Session Number____

Teaching Computational Intelligence for Engineers in the 21st Century via Distance Learning

Walker H. Land, Jr. [a] / **Dan McKee** [b] / **Douglas M. Reitz** [b]

[a] **Dept. Of Computer Science, Dept. of Bioengineering, Binghamton University, Binghamton NY , USA;**

[b] **Graduate Student, Dept. of Computer Science , Binghamton University, Binghamton NY , USA**

ABSTRACT

Engineers, computer scientists, medical personnel as well as some of the Liberal Arts disciplines will be using computers driven by intelligent software as a matter of course in 2020. This assertion is currently supported by the fact that computer aided diagnostic (CAD) intelligent software is currently being developed as a second opinion diagnostic aids for breast and lung cancer in addition to being used for diagnosing heart abnormalities and prostate cancer. In addition, intelligent computer detection and classification software has been and is being developed for the detection of deadly bio terrorism nerve agents, such as organophosphates.

The primary objective of this paper is to describe how distance learning is currently being employed to teach Support Vector Machine (SVM) concepts as a mechanism to develop these intelligent software packages. Specifically, this paper contains the following topics: Specifically, this paper addresses the following topics (1.) why have students do research projects on the computer aided diagnosis of breast cancer, (2.) some background on the SVM / GRNN oracle hybrid used by the students in the research work reported on here as well as a brief description of the screen film mammogram data base, (3.) Example and description of a student research project using this hybrid as well as the reporting requirements, (4.) Student (Douglas M. Reitz) research report and results for this project, and (5.) 25minute video of students presenting the results of their research

This presentation also contains a 25min movie titled “Topics in Computational Intelligence” that demonstrates the proficiency with which graduate and upper level undergraduate students’ learn this CAD technology in the distance learning environment, by reporting on research projects in which they developed a CAD tool to diagnose breast cancer from a mammogram screen film data set using SVM technology. This film was first shown as part of an invited paper on teaching Evolutionary Computational Concepts at the 2002 International World Congress for Computational Intelligence.

Keywords: Benefits of Distance Learning, Support Vector Machines, Generalized Neural Network Oracle, Student Projects and presentations

1. INTRODUCTION

This paper describes how local and distance learning can prepare students to use computational intelligence (CI) technology and thereby enable them to write intelligent software packages which have not only the ability to adapt to but also to learn from their environment (i.e. data base from which information is extracted). Consequently, by teaching this neural networks course, we are taking the first step in preparing the student to not only develop CI technology but also work with it.

Each time this course is taught, different aspects of CI technology are covered. For example, we have covered: families of neural networks derived by Evolutionary Programming (EP) technologies, Probabilistic Neural Networks (PNN) and/ or General Regression Neural Network (GRNN) trained by Differential Evolution (DE), EP / PNN hybrids, Support Vector Machines (SVM) trained by various numerical methods, the SVM / GRNN oracle trained by DE, integrating principal components analysis with the PNN to reduce the dimensionality of a the environment and thereby reduce the computational load, and many other CI and combinations of other CI topics. In general, the course contains: a mid term, oral or written final and two projects. (For more complex research topics, one project with two parts will suffice). Projects are generally selected from one of the three categories described: (1.)computer aided breast cancer diagnosis, (2.) computer classification of organophosphate nerve agents, (3.) research project a graduate student may select and approved by the instructor.

Specifically, and by way of example, this paper will focus on a specific semester and some of the material covered during that semester. This local and distance learning course covered the following topics for the 2003 fall semester: (1) Support Vector Machines, (2.) Generalized Regression Neural Network (GRNN), and (3) The SVM / GRNN oracle hybrid and Differential Evolution (DE) training methods. We will cover in some detail a research group project which uses the SVM /GRNN oracle hybrid to perform the computer aided diagnosis of breast cancer and measure the results of that hybrids diagnostic performance.

The following is an example a project assignment used in the Fall of 2003 for this cross-listed Graduate / Upper Level Undergraduate Neural Networks Course. Graduate offsite distance learning students were also included in this course. The onsite students worked in small groups while the offsite distance learning students worked individually. This is generally the paradigm which I use these types of courses. However, in rare cases two offsite students will work together, when they are in the same geographical location even though they may work for different companies. The remainder of the paper addresses the following topics: (1.) why have students do research projects on the computer aided diagnosis of breast cancer, (2.) some background the SVM / GRNN oracle hybrid used by the students in the research work reported on here as well as a brief description of the screen film mammogram data base, (3.) Example and description of a student research project using this hybrid as well as the reporting requirements, (4.) Student (Douglas M. Reitz) research report and results for this project, and (5.) 25minute video of students presenting the results of their research

(2.) Why Student Research Projects on Breast Cancer?

Why have the students work on computer aided diagnostic brest cancer reserch for their projects? In addition to giving them practical experience in dealing with a current problem, carcinoma of the breast is second only to lung cancer as a tumor-related cause of death in women. In 1993 it was reported that 180,000 new cases and 45,000 deaths would occur just in the US [1]. It has been proposed, however, that mortality from breast cancer could be decreased by up to 25% if all women in appropriate age groups were regularly screened [2]. Currently, the method of choice for the early detection of breast cancer is mammography, due to its widespread availability, low cost, speed, and non-invasiveness. At the same time, while mammography is sensitive to the detection of nonpalpable breast lesions, it has a low positive predictive value (PPV) for breast cancer, resulting in costly and invasive biopsies that are only 15%-34% likely to actually show malignancy at histologic examination [3] .

