Syllabus

Psychology 331, Cognitive Psychology Lab:
Neural Network Modeling
Spring, 2008

Instructor: John Hummel
Email: jehummel@uiuc.edu
Office: 825 Psychology Building
Off. Hrs.: by appointment
Lecture: Monday, 2:00 – 3:50

TA: Gary Oppenheim
Email: goppenh2+nn@uiuc.edu
Office: 834 Psychology Building
Off. Hrs.: by appointment
Lab Sect. 1: Tues./Thurs., 9:00 – 10:50
Lab Sect. 2: Tues./Thurs., 11:00 – 12:50

We will hold our office hours either in our offices or in the lab.

Lecture: Psychology Rm. 29; Monday, 2:00 – 3:50
Lab: Psychology 219A; TR, 9:00 – 10:50; TR 11:00 – 12:50

Purpose of Course

This course introduces the basics of neural network modeling. It emphasizes the formal characteristics of three main classes of networks and the evaluation of neural networks as models of human perception and cognition. The course requires familiarity with basic mathematics (matrix algebra and calculus are recommended; basic algebra is a must) and computer programming skills. Laboratory assignments will be completed in the programming language MATLAB.

Course Format:

Students will attend two hours of lecture each week (taught by the Instructor) and four hours of laboratory work (taught by the TA). Your grade will be based on one quiz (15 points), and five modeling projects of increasing complexity (10, 10, 20, 25, and 30 points each). Final grades will be assigned on the basis of a curve, but if everyone does really well, grades will be assigned on an eight point scale (93% and above is an A, 85-92% is a B, etc.), so there's no way you'll get 95% and end up with a C.

Quiz: The quiz will consist of short answer questions (e.g., one sentence to one paragraph answers). Anything from the lectures, class discussions or readings is fair game for a quiz question.

Late Assignments: Late projects will be penalized 10% per day. Points will be deducted until the TA has the project in hand. If anyone misses the quiz, we will arrange one time and place for a makeup. Only one makeup will be arranged, so if you miss the makeup, you lose the points. Be warned: I will make the make-up quiz harder than the original, so try not to miss the original!

Readings: The readings are available on the Electronic Reserves web page. A link is available on my web page: http://www.psych.uiuc.edu/people/showprofile.php?id=543
<table>
<thead>
<tr>
<th>Week</th>
<th>Date (Mon)</th>
<th>Lecture</th>
<th>Lab</th>
<th>Reading</th>
<th>Assignment Due (Mon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/14</td>
<td>Intro &amp; Overview • Why model?</td>
<td>Intro to MATLAB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1/21</td>
<td><strong>Hummel Away</strong></td>
<td>More Intro to MATLAB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1/28</td>
<td>Properties of NNs • States as vectors • Linear algebra</td>
<td>More Intro to MATLAB Matrix operations</td>
<td>Jordan</td>
<td>MATLAB Exercises and Linear algebra</td>
</tr>
<tr>
<td>4</td>
<td>2/4</td>
<td>Linear algebra • Activation functions</td>
<td>Activation functions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2/11</td>
<td>Activation functions • Hopfield</td>
<td>Applying MATLAB to NN architectures (in prep. for Hopfield)</td>
<td>Hopfield</td>
<td>Activation functions</td>
</tr>
<tr>
<td>6</td>
<td>2/18</td>
<td>Hopfield Model • Assignment • How to write a paper</td>
<td>Hopfield</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>2/25</td>
<td>Hopfield Model</td>
<td>Hopfield</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>3/3</td>
<td>Nonlinear perceptrons • Widrow-Hoff learning algorithm</td>
<td>Applying MATLAB to NN architectures (in prep. for Back Prop)</td>
<td>Rumelhart et al.</td>
<td>Hopfield</td>
</tr>
<tr>
<td>9</td>
<td>3/10</td>
<td>Back propagation</td>
<td>Back Prop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>3/17</td>
<td>Spring Break</td>
<td>Spring Break</td>
<td>Spring Break</td>
<td>Spring Break</td>
</tr>
<tr>
<td>11</td>
<td>3/24</td>
<td>Back propagation</td>
<td>Back Prop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>3/31</td>
<td>* Quiz * • Back propagation, mappings vs. variablized functions</td>
<td>Back Prop</td>
<td>Von der Malsburg</td>
<td>Quiz</td>
</tr>
<tr>
<td>13</td>
<td>4/7</td>
<td>Unsupervised learning &amp; von der Malsburg model</td>
<td>Unsup. Learning</td>
<td></td>
<td>Back Propagation</td>
</tr>
<tr>
<td>14</td>
<td>4/14</td>
<td>Unsupervised learning &amp; von der Malsburg</td>
<td>Unsup. Learning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>4/21</td>
<td>Review &amp; catch up</td>
<td>Unsup. Learning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>4/28</td>
<td>Review &amp; catch up -- no lab --</td>
<td>-- no lab --</td>
<td></td>
<td>U. Learning</td>
</tr>
</tbody>
</table>
How to Write a Paper Describing a Neural Network

