Theory Problems of Intelligent Control

Wen Yu

https://www.ctrl.cinvestav.mx/~yuw/
yuw@ctrl.cinvestav.mx

Courses

Theory Problems of Intelligent Control

- Intelligent control - 60 hours
- Theory problems of deep learning in automatic control - 60 hours
- Time series forecasting using deep learning - 60 hours

Theory Problems of Intelligent Control - 30 hours

Neural networks for control

1. Structure
2. Learning
3. Modeling
4. Control
5. Deep Learnig

Fuzzy control

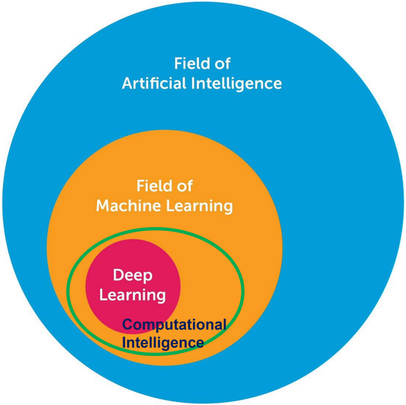
1. Fuzzy System
2. Neuro-fuzzy control

Reinforecement Learnig

Content of NN

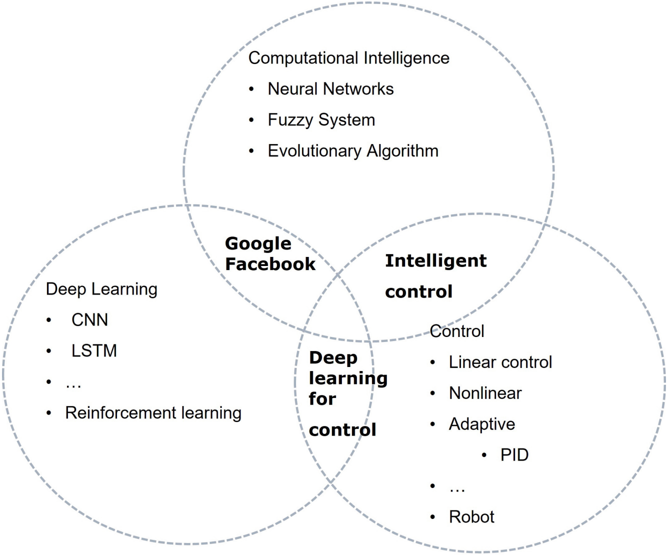
1. Learning methods of neural networks
2. Modeling and control with neural networks
3. Convolutional neural networks (CNN) for modeling and control
4. Long-short term memory (LSTM) for modeling and control
5. Reinforcement learning (RL)
6. Optimal control with RL
7. Time series forecasting
8. Time series forecasting using LSTM and CNN

Deep Learning and Artificial Intelligent



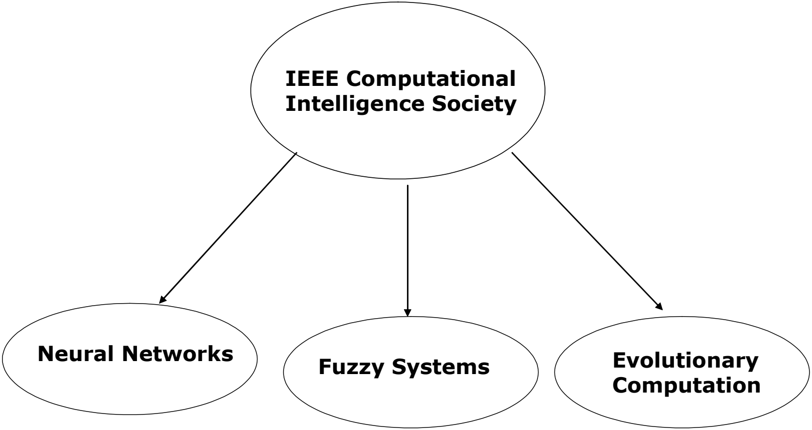
- Stuart Russell, Peter Norvig, *Artificial Intelligence: A Modern Approach*, Pearson, 4th ed, 2020

Intelligent control and Computational Intelligence



Reinforcement learning and deep learning

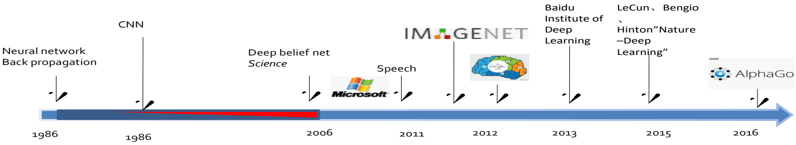
Before 2012



Intelligent control techniques include

1. Fuzzy system (FS)
2. Neural network, deep learning (NN)
3. Evolutionary computation (EC)
4. Reinforcement learning (RL)

Reinforcement learning and deep learning



- 2017. DeepMind's AI (AlphaGo) beats world's best Go player Jie Ke.

