously published methods for solving the problem
3. Description of the method and in the paper under discussion
4. Experimental results and discoveries
5. Future work to be done for this problem
6. Questions for individual work for those interested

1- Topic/task/problem specification:

Explainability concerns in eSNN: Explaining the relationships between input features made by evolving spiking neural networks (eSNN) is often challenging.

Limitations of deNFS modelling: Dynamic evolving neuro-fuzzy systems (deNFS) struggle to capture the temporal interactions among multiple streaming time series.

Balancing explainability and accuracy: It is a significant challenge to develop a model capable of enhancing explainability while maintaining regression accuracy.

2- Previously published methods for solving the problem:

Evolving Connectionist Systems (ECOS): These systems incrementally learn and evolve from incoming data and have been effective in various applications, outperforming other ANN approaches in rapid processing capabilities.

Spike-Time-Dependent Plasticity (STDP) Learning in SNN: Utilizes the temporal interaction between multiple time series connected to a dynamic eSNN (deSNN) to learn and make predictions.

Dynamic Evolving Neuro-Fuzzy System (deNFS): This system utilizes vector samples of data to predict future time series values and produce interpretable fuzzy rules.

Hybrid Models: Previous models have integrated SNN with classical artificial neural networks (ANNs) for tasks such as event-based optical flow estimation or combined SNN with Echo State Network (ESN) mechanisms for real-time learning and classification of EEG data, although these models had their own limitations.

2- Previously published methods for solving the problem (cont'd):

Hybrid SNN-ANN Architecture: This model is proposed for event-based optical flow estimation, integrating the strengths of both SNNs and ANNs. However, it didn't address the vanishing gradient problem and added complexity due to spiking layers.

Hybrid SNN and Echo State Network: This model is proposed for real-time learning and classification of EEG data. The model's complexity requires careful hyperparameter tuning, which could cause optimization delays and issues with explainability.

Hybrid Deep + Non-deep Learning Model: This model is aimed at improving the classification of low-volume high-dimensional data. However, it is dependent on more data for training and the selection of feature extraction layers, leading to inconsistent performance across different datasets.

Hybrid Model for Building Energy Optimization: This model integrates discrete and continuous data but has limited applicability across different machine learning algorithms, requiring broader validation.

Hybrid Graphical Models and Neural Networks: These models aim to combine the inference accuracy of graphical models with the dynamic learning capabilities of deep neural networks, suggesting a future convergence of these models for efficient and accurate complex data analysis.

3- The Proposed Method – NEUDEN:

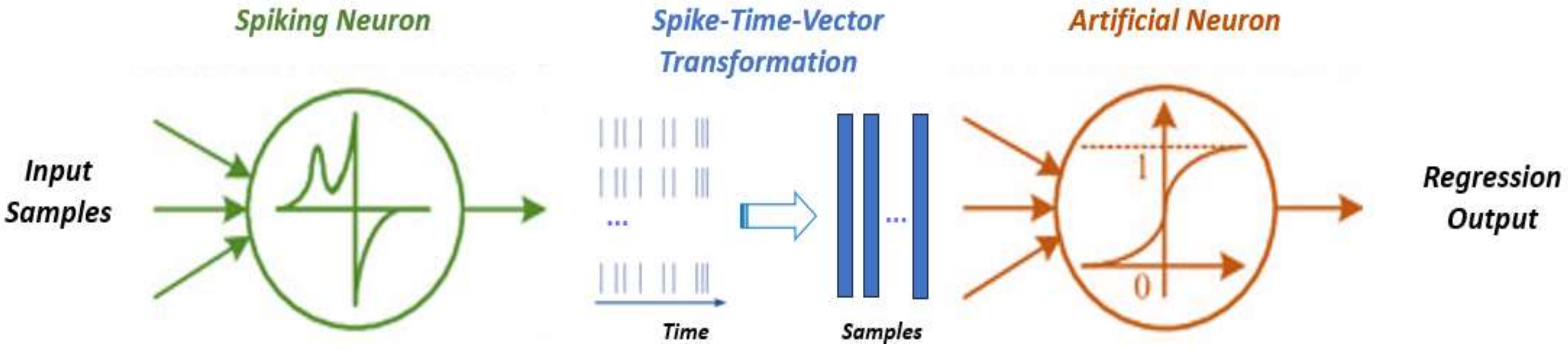
and

we offer

NeuDen

a novel framework
integrating two distinct
computational paradigms

3- The Proposed Method – NEUDEN (cont'd):



'**NeuDen**' is a novel framework integrating two distinct computational paradigms –

- **eSNN:** Neuromorphic evolving spiking neural networks that efficiently learn multiple time series in their temporal association and interaction.
- **deNFS**: Dynamic evolving neuro-fuzzy systems that incrementally learn from eSNN feature vectors to predict future time-series values and produce interpretable fuzzy rules.

