Advanced Artificial Intelligence Technologies and Applications

Course organiser: A/Prof. Shihua Zhou

Course presenter

Prof Nikola Kasabov
Visiting Professor at Dalian University
Life FIEEE, FRSNZ, FINNS, DVF RAE UK
Founding Director KEDRI
Professor, Auckland University of Technology, NZ
George Moore Chair/Professor, Ulster University, UK
Honorary Professor, University of Auckland NZ , Peking University China
Visiting Professor IICT/Bulgarian Academy of Sciences and Teesside University UK
Doctor Honoris Causa Obuda University Budapest
Director, Knowledge Engineering Consulting Ltd (https://www.knowledgeengineering.ai)

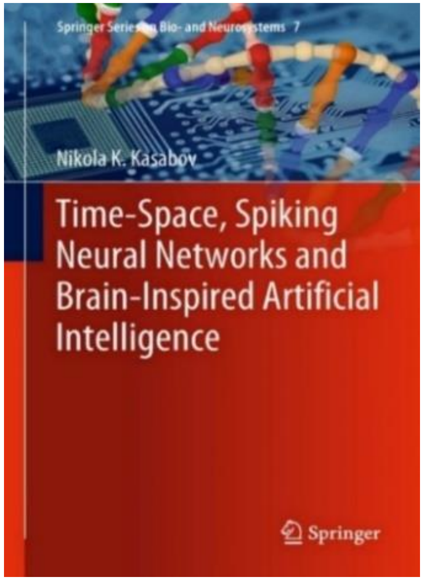
Assistants

A/Prof. Wei Qi Yan
Director of the CeRV Center, AUT
Weiqi.yan@aut.ac.nz
https://academics.aut.ac.nz/weiqi.yan

Ms Iman AbouHassan
iabouhassan@tu-sofia.bg
abouhassan.iman@gmail.com

Advanced Artificial Intelligence Technologies and Applications

1. AI and the evolution of its principles. Evolving processes in Time and Space (Ch1, 3-19)
2. From Data and Information to Knowledge. Fuzzy logic. (Ch1,19-33 + extra reading)
3. Artificial neural networks - fundamentals. (Ch2, 39-48). Computational modelling with NN. NeuCom.
4. Deep neural networks (Ch.2, 48-50 + extra reading).
5. Evolving connectionist systems (ECOS) (Ch2, 50-78). Experiments with NeuCom.
6. Deep learning and deep knowledge representation in the human brain (Ch3)
7. Spiking neural networks (Ch4). Evolving spiking neural networks (Ch5)
8. Brain-inspired SNN. NeuCube. (Ch.6). NeuCube software (IA)
9. Evolutionary and quantum inspired computation (Ch.7)
10. AI applications in health (Ch.8-11)
11. AI applications for computer vision (Ch.12,13)
12. AI for brain-computer interfaces (BCI) (Ch.14)
13. AI for language modelling. ChatBots (extra reading)
14. AI in bioinformatics and neuroinformatics (Ch15,16, 17,18)
15. AI applications for multisensory environmental data (Ch.19)
16. AI in finance and economics (Ch19)
17. Neuromorphic hardware and neurocomputers (Ch20).



Course book: N.Kasabov, Time-Space, Spiking Neural Networks and Brain-Inspired Artificial Intelligence Springer, 2019, https://www.springer.com/gp/book/9783662577134

Additional materials: https://www.knowledgeengineering.ai/china

N. Kasabov Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering, MIT Press, 1996.

ZOOM link for all lectures: https://us05web.zoom.us/j/4658730662?pwd=eFN0eHRCN3o4K0FaZ0lqQmN1UUgydz09

Lecture 4.

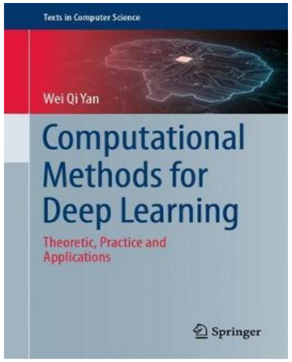
Deep Neural Networks

(Ch2 from the text book, 48-50 + extra reading)

1. Convolutional neural networks (CNN) and deep neural networks (DNN). Advantages and problems.
2. Recurrent NN. Reservoir computing and Liquid State Machines
3. Hybryd systems that combine NN and FS

