

Bioinformatics CSM17

Week 8: Simulations (1)

Soft Computing:

- Genetic Algorithms
- Evolutionary Computation
- Neural Networks

Genetic Algorithms (GAs)

- simulate sexual reproduction
- use artificial ‘chromosomes’
- simulate evolution

‘Real’ Chromosomes

- humans have 46 in total
 - 23 homologous pairs
- half from each parent

Mitosis

- normal cell division e.g. for growth, repair
- all cells are diploid (usually)
- i.e. they are said to be '2n'

Meiosis

- cell division to produce gametes
 - gametes
 - Female: eggs or ova (singular ovum)
 - Male: sperm
- daughter cells are haploid (n)

Main features of GAs

- crossover (chiasma)
- ‘chromosomes’
- population containing individuals
- successive generations
- survival of the ‘fittest’
- only the ‘most fitted’ reproduce
- (removal of the worst)
- mutation

A Simple Example

- population of 4
- attributes are simple numbers
- fitness function is a minimisation function
- only 2 best fitted survive to reproduce

Mutation

- changes of nucleotide bases
- caused by
 - ionizing radiation, mutagenic chemicals
- usually harmful (damaging)
- may be
 - single base (changing one amino acid)
 - frameshift (more serious)

Karl Sims

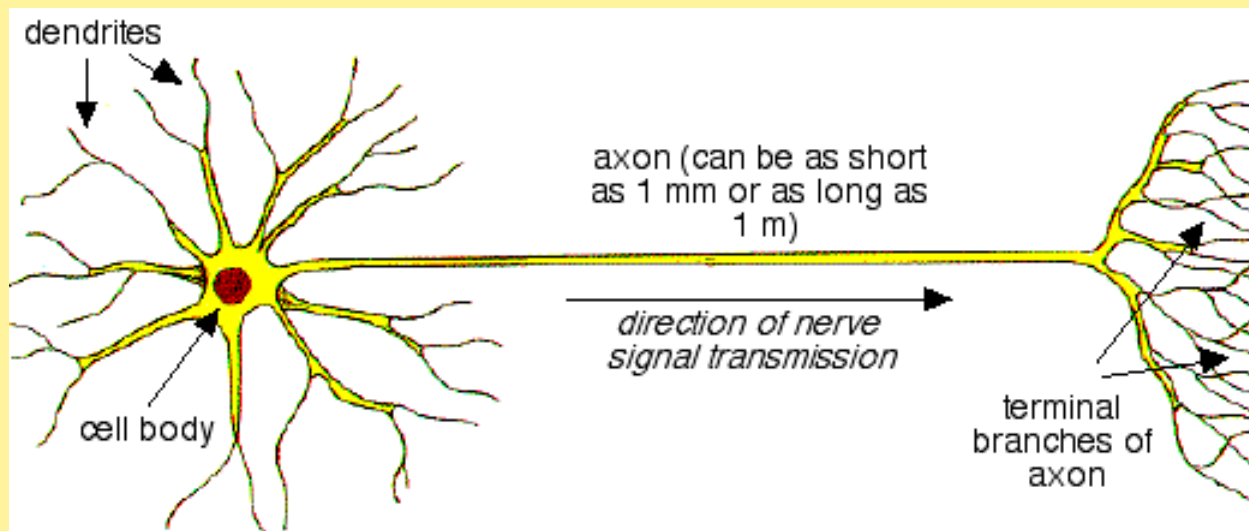
- Evolved creatures
- Swimming
- Jumping
- Walking
- Following....etc.

Neural Networks

- biological neurons
- natural neural networks
- artificial neural networks
- applications

A Biological Neuron has...

- soma (the 'body' of the neuron)
- dendrites (for inputs)
- axon (for output)
- synapses



Natural Neural Networks

- nerve net
 - in Coelenterates
 - *e.g. Hydra*, sea anemones

The Human Brain

- ~100 billion neurons
- about as many trees in Amazon Rain Forest
- the number of connections is about the same as the total number of leaves
- up to 100 thousand inputs per cell