3- The Proposed Method – NEUDEN (cont'd):

A. **NeuDen Framework Implementation:**
   - Combining Models: The 'NeuDen' framework aims to maximize the benefits of both eSNN and deNFS models
   - Spatio-Temporal Model: STDP learning is used in SNN to capture temporal interactions connected to a dynamic eSNN (deSNN) as a regressor/classifier.
   - Feature Extraction: Feature vectors are extracted from the trained deSNN.
   - Neuro-Fuzzy Learning: Dynamic fuzzy inference and rule extraction occur in a deNFS, resulting in accurate predictions

B. **Advantages of NeuDen:**
   - Explainability Enhancement: NeuDen overcomes the explainability issues of eSNN.
   - Improved Modeling: It addresses the limitations of deNFS in modelling multiple streaming time series interactions.

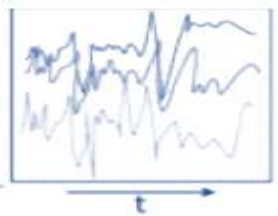
C. **Performance Comparison:**
   - NeuDen surpasses both deSNN and deNFS by providing multiple regression models and achieving higher accuracy.

D. **Demonstrated Performance:**
   - NeuDen's effectiveness is demonstrated on benchmark and financial/economic time series data, showing significantly smaller RMSE (**3 to 100 times smaller**) when compared with other evolving systems.

8

3- The Proposed Method – NEUDEN (cont'd): Data Flow

NeuDen combines spiking NN with neuro-fuzzy systems to process streaming data efficiently.



Input Time Series (N variables) → Spikes Encoding → Spatio-Temporal Initialization of a SNN

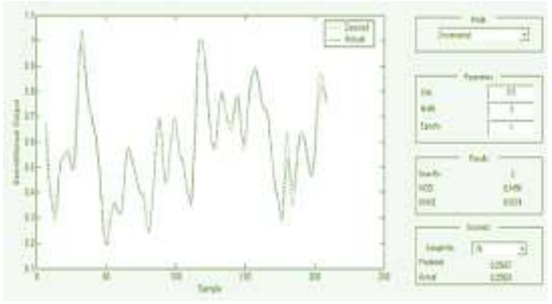
↓

Setting Hyper Parameters → Unsupervised Learning in SNN (STDP) → Supervised Learning in SNN (deSNN)

↓

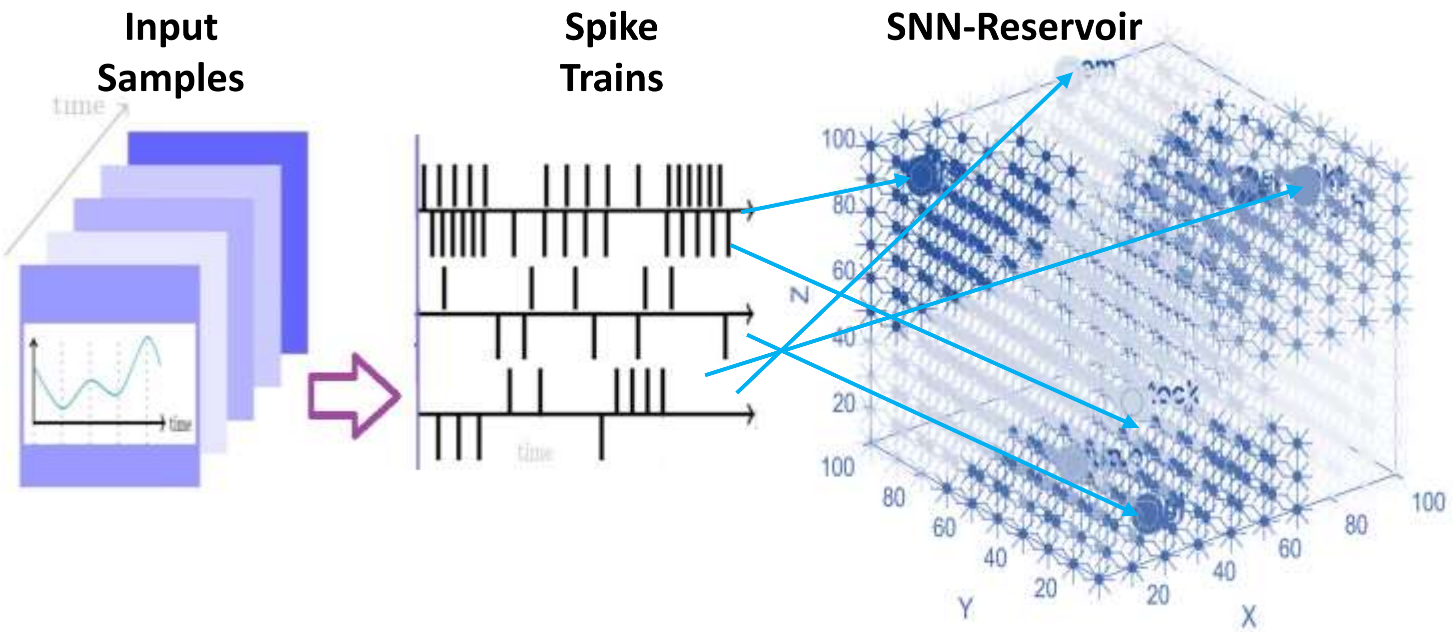
N-element Feature-Vector Extraction → Learning and Fuzzy Inference in a DNFS → Regression Output & Extracting rules →