4. Developing MLP applications in NeuCom (Tutorial, Ms Iman AbouHassan)

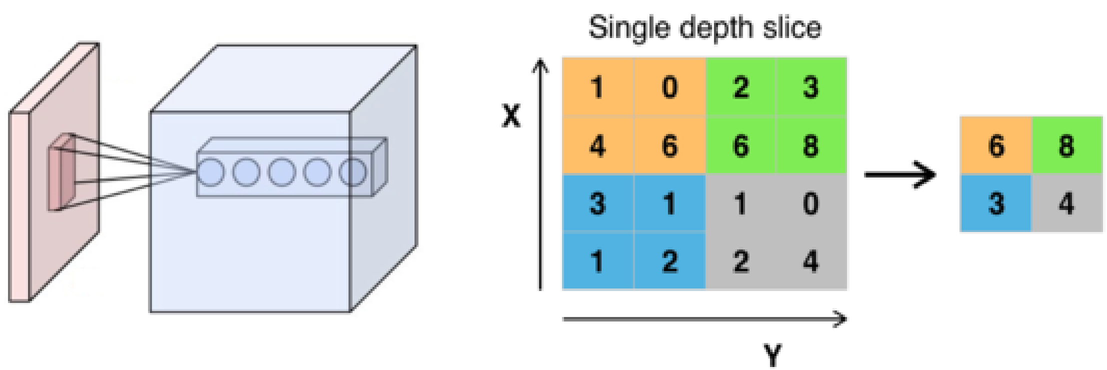
5. Questions for individual work

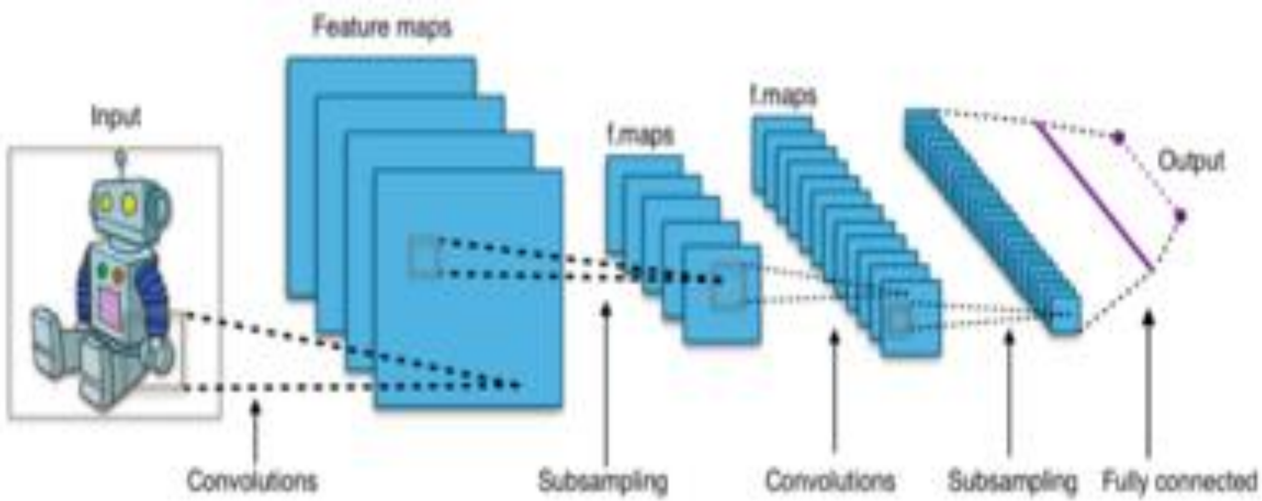


nkasabov@aut.ac.nz https://www.knowledgeengineering.ai

1. Convolutional NN (CNN) and deep NN (DNN)

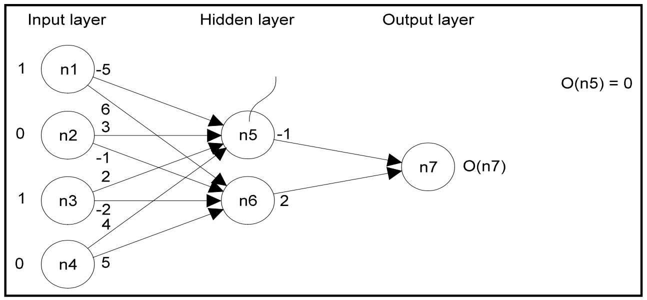
The input data is segmented into segments and a function is allocated to each segment that can be performed in a neuronal unit.





Deep NN (DNN)

- Neural networks that have many layers of neurons and connections.
- They are MLP with more layers:
  - A single hidden layer MLP trained by the BP algorithm



Forward pass:

BF1. Apply an input vector **x** and its corresponding output vector **y** (the desired output).

BF2. Propagate forward the input signals through all the neurons in all the layers and calculate the output signals.

BF3. Calculate the Err_j for every output neuron j as for example:
$Err_j = y_j - o_j$, where y_j is the jth element of the desired output vector **y**.