- AlphaGo improved its game after playing itself millions of times: AlphaZero
- It uses neural networks (**Deep Learning**) and **reinforcement learning**

Referemce books

1. Martin Hagan, ect., *Neural Network Design* (2nd Edition, 802 pages), PWS Publishing Company, septiembre 2014
2. Li-Xin Wang, *A Course in Fuzzy Systems and Control*, Prentice Hall, Upper Saddle River, NJ, 1997
3. R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction*, 2nd edition, The MIT Press, 2018
4. L.Busoniu, R.Babuska, Bart.Schutter, D.Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators Technology*, CRC Press, 2010
5. Wen Yu, PID Control with Intelligent Compensation for Exoskeleton Robots, Academic Press, 2018
6. Simon Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 2nd Edition, 1998. Neural Networks and Learning Machines, 3rd Edition, Person, 2016

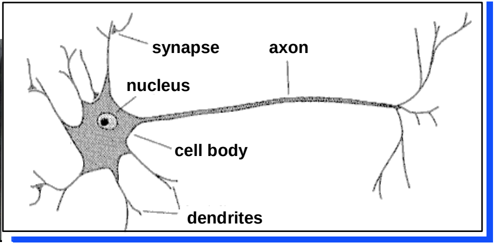
Timeline

PROBLEMAS TEÓRICOS DEL CONTROL INTELIGENTE (28 hrs $\longrightarrow$ 14)

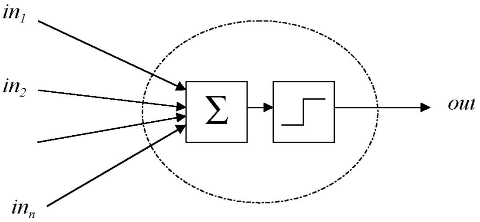
-
    1. NN (6)
    2. DL (2)
    3. Fuzzy (3)
    4. RL (3)

The McCulloch-Pitts Neuron

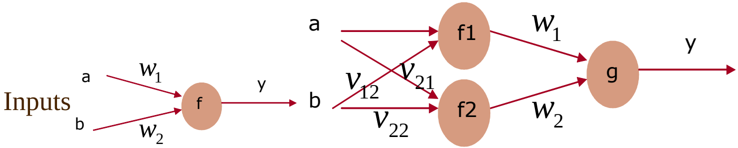
Brain



One neuron



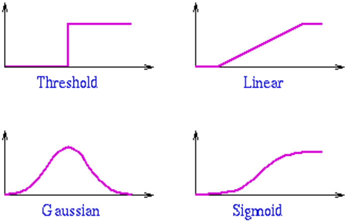
Examples



$$y = f\left(w_1 a + w_2 b\right)$$

$$y = g\left[w_1 f_1\left(v_{11} a + v_{12} b\right) + w_2 f_2\left(v_{21} a + v_{22} b\right)\right]$$

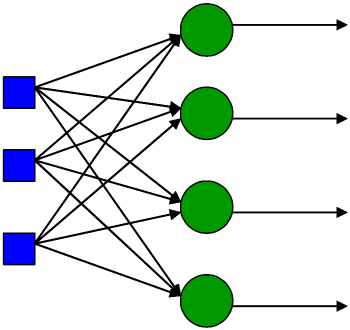
Activation functions



$$y = sgn\left(U\right)$$
$$y = sat\left(U\right), \qquad y = U$$
$$y = e^{\frac{(U-x_0)^2}{\sigma^2}}$$
$$y = \frac{a}{1+e^{-bU}} + c, \qquad y = \tanh(U) = \frac{e^U - e^{-U}}{e^U + e^{-U}}$$

Perceptron

Connect any number of McCulloch-Pitts neurons together.
It is Networks of McCulloch-Pitts Neurons or *Neural Networks*



$$y_k = f_k\left(\sum_{i=1}^{n} W_i u_i\right), \qquad k = 1, 2 \cdots$$

Perceptron Limitations

Linear Separability in Perceptrons

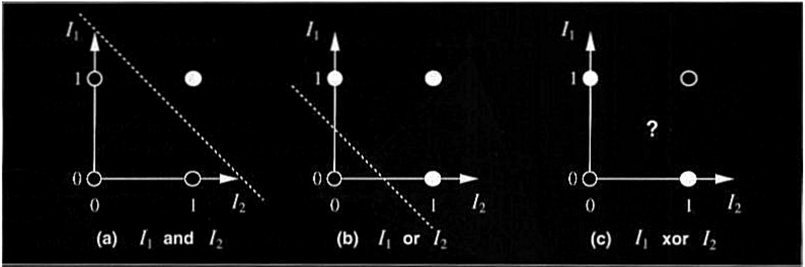
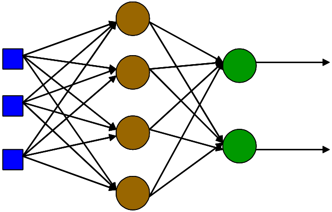