Rule:
if X1 is GMF(c1, r1)
and X2 is GMF (c2, r2)
then Y = a * X1 + b * X2 + ε



3- The Proposed Method – NEUDEN (cont'd):

NeuDen Module 1
Spatio-Temporal Data Processing

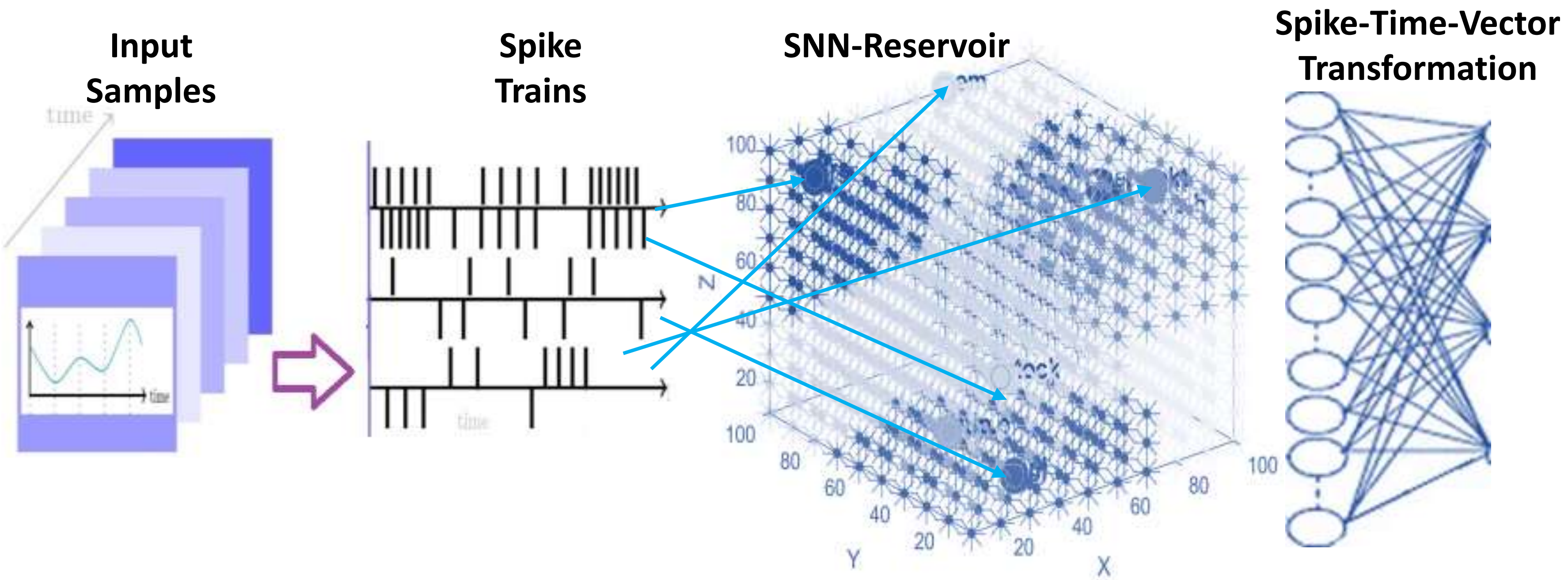


- NeuDen employs a spike encoding algorithm to encode continuous time series data into sequences of spikes.
- NeuDen employs the spike-time-dependent plasticity (STDP) learning method in the unsupervised stage in its reservoir to adjust the connections between neurons based on the timing of their spikes.
- NeuDen employs (DeSNN) learning in the supervised stage to further adjust the connections between neurons based on the timing of their spikes.

10

3- The Proposed Method – NEUDEN (cont'd):

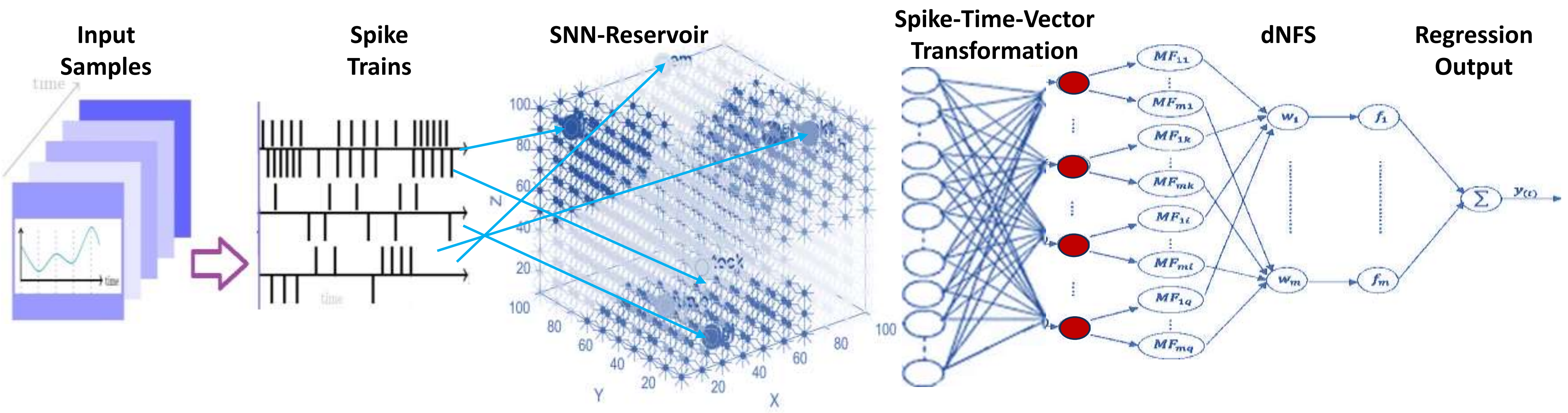
NeuDen Module 2
Feature Vector Extraction



- NeuDen generates feature vectors after the SNN learns the temporal patterns.
- Feature vectors are structured representations of the data that summarize the information captured by module 1.

3- The Proposed Method – NEUDEN (cont’d):

NeuDen Module 3
Dynamic Evolving Neuro-Fuzzy System



- NeuDen employs dynamic evolving neuro-fuzzy learning on the feature vectors to produce a fuzzified outcome.
- NeuDen adopts evolving clustering methods and fuzzy inference mechanisms to model complex data sequences.
- NeuDen generates and extracts fuzzy rules.

4- A NeuDen Model for Time Series Regression using a Multiple Stock Benchmark Dataset:

- Stock indices (spatial features): Apple Inc., Google, Intel Corp, Microsoft, Yahoo, and NASDAQ.
- Original dataset (temporal features): 150 daily observations for 6 variables.
- Sample generation: 50 Samples, each of which contained 100 timed sequences of daily closing prices.
- New dataset = 30,000 data point (5,000 observations for 6 variables)
- The target values representing the closing price of ***NASDAQ*** at the next day are arranged in a column in the target file.