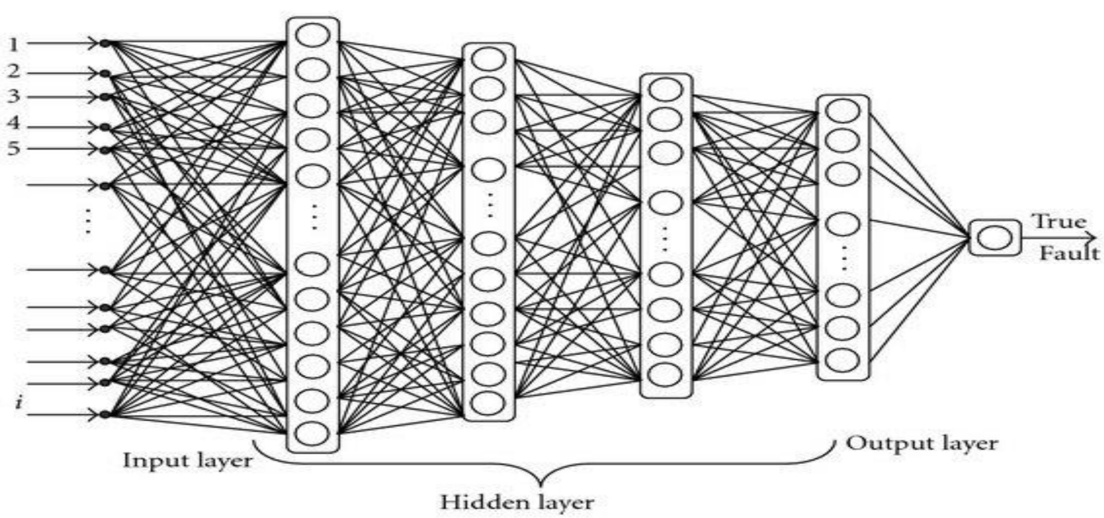
Backward pass:

BB1. Adjust the weights between the intermediate neurons i and output neurons j according to the calculated error:
$\Delta w_{ij}(t+1) = lrate.\ o_j(1 - o_j).\ Err_j.\ o_i + momentum.\ \Delta w_{ij}(t)$

BB2. Calculate the error Err_i for neurons i in the intermediate layer:
$Err_i = \sum Err_j.\ w_{ij}$

BB3. Propagate the error back to the neurons k of lower level:
$\Delta w_{ki}(t+1) = lrate.o_i(1 - o_i).\ Err_i.x_k + momentum.\ \Delta w_{ki}(t)$

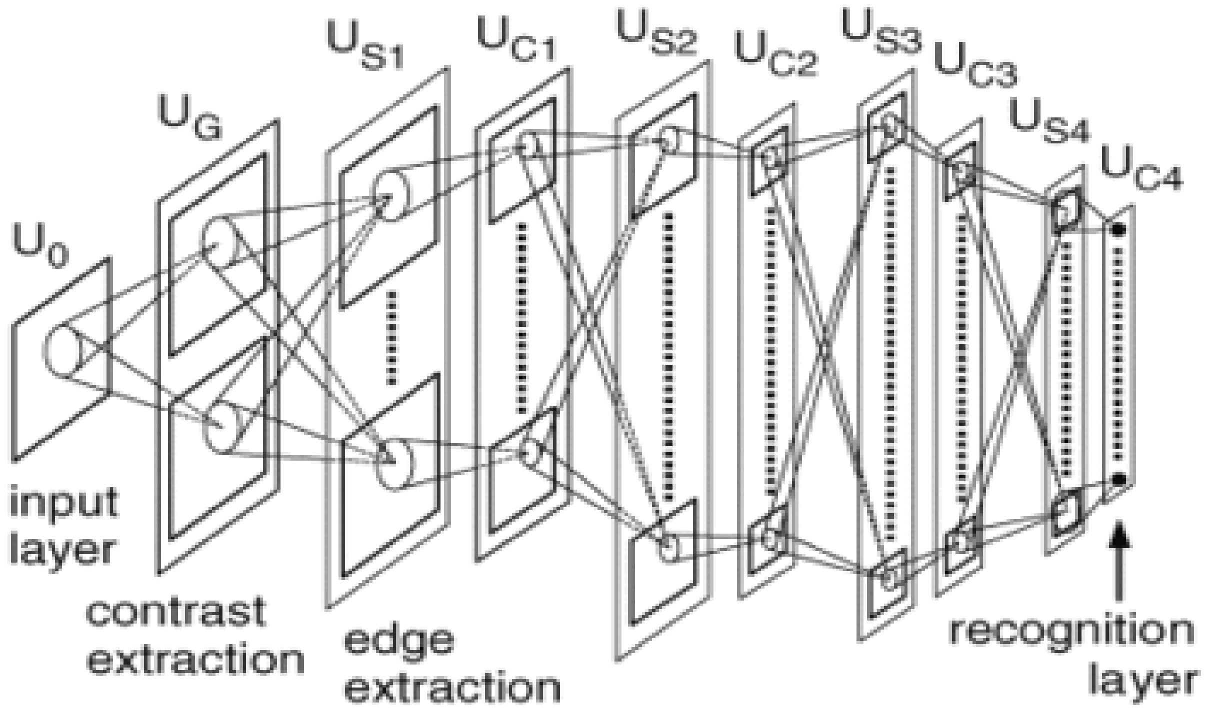
  - Multiple layers MLP, still trained by a BP algorithm:



Early deep convolutional NN in computer vision

Spatial features are represented (learned) in different layers of neurons

Fukushima's Cognitron (1975) and Neocognitron (1980) for image processing



ChatGPT (https://chat.openai.com) describes DNN as:

- Deep Neural Networks (DNNs) are a subset of Artificial Neural Networks (ANNs) that consist of multiple layers of interconnected neurons. These networks are designed to mimic the way the human brain works, by processing information through multiple layers of neurons that progressively extract more complex features from the input data.