The Human Brain

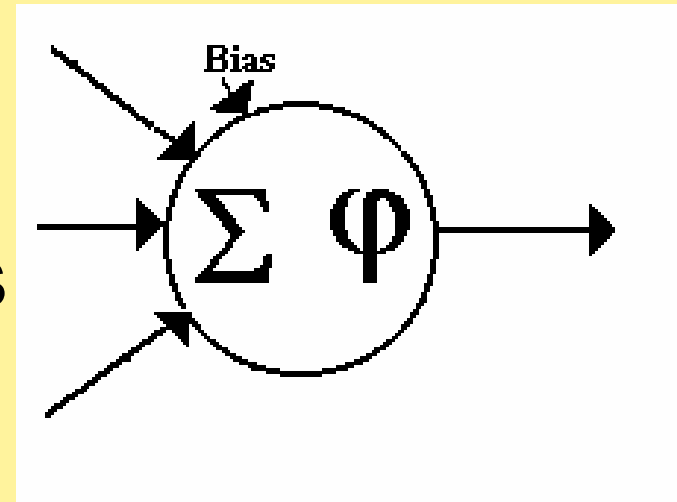
(from the visible human project)



JYC: CSM17

Artificial Neurons

- McCulloch & Pitts
 - single neuron model (1943)
- ... with weights becomes Hebbian Learning
- Rosenblatt's Perceptron
 - multi-neuron model (1957)



Artificial Neural Networks

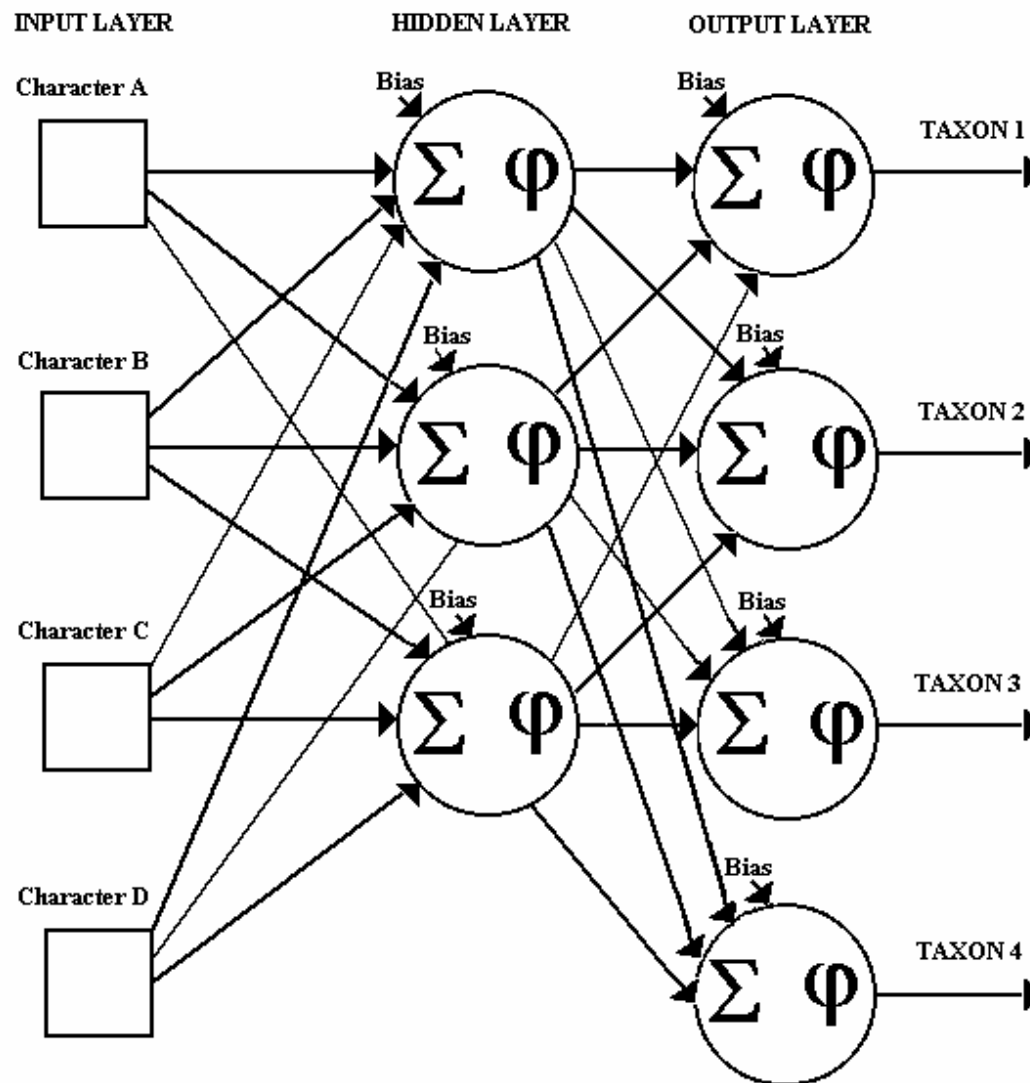
- supervised
 - known classes
- unsupervised
 - unknown classes