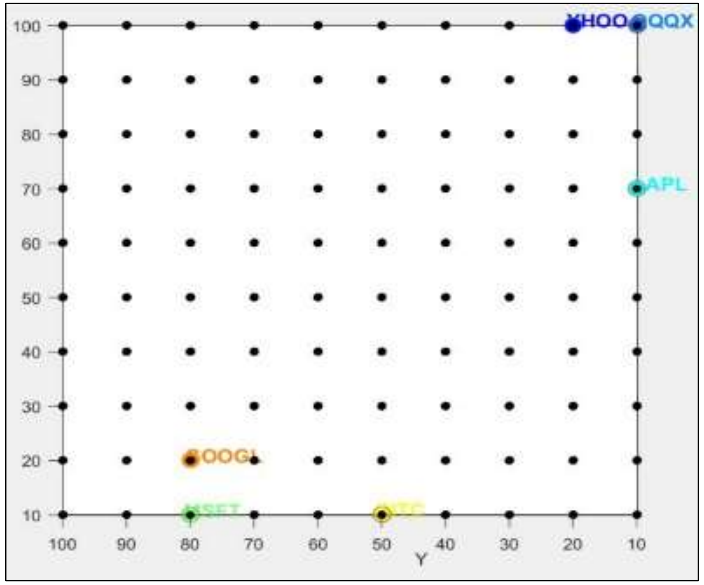
4- A NeuDen Model for TS Regression using a Multiple Stock Benchmark Dataset (cont'd):

- The generated samples are transformed into trains of spikes (Th = 0.5) with a split of 50/50 for training/validating processes.
- Assign 1000 spiking neurons using a graph-matching algorithm to allocate input neurons based on their time series similarity.
- Regulating the initial connection weights between neighbouring neurons in the reservoir with small-world connectivity of radius 2.5.
- Setting the unsupervised learning parameters: Spike-Time Dependent Plasticity (STDP) = 0.001, leak rate = 0.002, firing threshold = 0.5, refractory period = 6, and 3 training iterations.
- Setting the supervised learning parameters: Modulation factor = 0.9, and drift parameter = 0.01.
- Feature-vectors Extraction.
- The fuzzy inference learning stage uses 3 membership functions, 2 learning rounds, & threshold of 0.1.
- The achieved RMSE on 50% of the data selected for testing is 0.02.

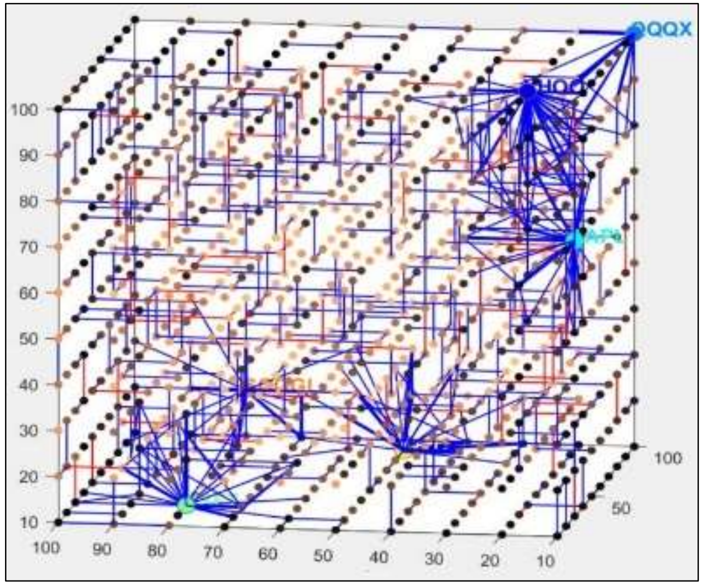
14

4- A NeuDen Model for TS Regression using a Multiple Stock Benchmark Dataset (cont'd):

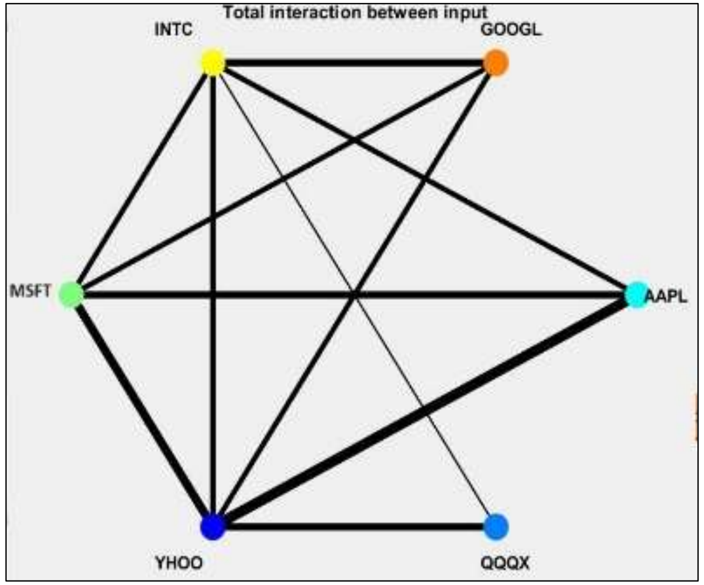
Mapping and neuronal connections in the reservoir



Mapping of input features in a 2D view.