- Each layer in a DNN is composed of a set of neurons that receive input from the previous layer and produce output that is passed to the next layer. The neurons in each layer are connected to the neurons in the next layer by a set of weights, which determine the strength of the connection between them. During the training phase of the network, these weights are adjusted using a process called backpropagation, which involves iteratively adjusting the weights to minimize the difference between the network's predicted output and the actual output.

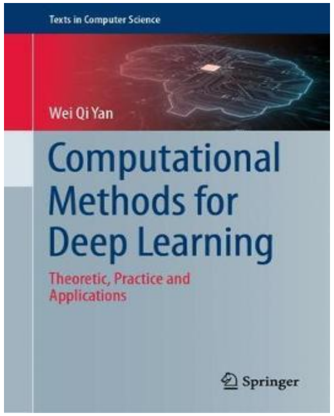
- The number of layers in a DNN can vary, but typically they have at least three layers: an input layer, one or more hidden layers, and an output layer. The input layer is where the input data is fed into the network, and the output layer produces the final output of the network. The hidden layers are responsible for processing the input data and extracting features that are used to make predictions.

Advantages and problems with DNN

Advantages:

- Deep NN are excellent for vector, frame- based data (e.g. image recognition)

- DNNs have been successful in a wide range of applications, including image and speech recognition, natural language processing, and autonomous driving.



Problems:

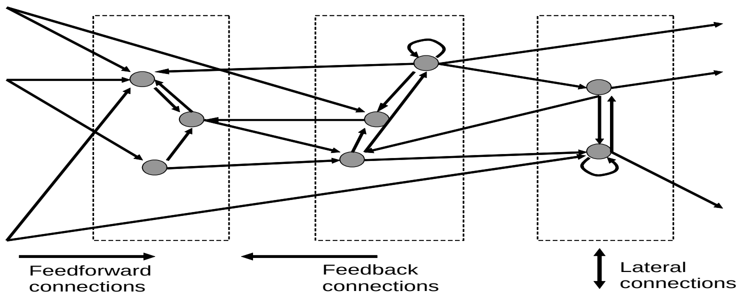
- Modelling of spatio-temporal data

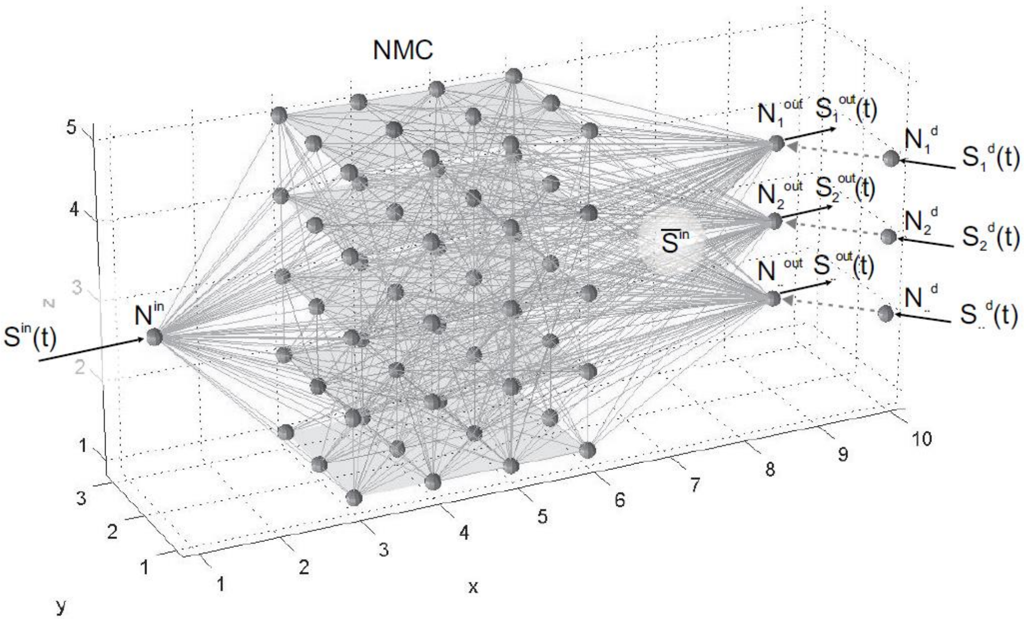
- Knowledge extraction and explainability.