Supervised Neural Networks

- multilayer perceptron (MLP)
- used where classes are known
- trained on known data
- tested on unknown data
- useful for identification or recognition

MLP Architecture

- usually 3-layered (I:H:O)
 - one node for each attribute / character
- input layer
 - one node for each attribute / character
- hidden layer
 - variable number of nodes
- output layer
 - one node for each class



MLP Learning Algorithms

- summation is carried out by

$$v_j = \sum_{i=0}^n w_i \cdot x_i$$

- where w_i is the weight and x_i is the input value for input i .

MLP Learning Algorithms

- the non-linear *activation function* (φ) is given by

$$\varphi(v_j) = \frac{1}{1 + \exp(-v_j)}$$

- where v_j is the weighted sum over n inputs for node j

MLP Learning Algorithms

- backpropagation
 - (Werbos) Rumelhart & McClelland 1986
- contribution of each weight to the output is calculated
- weights are adjusted to be 'better' next time...using the delta rule

MLP Learning Algorithms

- delta rule

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j y_i$$

- ... for output nodes

$$\delta_k = y_k [1 - y_k] \bullet (t_k - y_k)$$

- ... for hidden nodes

$$\delta_j = y_j [1 - y_j] \bullet \sum_{k=1}^K [\delta_k w_{kj}]$$

Applications

- identification / recognition
- fault diagnosis e.g. teabag machine
- medical diagnosis
- decision making