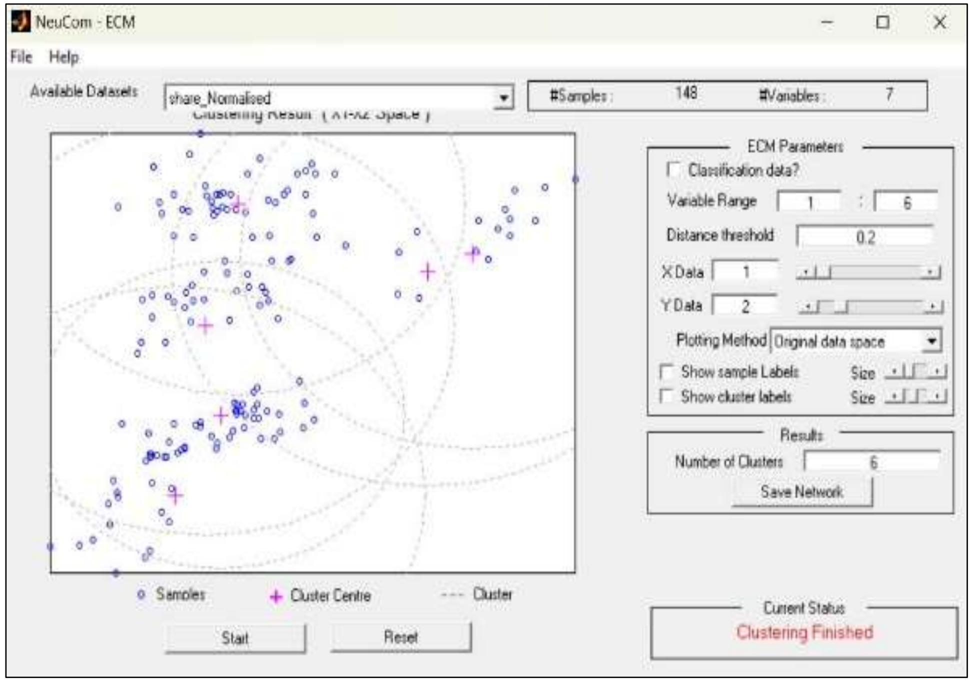
Synaptic connections after unsupervised training, blue lines indicate positive weight and red - negative weight.



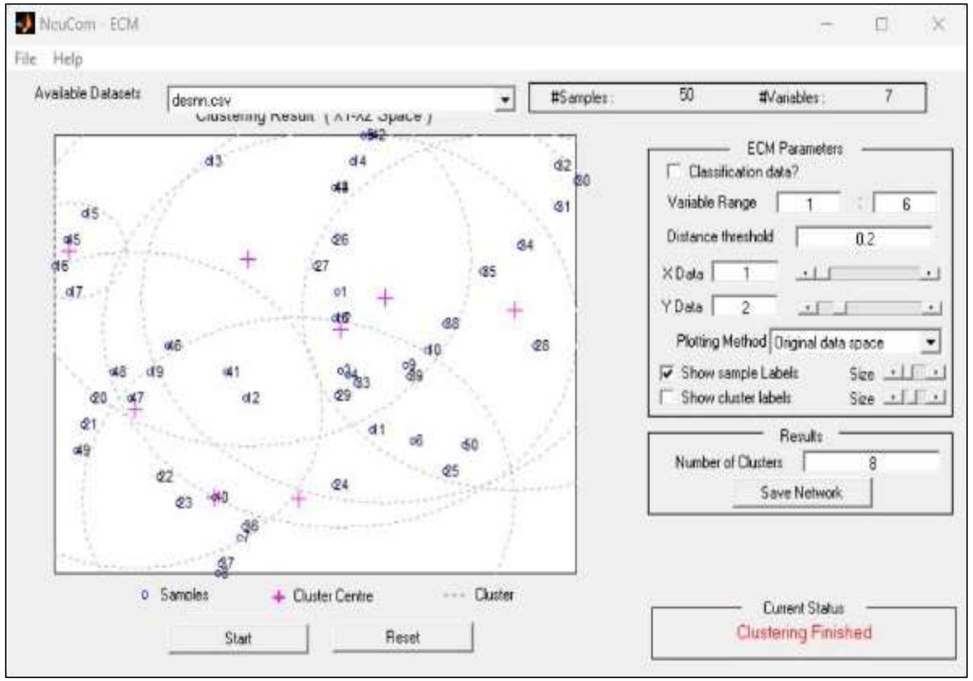
Feature Interaction Network with neuron connection weights, is level of information exchanged between spiked input variables.