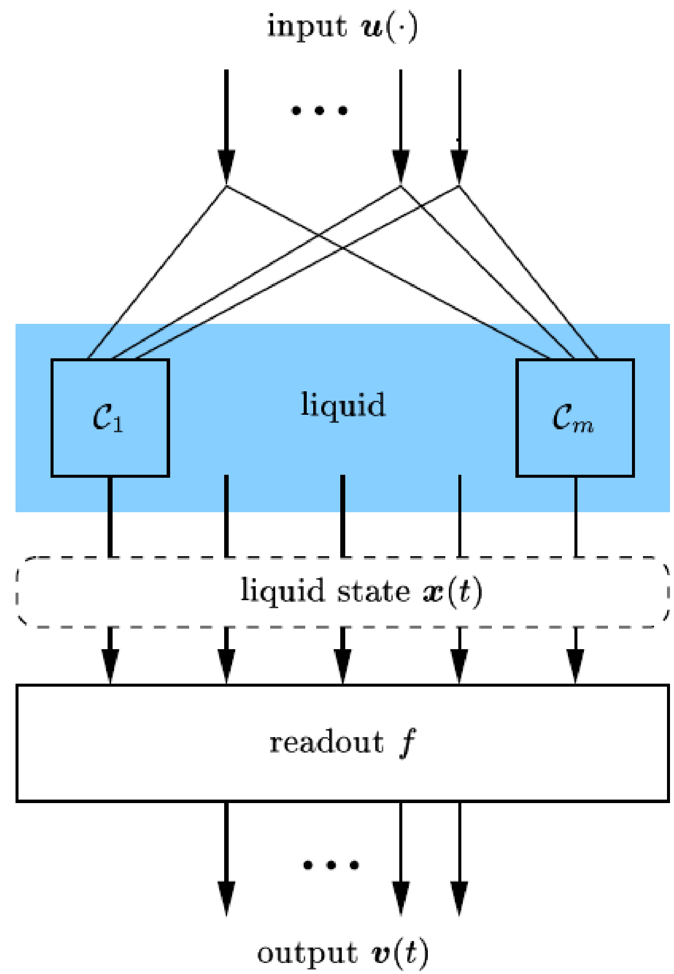
- Training deep neural networks can be challenging due to the large number of parameters and the potential for overfitting. Techniques such as regularization, dropout, and batch normalization are often used to improve the performance of DNNs and prevent overfitting.