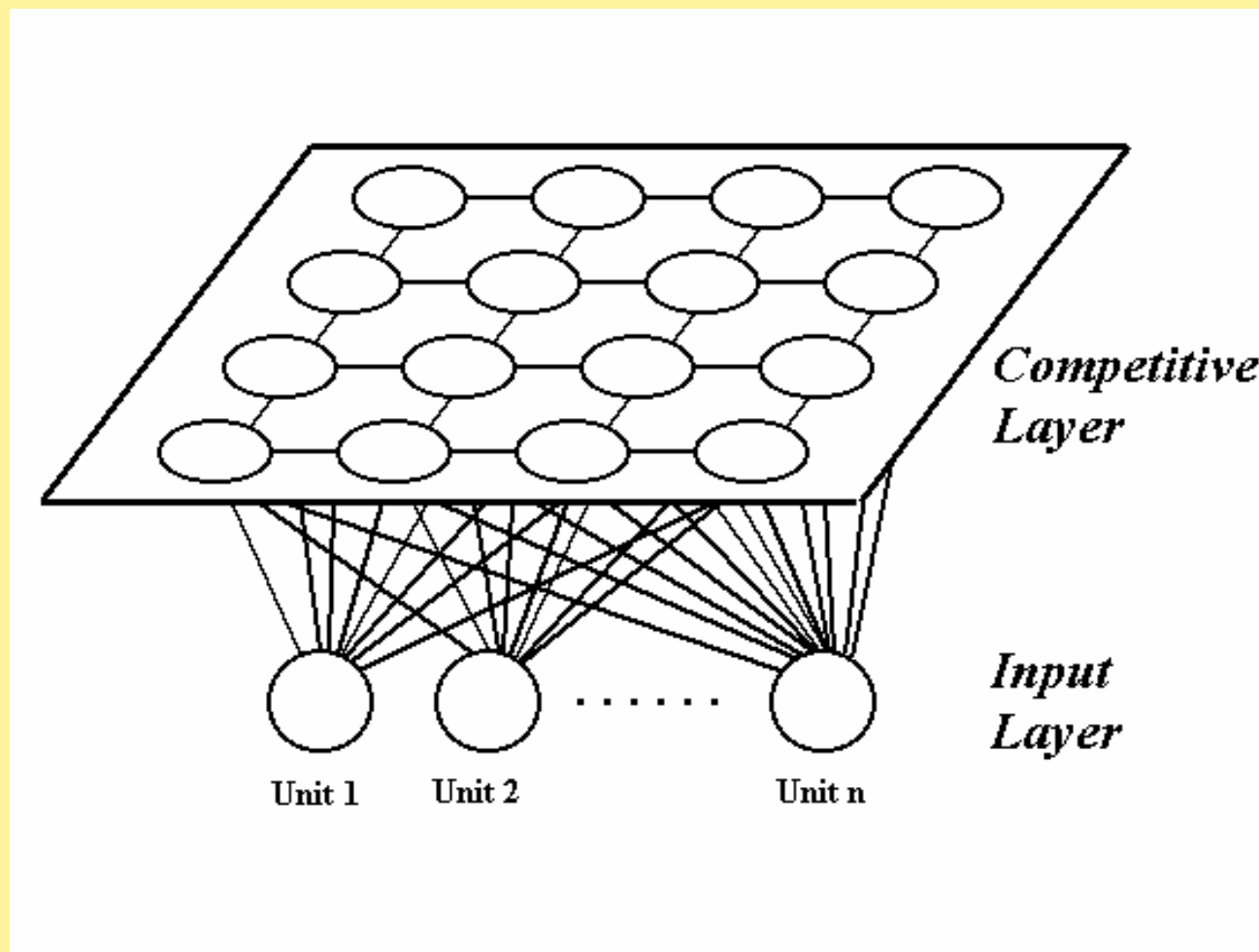
Unsupervised NNs

- self-organising (feature) maps
- 'Kohonen' maps
- topological maps

Kohonen Self-Organising Feature Map (SOM, SOFM)

- Teuvo Kohonen (1960s)
- input layer
 - one node for each attribute / character
- competitive 'Kohonen' layer

Kohonen SOM Architecture



Kohonen Learning Algorithm

- initially random weights between input layer and Kohonen layer
- data records (input vectors) presented one at a time
- each time there is one 'winner' (closest Euclidean distance)
- the weights connected to the winner and its neighbours are adjusted so they are closer
- learning rate and neighborhood size are reduced