4- A NeuDen Model for TS Regression using a Multiple Stock Benchmark Dataset (cont'd):

Clustering of Real Data



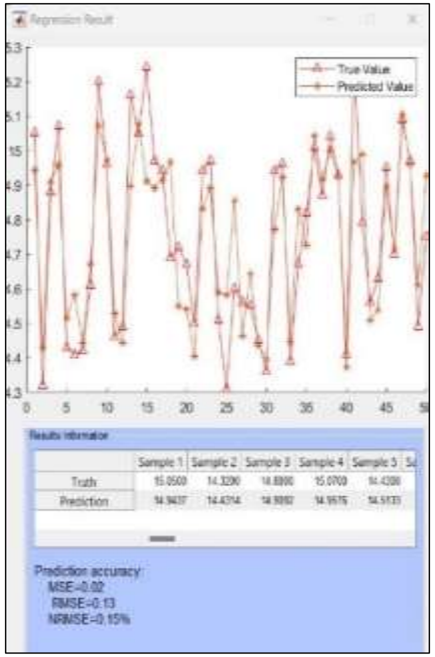
in DENFIS model

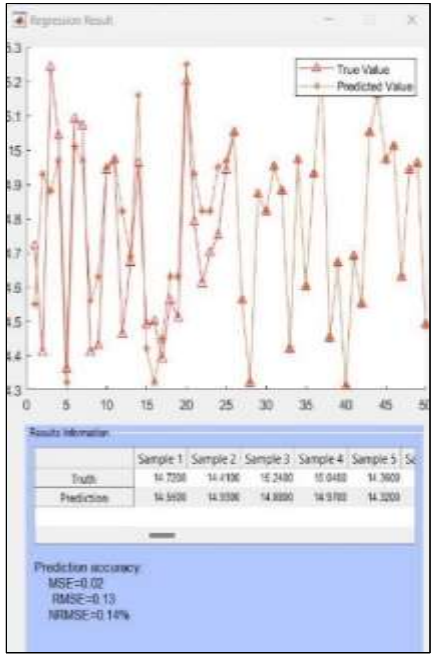


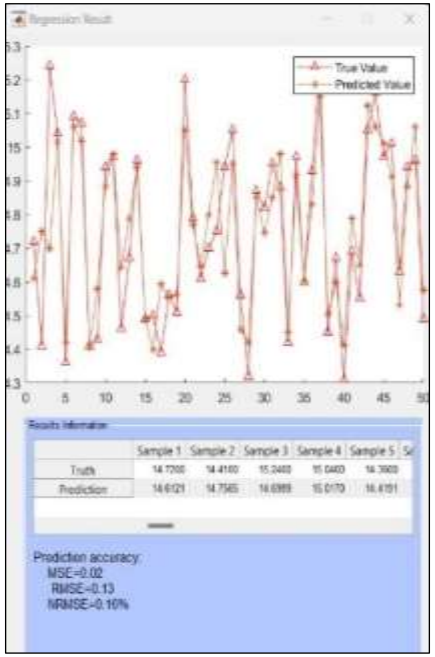
in NeuDen model

4- A NeuDen Model for TS Regression using a Multiple Stock Benchmark Dataset (cont'd):

