CS 295 **Pattern Recognition** **Spring 2002**

Course Syllabus

Instructor: Robert R. Snapp, 353 Votey Bldg., 656–0735, `snapp@cs.uvm.edu`.

Office Hours: M 2:30–4 p.m., Th 1:30–3 p.m., F 11 a.m.–12 noon, and by appointment.

Lectures: MWF 1:25 – 2:15 a.m., in 367 Votey Bldg.

Web Page: `www.cs.uvm.edu/~snapp/cs295.html`

Description: Following a rigorous description of the statistical foundations of pattern classification, this course will survey a variety of statistical paradigms and popular pattern recognition algorithms. Topics will include maximum likelihood estimation, Bayesian parameter estimation, Parzen windows, hidden Markov models, linear discriminants, multilayer neural networks, radial-basis functions, support vector machines, decision trees, k nearest neighbor classifiers, and k-means clustering.

Textbooks: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*, Second Edition, John Wiley & Sons, New York, 2001. ISBN 0–471–05669–3 (*required*).

Prerequisites: Stat 251 or a desire to learn probability theory.

Grading Policy: The course grade is based on homework assignments (1/4), a take-home midterm (1/4), an independent term project (1/4), and a take-home final exam (1/4). Higher performance standards will be applied to graduate students.

Homework: A few written exercises will be assigned each week, including an occasional programming assignment. Homework will not be collected. Instead, we will review the solutions as a class, usually on Fridays. Your homework grade will be determined by the level and quality of your participation in these sessions.

Midterm Exam: There will be one take-home exam, that will be distributed on March 4, and collected in-class on March 8.

Final Exam: A take-home final exam will be distributed on April 29, and collected at 5 p.m., Friday, May 3, 2002.

All exams will be open book, and open notes; however, any reference materials that you employ should be cited in your work.

Course Projects: All students are required to complete a term project that consists of an independent investigation of an approved topic in pattern recognition. Each project contains three parts:

1. On March 1, 2002, a 2–3 page, typed proposal is due. The proposal should describe your topic and its relation to computer science; summarize what you have accomplished to date; describe what you intend to accomplish during the remaining seven weeks; and list at least *three* relevant books or published articles. If relevant, describe any software that you intend to create. (10%);

2. During the last two weeks of class (April 22 – May 1) you should present a 20 minute presentation about your project. (Please schedule your presentation date at least two weeks in advance.) (40%)

3. On Wednesday, May 1, a 10 page typed report is due. It should describe your project in detail. (The suggested page range is exclusive of code: if you wrote some original software, please include it in a separate appendix.) The report should be clearly written, and will be graded on originality, correctness (including spelling and grammar), effort, and clarity. (50%)

A broad range of topics are acceptable. Projects can be experimental, theoretical, or both. Here are a few ideas to get you started:

1. Train a neural network to classify one of the pattern sets stored in the University of Californine, Irvine, Machine Learning Repository [1]. See also the KDD Archive.[2] Study how the classification accuracy depends on the number of hidden units, or other architectural or algorithmic parameters.

2. Study the convergence properties of an unsupervised algorithm, e.g. the k-means algorithm. Apply the algorithm to some public high-dimensional data sets. How do your experiments agree with published theories?

3. Demonstrate that independent component analysis can be applied to solve the cocktail party problem, (also known as blind source separation).

4. Create a support vector classifier, and train it on some examples from the UCI Machine Learning Repository.

[1] http://www1.ics.uci.edu/˜mlearn/MLRepository.html
[2] http://kdd.ics.uci.edu/

5. Perform a literature search, and review, about current investigations of automatic face recognition. Alternatively, design and test a pattern classifier that can be used as a face identification system.
6. Study and describe the geometry of classification regions induced by the k-nearest neighbor classfier for several different metrics.
7. Can pattern recognition systems be used to detect commercials in television broadcasts, or to classify a musical recording by its style? What features are relevant for these kinds of problems? These questions can be answered by literature search and analysis, or by original experimentation.
8. A number of publications quantify the statistics of natural images (e.g., photographs of animals, landscapes, flowers, rocks, buildings, and office interiors). How can these results be applied to determine useful sets of features for image classification and analysis?
9. Investigate how hidden Markov models are used in speech recognition, and/or DNA sequence analysis. Construct a computer program that implements an HMM for one of these applications.
10. How do palm-top computers recognize handwritten characters? Implement your own version of an algorithm that recognizes Gregg shorthand and converts it into ASCII encoded text. (Wouldn't this be useful?)

Late Assignments: No late exams or projects will be accepted without a written and valid excuse.

Collaboration: Collaboration on the homework assignments is strongly encouraged. Collaboration on the take-home exams is not allowed; any suggestion of the latter will be treated as a major violation of the University's policy on academic honesty.

References

1. Martin Anthony and Peter L. Bartlett, *Neural Network Learning: Theoretical Foundations*, Cambridge University Press, Cambridge, U.K., 1999, ISBN 0–521–57353–X.
2. Christopher M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, 1995, ISBN 0198538642.
3. Luc Devroye, Lazlo Gyorfi, and Gabor Lugosi, *A Probabilistic Theory of Pattern Recognition*, Springer-Verlag, New York, 1996. ISBN 0387946187.
4. Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition*, Second Edition, Academic Press, 1990. ISBN 0122698517
5. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer-Verlag, New York, 2001, ISBN 0–387–95284–5.
6. Ian Nabney, *Netlab: Algorithms for Pattern Recognition*, Springer-Verlag, New York, 2001.
7. Brian D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, 1996. ISBN 0521460867.