The Rule: Remember why you are simulating the neural net in the first place.

You implement a neural net so you can see how it behaves:

With neural nets (and other forms of computer modeling), you have some ideas about how some process (e.g., content-addressable memory) works

You think you know how the individual micro-components work, but you cannot prove analytically how they will all work together. As an aside: Simulations is only necessary when there is no analytic solution. If you could prove mathematically/logically that your system must behave a certain way with a certain input, then there is no need to simulate it.

So you write a computer program to implement your ideas and you watch for yourself how it works.

What this means is that you have to start with a set of well-motivated questions:

Motivation: I have reason to believe that the kinds of patterns I train it on will affect its performance. Question: How will this network work if I train it on this type of pattern or that type of pattern?

Motivation: I have reason to believe that the network will have a limited memory capacity. Question: How will it perform if I train it on this many patterns? That is, how does it scale up with larger training sets?

Motivation: People behave in such-and-such a way in such-and-such a situation. Question: Does my network capture (i.e., accurately simulate) this aspect of human performance.

The purpose of running simulations is to answer these questions for yourself.

The purpose of writing about your simulations and their results is to communicate these questions and their answers to other people.

Although this may seem obvious, most papers about NNs (both by students and by pros) are not written with an eye toward organizing the paper around this theme.

However, the good ones are.

What this means is that your paper must make the questions you are asking very, very clear. It must also make the relationship between the questions and their answers very clear. This happens at several levels of abstraction all at once:

The model as a whole is a suggested answer to a general kind of question. Both this general question and the proposed (general) answer are posed in the Introduction.
The nitty-gritty of the model consists of a bunch of answers to a bunch of specific questions (How should nodes be connected? How should they update their activations?, etc.) These are described in the **Model Operation** section.

Each simulation you run is an attempt to answer a specific question about how the model behaves. The questions, the simulations you ran to answer them, and their answers (the simulation results) are detailed in the **Simulations** section.

The simulations and results must both be very clear: What did you do **exactly** and why? What did the model do and what does it mean? Graphics are a very good way to express simulation results.

This section should be very clear about (1) How each simulation you ran relates to the question it is supposed to answer (Why can this simulation answer this question?) and (2) How the simulation results translate into an answer. (Why do the simulation results imply that the answer to the question is $x$ rather than $y$?)

By the end, hopefully you will have learned something more general, going back to the big question that motivated the model in the first place. This is summarized, usually by reference to the broad strokes of the simulation results, in the **Discussion** section.

Remember that your reader does not have the same answers and questions in mind as you do -- otherwise they would not be reading your paper. Thus, you must describe these things in your paper as if they are for people who do not know beforehand what you did and why.
Grading Criteria

We will reward clarity & brevity. We won't deduct points for papers shorter than the recommend length, but we will stop reading after the limit. Grading will emphasize organization, clarity, logic, accuracy, completeness and insightful analysis.

