Laboratory in Neural Modeling
Cognitive Science 102

James Anderson
Department of Cognitive and Linguistic Sciences,
January 25, 2007

Cognitive Science 102 has two main topics:

First, it is an introduction to neural network models for cognition.

Second, it serves as an introduction to the design of a couple of types of "brain-inspired" computing systems that may be helpful for some practical cognitive applications.

The lectures in the course will be in Metcalf Chemistry 204, the computer classroom, on Tuesday and Thursdays, 10:30-11:50.

There will be several Cognitive Science Colloquium talks this year Monday at 4:00, with a couple of talks directly related to course material. The Neuroscience Seminar Series on Thursdays at 4 also has occasional talks of interest. I will announce them when they occur. Although they are not required you will learn a lot by attending.

The instructor of this course is James Anderson. The teaching assistant this year is Socrates Dimitriadis. Email addresses are socrates@brown.edu and James_Anderson@brown.edu. We can also be reached on the telephone (Anderson X32195; Dimitriadis, X31167 and as well as physically, (Anderson Metcalf 221; Socrates, Metcalf 136.) Conference hours will be arranged after the times for the sections are chosen. Dimitriadis and Anderson are frequently available at times other than conference hours but may be surly when disturbed, Anderson in particular. The email system is much preferred for most communications. Email requires precision in formulation that is conducive to well posed questions and rapid answers. No text messaging.

Thanks to Microsoft, Brown, and the Burroughs-Wellcome Foundation, we have a number of machines, a video projector, plus some (relatively) new furniture in Metcalf 204.The computers in room Metcalf 204 can be used for programming and assignments if you need them. If you plan to use these machines contact us because we have to arrange it so your Brown ID can be used as an access card to the room. You can use any language or any machine you want. We currently do no computing that is beyond the capabilities of a basic PC or Mac. However, if you use another machine, or use a language other than the ones we know, you are on your own as far as most technical help is concerned.

Anderson can help with programming problems relating to Pascal and FORTRAN (don't laugh) and Dimitriadis has a good knowledge of Matlab, C and C++ as well. Please check your email for announcements relating to the course and make sure to provide us with a useful email address. There are any number of good programming environments and compilers available for both PC's and Mac's. The computer assignments are simple numerical programs and not time consuming programming exercises.

There are a number of good programming environments around so feel free to use your favorite. Again, however, if you get into trouble with a system we do not know, we cannot help you.

Anderson is particularly partial to the easy to use and flexible programming environment **Delphi** from Borland. Delphi-developed Windows programs will be used for several classroom demos during the term. Code is available for any demos. Some fully functional, versions of Delphi for student use are available free. (Check out the Wikipedia article on Delphi http://en.wikipedia.org/wiki/Borland_Delphi .)

The Departmental Computer Coordinator is Robert Fifer. His email address is Robert_Fifer@brown.edu. He can help with network and communications problems and any problems with accounts on the computers in Metcalf 204.

Course Topics

1. Biological background: Brief discussion of neuron function and brain organization. Neurons as transducers and integrators of their inputs. Fascinating and profound historical anecdotes combined with insightful philosophical digressions.

2. Vector and matrix operations (very brief, possibly absent).

3. Random number generators. Generation of random vectors. Statistical properties of random vectors. Brief discussion and demonstration of **Monte Carlo** simulation methods.

4. *Limulus*: a simple visual system and its simulation. The wonders of animal eyes.

5. Memory. The Hebb synapse and the linear associative model. The biology of the **Hebb** synapse. Systems build using the simplest form of Hebb synapse.

6. The linear associator extended. Simple error correction algorithms. Gradient descent. The **Widrow-Hoff** (LMS) error correcting algorithm: its use and limitations.

7. Concepts, categories, and prototypes in cognitive science. ‘Concept forming' neural net systems and their implications and applications. Why the hard cognitive problem is not “processing” data but deciding how best to throw almost all of it away. (Could God have done better with more bandwidth?)

8. Associative computation and semantic networks. Network disambiguation.

9. Attractor networks and non-linear dynamical systems. **Hopfield** networks, **Boltzmann** machines. Energy functions. The **BSB** model. Simulated annealing. Categorical perception. Unsupervised clustering and a radar application.

10.A rapid discussion of some important aspects of mammalian neocortical function leading to:

11. Speculations about brain-like computation: hardware and software. Our Ersatz Brain Project, a valiant attempt to define the architecture of a brain-like computer. What the Ersatz brain can do, how it does it, and how to get someone else to pay for designing it. Building a brain-like computer for fun, insight, and profit.

12. Research project in neural modeling. This is the most important part of the course and will form a major part of your grade, if such things concern you. Neural networks in particular and brain-like computers in general are tools to be applied to specific problems. You should be thinking about what you want to do for a project all during the term.

Texts

1. (Required) *An Introduction to Neural Networks*, James A. Anderson, MIT Press.

2. A good book on linear algebra is useful. There are many good text books. Avoid the very abstract mathematical ones.

3. A good reference book for the mathematically inclined is Simon Haykin's, *Neural Networks* (Prentice-Hall), now in a second edition with a third edition to arrive on May 1. This book is designed for graduate level engineers and is difficult but very good on the mathematical analysis of algorithms. Haykin has an encyclopedic knowledge of the engineering neural net literature. The book is, unfortunately, expensive.