nkasabov@aut.ac.nz

2. **Recurrent NN. Reservoir Computing and Liquid State Machine (LSM)**



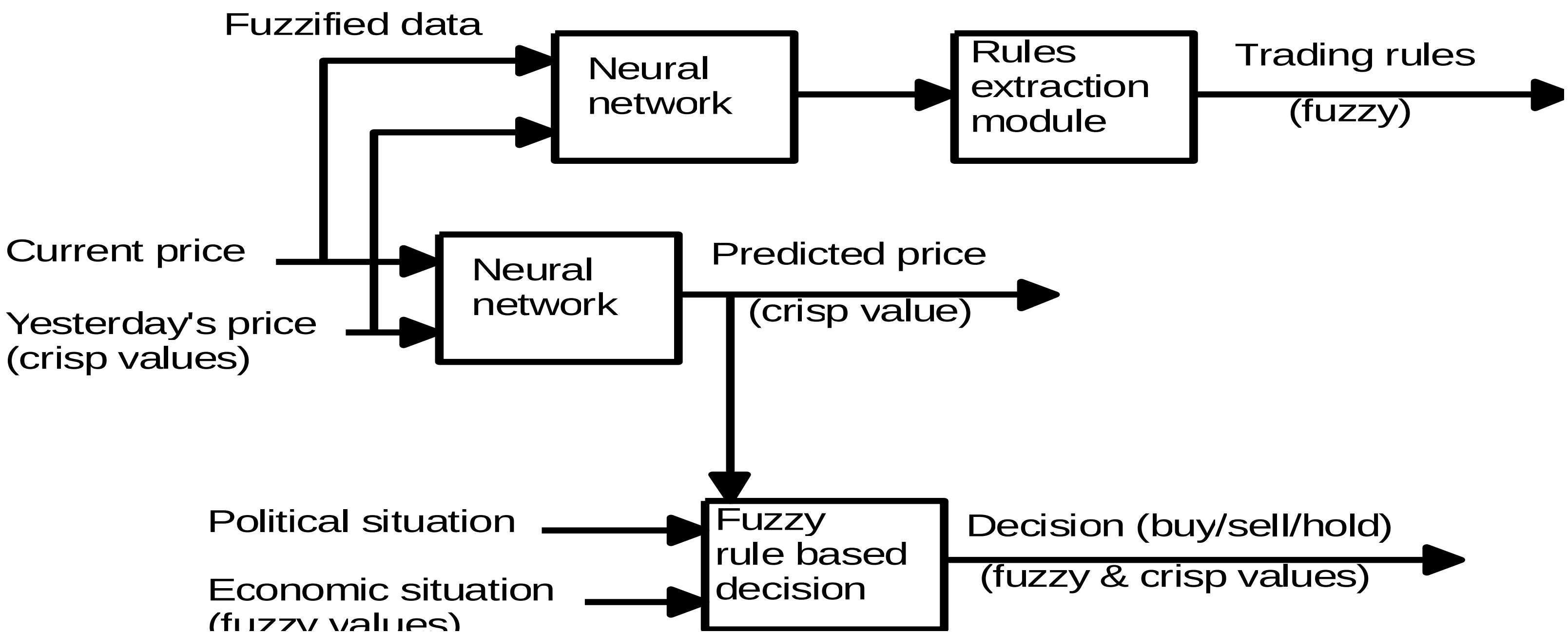




The liquid transforms the input into Liquid states *x(t)* which are mapped by a **readout function** to a trainable classifier to [produce an output *v(t)=f(x(t))* .

3. Hybrid systems that combine NN and fuzzy rule based models

(Example stock value prediction)



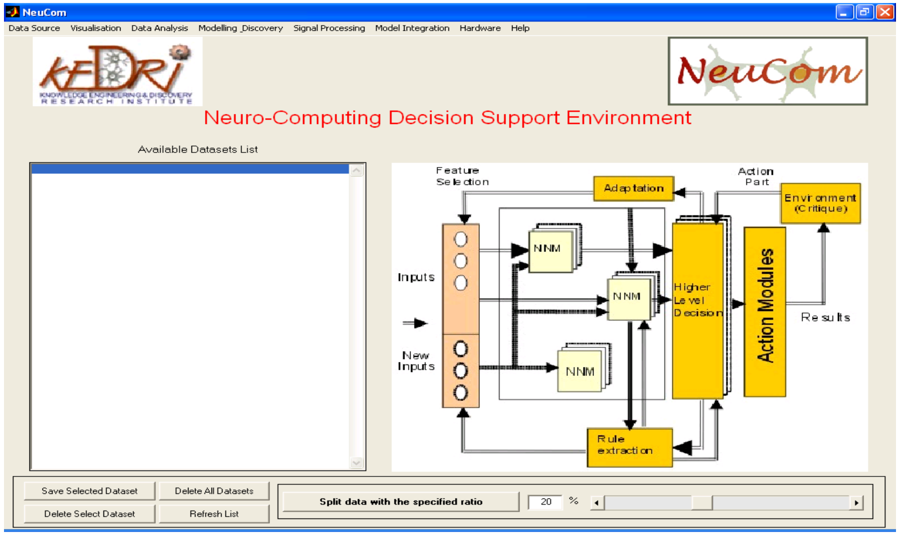
1. N.Kasabov, Time-Space, Spiking Neural Networks and Brain-Inspired AI, Springer 2019 (course book).
2. N. Kasabov Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering, MIT Press, 1996 (additional reading)

4. Developing NN applications in the NeuCom software environment (www.theneucom.com)

Computational modelling with NN:

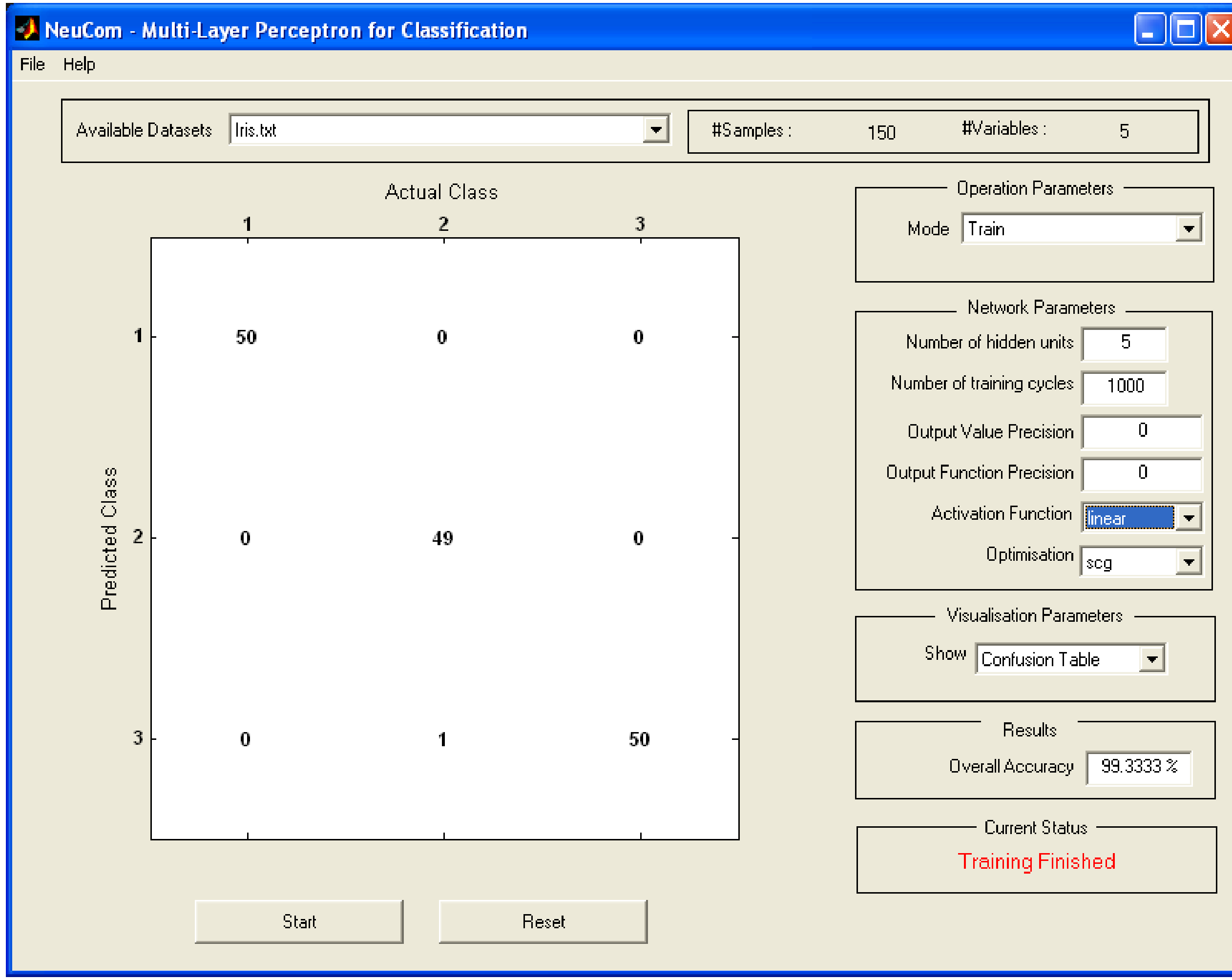
- Data preparation;
- Feature ranking and feature selection;
- NN methods for classification;
- NN methods for regression (time series prediction);
- NN methods for explanation (rule extraction; knowledge discovery);

- NeuCom is a generic environment, that incorporates 60 traditional and new techniques for intelligent data analysis and the creation of intelligent systems
- A free copy available for education and research from: www.theneucom.com