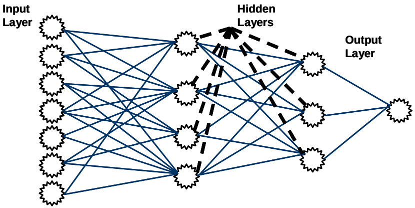
Figure 19.9 Linear separability in perceptrons.

Multilayer Perceptron (MLP)

Single Layer Feed-forward : Rosenblatt's Perceptron



$$y = g\left[\sum_{j=1}^{m} V_j f_j\left(\sum_{i=1}^{n} W_i u_i\right)\right]$$



$$y\left(k\right)=\phi_p\left\{W_p\phi_{p-1}\left[\ldots W_3\phi_2\left\{W_2\phi_1\left[W_1x\left(k\right)\right]\right\}\right]\right\} \tag{1}$$

where $\widehat{y}\left(k\right)\in\Re^m$, $W_1\in\Re^{l_1\times n}$, $b_1\in\Re^{l_1}$, $W_2\in\Re^{l_2\times l_1}$, $b_2\in\Re^{l_2}$, $W_p\in\Re^{l_p\times l_{p-1}}$, $b_p\in\Re^{l_p}$,

The active functions are in sigmoid form,

$$\phi_i\left(\omega_j\right)=a_i/\left(1+e^{-b_i^{\mathrm{T}}\omega_j}\right)-c_i$$

where a_i, b_i, and c_i are prior defined positive constants, ω_j are the input variables to the sigmoid functions.

Or

Univeral approximation

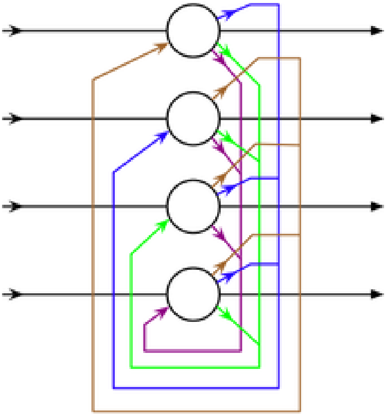
MLP (multilayer Perceptron), SVM, RBFNN, Logistic regression, Linear regression (without hidden layers)

Theorem

(Cybenko) Let activation function of neural network ϕ be a stationary, bounded, and monotone increasing. Then for any continuous functions $y = f(x)$, $x \in R^n$, and any small $\varepsilon > 0$, there exist an integer m, and real constants $W_{1,i}$ and $W_{2,ij}$, such that the neural network with ONE hidden layer satisfies

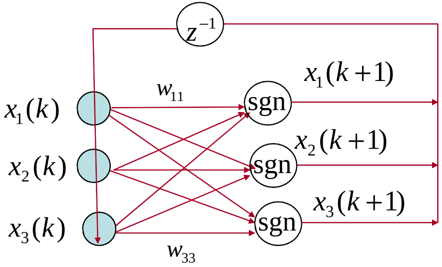
$$\left| \sum_{i=1}^{m} W_{1,i} \phi \left(\sum_{j=1}^{n} W_{2,ij} x_j \right) - f(x) \right| < \varepsilon$$

Hopfield networks (Recurrent NN)



Discrete-time Hopfield neural networks

Parallel form



$$x_i\left(k+1\right) = sgn\left(\sum_{j=1}^{n} W_{ij}x_j\left(k\right) + \theta_i\right), \quad \forall i$$

Timeline

TEMAS SELECCIONADOS APRENDIZAJE PROFUNDO (28hrs $\longrightarrow$ 14)

1. Wen (13/09, 20/09)
2. Shengbo Hong, Eduardo Yudho (27/09)
3. Suting Gao, Carlos Castillo (4/10)
4. Yingqin Zhu, Victor Lechuga (11/10)
5. Saul Villlegas, Edson Cruz (24/10)
6. Fernamdo Ramirez, Jose Padilla (31/10)

- Shengbo Hong, Wen (07/11)
- Yingqin Zhu, Eduardo Yudho (21/11)
- Wen (06/12)