Stock Price Prediction Modelling Test Results





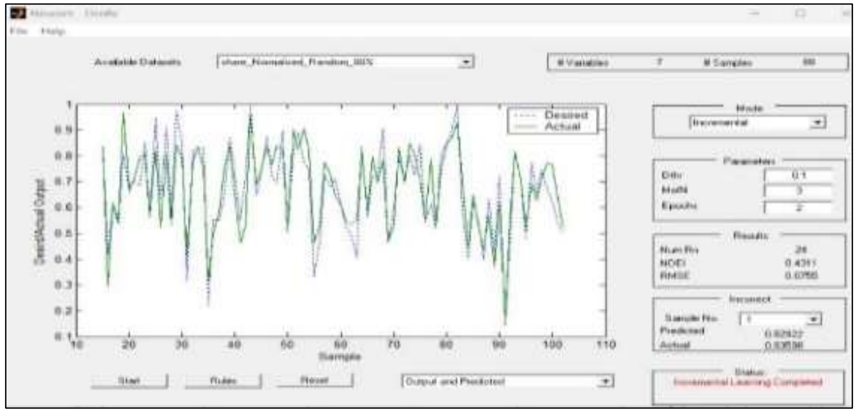


NeuCube after cross-validation optimization of 2 rounds

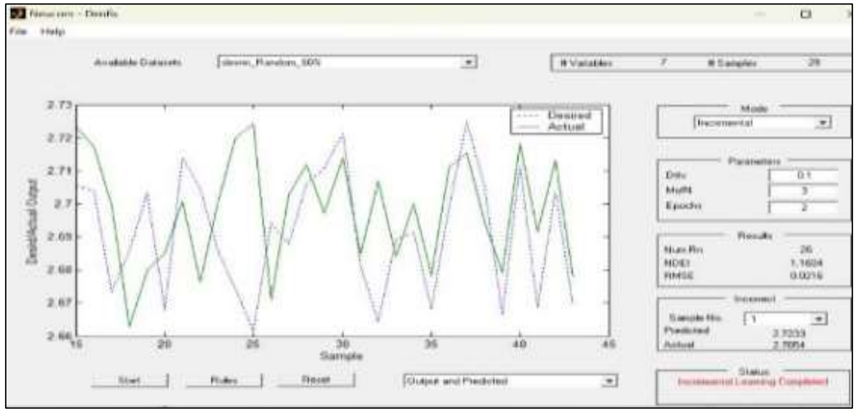
deSNN regressor
RMSE = 0.13

Linear regressor
RMSE = 0.13

SVM regressor
RMSE = 0.13



DENFIS incremental validation RMSE=0.08



NEUDEN incremental validation
RMSE=0.02

4- A NeuDen Model for TS Regression using a Multiple Stock Benchmark Dataset (cont'd):

Comparative Analysis: NeuDen performance vs other regression models

Model + Regressor	No optimization	After Optimization
SNNcube + deSNN	0.17	0.13
SNNcube + Linear	0.20	0.13
SNNcube + SVM	0.17	0.13
DENFIS	0.08	*none*
NeuDen (SNNcube + dNFS)	**0.02**	*none*

4- A NeuDen Model for TS Regression using a Multiple Stock Benchmark Dataset (cont'd):

Fuzzy Rules Extraction and Regression Output

Fuzzy Rules Extraction:

IF

X1 is GaussianMF (0.50, 0.50) &
X2 is GaussianMF (0.49, 0.72) &
X3 is GaussianMF (0.49, 0.81) &
X4 is GaussianMF (0.44, 0.67) &
X5 is GaussianMF (0.50, 0.20) &
X6 is GaussianMF (0.50, 0.57)

THEN

Regression Output:

Y = 1.78 + 0.45*X - 0.92*X2 - 0.53*X3 + 0.49*X4 + 0.06*X5 + 0.15*X6

19

5- Conclusion:

NeuDen

The method's innovation lies in its hybrid approach, combining the temporal processing capabilities of SNNs with the learning and interpretability of fuzzy systems. This not only enhances the predictive accuracy but also provides a level of explainability that is often lacking in complex, black-box models. The framework can adapt and evolve with incoming data, making it suitable for dynamic and streaming data environments.

Integration of eSNN and deNFS: NeuDen combines eSNN's efficiency in handling temporal data with deNFS's incremental learning and knowledge discovery capabilities.

Objectives of NeuDen: NeuDen aims to not only improve predictive accuracy but also significantly enhance model explainability for complex, continuously evolving data.

A bridge between two paradigms: By bridging the gap between eSNN and deNFS, NeuDen provides highly accurate predictions, even in cases of continuous data inflow.

Expanding applications: NeuDen has the potential to revolutionize data-driven decision-making across diverse domains.

20

Thank you
for your attention

21