Below is an exact copy of the grading criteria I gave my TA for the final projects the last time I taught this course. We will use these same criteria this time. The points listed (e.g., 6 points for excellent program code, etc.) apply to the final project only (which is worth 30 points), but the percentages (e.g., 20% for the code) are applicable to all the projects. More importantly, the criteria will be the same on all the projects. The point of this handout is to help you understand what we think makes a good paper. Hopefully, you will use this information to improve the quality of your work rather than simply engineer your paper to get a good grade. Also, we reserve the right to change the percentages, and criteria slightly, but if we do, we'll inform you first.

1) Program Code and Output (20%). -- 6 points
The code must be well enough commented that a non-native MATLAB speaker can understand what it is doing, how and why. Same with the output.

Point Assignment:
1-2: Varying degrees of inadequacy
3: Adequate -- you can understand it with effort
4: Good -- you can understand it with little effort
5: Very Good -- you can easily understand it
6: Excellent -- you can easily understand it and it’s extremely clever

The output should: Demonstrate that the program works. Include detailed and readable outputs for each of your simulations.

2) Paper (80%) -- 24 points.
a) Introduction: -- 6 points
The intro must make explicit:
1) What is the general question or problem?
2) What is the general form of the proposed solution, or how do they intend to go about answering the question/attacking the problem?
3) How does the model’s operation relate to the general answer proposed in (2).

Point Assignment:
1-2: Intro fails to answer one or more of (1), (2) or (3) above, or it answers all three but it took effort on your part to figure that out
3-4: Intro clearly answers (1) … (3)
5-6: Intro brilliantly answers (1) … (3)
b) Description of the model: -- 6 points
This part must make clear how the model works in its essential characteristics. Ideally, you should
be able to implement the model yourself based only on the student’s description.

Point Assignment:
1: Inadequate -- you cannot understand how the important parts of the model work
2: Adequate -- you can understand the model, but with effort
3-4: Good -- you can easily understand the model
5-6: Very Good -- you are impressed by how easily you can understand the model or by
   how well the student obviously understands the model

c) Simulation results: -- 6 points
This part should be evaluated both on the description of the motivation for each simulation, the
description of the results and the description of what the results mean vis. the question.
For each simulation, the student must make explicit:
1) What is the question this simulation is designed to address? I.e., What is the purpose of the
   simulation?
2) How did they go about asking the question? I.e., What is the method: What was the
   simulation they ran? What did they manipulate, and how does the manipulation relate to the
   purpose of the simulation?
3) What was the result? The results should be expressed both mathematically or graphically and
   verbally.
4) How do these results speak to the purpose of the simulation? I.e., If you regard the purpose of
   the simulation is a question, what is the answer?

Point Assignment:
1-2: They fail to make one or more of the above explicit
3: Adequate -- all of the above were fairly clear
4: Good -- all of the above were very clear
5-6: Very Good/Excellent -- all of the above were very clear and you were impressed by the
   cleverness of the manipulation and/or interpretation
d) Discussion -- 6 points
1) The discussion must address the following:
   Summarize the motivation for, general design of, and performance of the model. How well does it perform the task for which it was designed?
2) It's better still if it addresses one or more of these points:
   Why does it perform the way it does? What are its strengths and weaknesses? Are there additional simulations that it would be particularly interesting to run? What extensions could improve the model and in what ways? To what broader issues do the model and/or the results speak?

Point Assignment:
1-2: Inadequate -- The paper fails to clearly address even the points in (1)
3: Adequate -- The paper clearly address the points in (1)
4: Good -- The paper clearly address the points in (1) and some of the points in (2)
5-6: Very Good/Excellent -- The paper very clearly or very cleverly address the points in (1) and some of the points in (2). The discussion betrays a deep understanding of the model on the student's part.

e) References: (0 points, but you can deduct points for egregious errors)
   List the references for any papers, books, etc. using APA style.