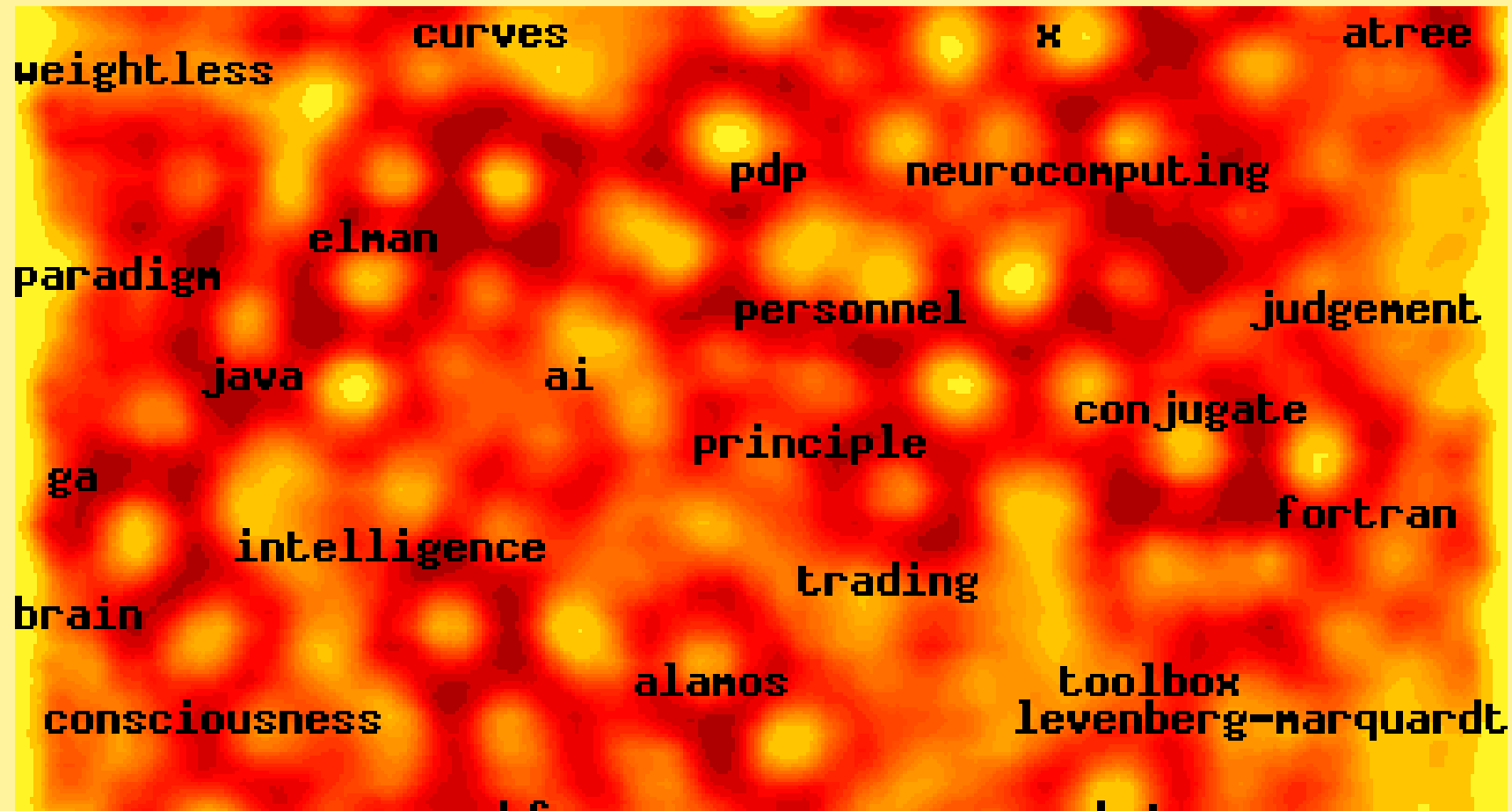
SOM Learning Algorithm

$$i(x) = \arg \min \| x(t) - w(t)_j \|$$

$$\| x(t) - w(t)_j \| = \sqrt{\left[\sum_1^I (x(t)_i - w(t)_{ji})^2 \right]}$$

$$w_j(t+1) = w_j(t) + \eta [x(t) - w_j(t)]$$

WebSOM of comp.ai.neuralnets



Summary

- biological neurons
- natural neural networks incl. the brain
- artificial neural networks
- applications

Useful Websites GAs

- Evolutionary design by computers:
<http://www.cs.ucl.ac.uk/staff/P.Bentley/evdes.html>
- Evolving creatures (Karl Sims):
<http://www.genarts.com/karl/evolved-virtual-creatures.html>

Useful Websites: Neural Nets

Visible Human Project

<http://www.nlm.nih.gov/research/visible/>

Stuttgart Neural Network Simulator (Unix)

<http://www-ra.informatik.uni-tuebingen.de/SNNS/>

Microsoft's List of Neural Network Websites

<http://research.microsoft.com/~jplatt/neural.html>

Neural Network FAQ

<ftp://ftp.sas.com/pub/neural/FAQ.html>

WebSOM

<http://websom.hut.fi/websom/>

GAs: References & Bibliography

- Bentley, P. (ed). Evolutionary design by computers, Morgan Kaufmann. ISBN: 155860605X
- Mitchell, M. (1996). An introduction to genetic algorithms. MIT Press, Cambridge, USA. ISBN 0-262-13316-4
- Gibas & Jambeck (2001). *Bioinformatics Computer Skills*. p401.
- Fogel, G. B. & Corne, D. W. (eds.). (2003) Evolutionary computation in bioinformatics. Morgan Kaufmann. ISBN 1-55860-797-8

Neural Nets: References & Bibliography

- Greenfield, S. (1998). *The human brain : a guided tour*. - London : Phoenix, 1998. - 0753801558
- Greenfield, S. (2000)- *Brain story*. - London : BBC, 2000. - 0563551089
- Haykin, S. (1999). *Neural networks : a comprehensive foundation* , 2nd ed. – Prentice Hall, Upper Saddle River, N.J., USA. 0139083855, 0132733501
- Dayhoff, Judith E. (1990). *Neural network architectures : an introduction*. Van Nostrand Reinhold, New York. 0442207441
- Beale, R., Russell & Jackson, T. (1990). *Neural computing : an introduction*. Hilger, Bristol, UK. 0852742622
- Looney, C.G. (1997). *Pattern recognition using neural networks*. Oxford University Press, New York, USA. 0195079205
- Aleksander, I, & Morton, H. (1990). *An introduction to neural computing*. Chapman and Hall, London. - 0412377802