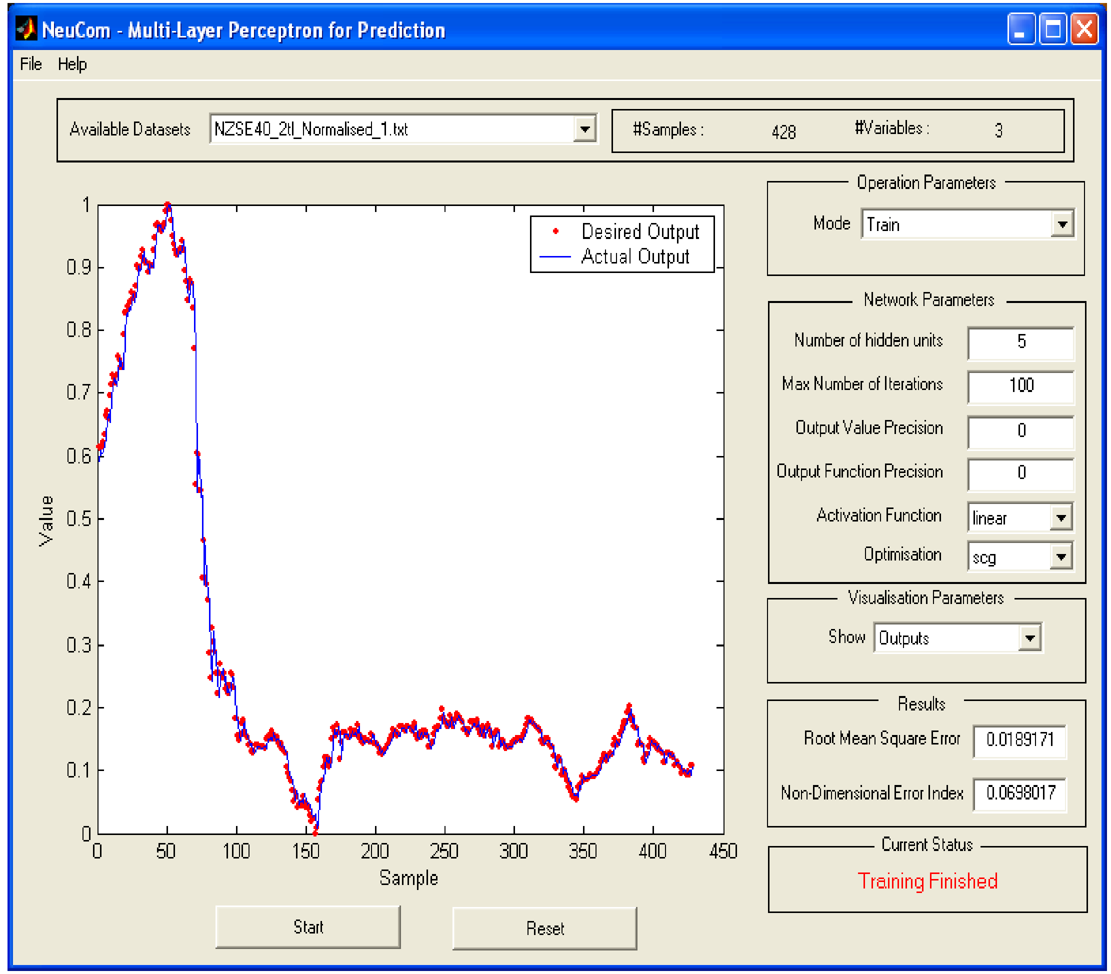
MLP for classification

- The outputs are class labels
- Calculating the confusion matrix:
  - True-positive (sensitivity)
  - True negative (specificity)
- Iris data
- Comparison between different methods in NeuCom



MLP for regression

- Time series prediction
- Choosing the time-lags and the features
- Case studies using NeuCom
- Training on data
- Model verification
- Gas furnice time series prediction
- Stock index time series prediction



Course References

1. N.Kasabov, Time-Space, Spiking Neural Networks and Brain-Inspired AI, Springer 2019 (course book).
2. N. Kasabov Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering, MIT Press, 1996 (additional reading)
3. N.Kasabov, Evolving connectionist systems, Springer 2003 and 2007 (additional reading)
4. Kasabov, N. (ed) (2014) The Springer Handbook of Bio- and Neuroinformatics, Springer. (additional reading)
5. NeuCube: http://www.kedri.aut.ac.nz/neucube/
6. NeuCom: https://theneucom.com
7. KEDRI R&D Systems available from: http://www.kedri.aut.ac.nz
8. N. Kasabov, et al, Design methodology and selected applications of evolving spatio- temporal data machines in the NeuCube neuromorphic framework, Neural Networks, v.78, 1-14, 2016. http://dx.doi.org/10.1016/j.neunet.2015.09.011.
9. Furber, S., To Build a Brain, IEEE Spectrum, vol.49, Number 8, 39-41, 2012.
10. Benuskova, L., N.Kasabov (2007) Computational Neurogenetic Modelling, Springer, New York
11. Indiveri, G. et al, Neuromorphic silicon neuron circuits, Frontiers in Neuroscience, 5, 2011.
12. Kasabov, N. (2014) NeuCube: A Spiking Neural Network Architecture for Mapping, Learning and Understanding of Spatio-Temporal Brain Data, Neural Networks, 52, 62-76.
13. Kasabov (2010) To spike or not to spike: A probabilistic spiking neural model, Neural Networks, v.23,1, 16-19
14. Merolla, P.A., J.V. Arhur, R. Alvarez-Icaza, A.S.Cassidy, J.Sawada, F.Akopyan et al, “A million spiking neuron integrated circuit with a scalable communication networks and interface”, Science, vol.345, no.6197, pp. 668-673, Aug. 2014.
15. Wysoski, S., L.Benuskova, N.Kasabov (2007) Evolving Spiking Neural Networks for Audio-Visual Information Processing, Neural Networks, vol 23, issue 7, pp 819-835.
16. Kasabov, Nikola; Tan, Yongyao Tan; Doborjeh, Maryam; Tu, Enmei; Yang, Jie (2023): Transfer Learning of Fuzzy Spatio-Temporal Rules in the NeuCube Brain-Inspired Spiking Neural Network: A Case Study on EEG Spatio-temporal Data. TechRxiv. Preprint. https://techrxiv.org), https://doi.org/10.36227/techrxiv.21781103.v1, licence CC BY 4.0)
17. Nikola K. Kasabov, Iman AbouHassan, Vinayak G.M. Jagtap, Parag Kulkarni, Spiking neural networks for predictive and explainable modelling of multimodal streaming data on the Case Study of Financial Time Series Data and on-line news, SREP, Nature, pre-print on the Research Square, DOI: https://doi.org/10.21203/rs.3.rs-2262084/v1, licence CC BY 4.0,

- https://orcid.org/0000-0003-4433-7521
- https://knowledgeengineering.ai
- http://scholar.google.com/citations?hl=en&user=YTa9Dz4AAAAJ&view_op=list_works
- https://www.scopus.com/authid/detail.uri?authorId=35585005300

5. Questions, exercises, assignments and project work

1. What are Convolutional Neural networks (CNN)?
2. What are Deep Neural Networks (DNN)?
3. What are the advantages and the problems with DNN?
4. What is reservoir computing and what are LSM?
5. What are hybrid systems?
6. How do you develop a NN application in NeuCom?

nkasabov@aut.ac.nz https://www.knowledgeengineering.ai