The reader may be interested to know that several applications of computational intelligence to breast cancer screening have been done. For example, earlier work by Floyd et al. [4] and Lo et al. [5] demostrated the applicability of backpropagation-trained Multiple Layer Feedforward Neural Networks (MLFNs) to the task of classifying lesions. EP-based feedforward networks have been applied by Fogel et al. [6],[7],[8][9] and Land et al. [10]. They were able to achieve similar or better results than those trained by backpropagation, but with much simpler architectures (i.e., fewer nodes in the hidden layer). Evolutionary-Programming (EP) based linear discriminate analysis, [11] and an adaptive boosting/EP hybrid [12] also have been investigated as potential classification mechanisms. Referecce [13] describes the extension of previous research. Specifically, this paper explores the use of two novel approaches called Partial Least Squares (PLS) and Kernel-PLS (K-PLS) to the diagnosis of breast cancer. The approach is based on optimization for the partial least squares (PLS) algorithm for linear regression and the K-PLS algorithm for nonlinear regression.

(3.) Background Supporting CAD Breast Cancer Benign / Malignant Paradigm and Mammogram Data Base Information

This section contains a brief summary of the the GRNN oracle as well as the screen film mammogram data base used by the students to develop the cancer diagnostic accuracy results. An overview of the GRNN-based oracle is presented in Figure 1.

Figure1. GRNN Oracle

Suppose there are two or more prediction models of any type, each of which predicts the same scalar output variable. (Extension to multiple outputs is easy, but in practice one never wants to do this. If there are multiple outputs, a separate oracle should be used for each.) Neither the nature of these models nor their inputs is important. In fact, the model inputs do not enter into this discussion in any way. One or more *gate variables*, whose values presumably have an effect on deciding which of the competing models is most valid, are important. For present purposes, the individual prediction models also are not important. They are assumed to be black boxes that work reasonably well.

(3.1) Mammogram Data Base Findings

The mammogram data set (USF DDSM) used in this research contains 1,979 screening Mammograms obtained from 1988 to 1999 from the following four hospitals: Massachusetts General Hospital, Wake Forest University School of Medicine, Sacred Heart Hospital, and Washington University in St. Louis School of Medicine. This data set contains six BI-RADS™ indicators (extracted from screen film images) and patient age (Table 1). Of the 1,979 cases, 994 (~50%) were benign. This data set was integrated with another independently developed data set from Duke University containing 500 cases, of which approximately 2/3were benign and 1/3 were malignant. Consequently, a data set of approximately 2500 samples was used in the analysis.

Variable	Category
Calcification Distribution Calcification Description	Mammographic Calcification Findings
Mass Margin Mass Shape	Mammographic Mass Findings
Associated Findings Special Cases	Mammographic Miscellaneous
Age	Clinical History Var

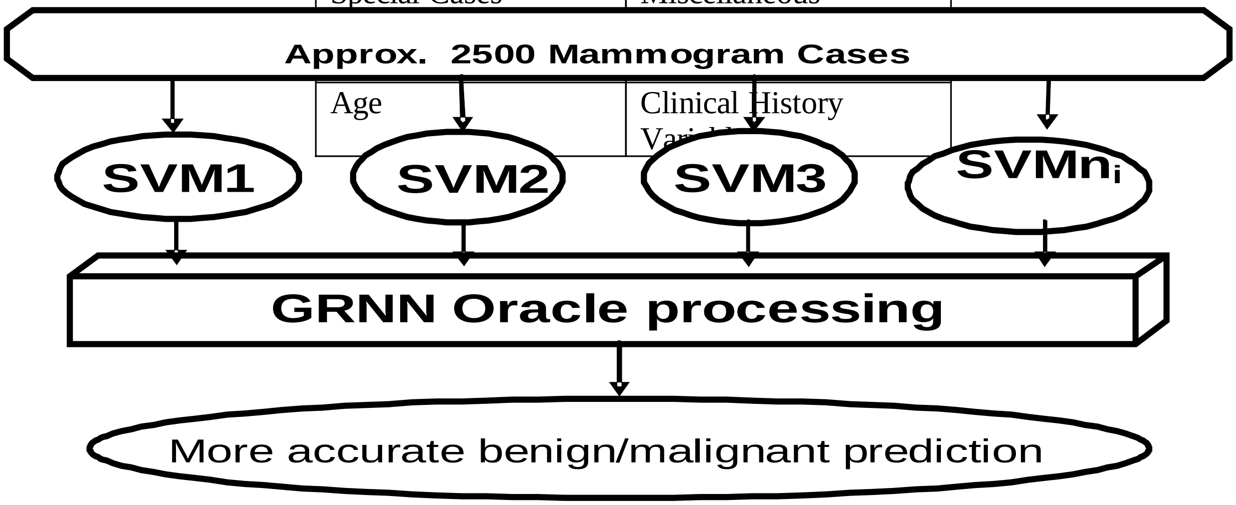

Approx. 2500 Mammogram Cases
SVM1
SVM2
SVM3
SVMn$_i$
GRNN Oracle processing
More accurate benign/malignant prediction

Table 1. Combined USF DDSM and Duke University Reduced Data Set Components

(4.) Example and description of Project Assignment

The objective of this project assignment was to first synthesize a fairly complex hybrid Support Vector Machine (SVM) / Generalize Regression Neural Network (GRNN) Oracle software package (SVM / GRNN oracle), written in object oriented C++. The students were then to use this package, with appropriate modifications and additions, to establish the computer aided diagnostic (CAD) accuracy of this software package in the diagnosis of breast cancer using a combined data set of approximately 2500 samples for six separate hospitals and cancer research institutions. Both the graduates and the undergraduates worked on the same project. However, the graduate students had more difficult and additional assignments, depicted in *italics* as described in the following section.

(4.1)Project Title: CS562 GRNN Oracle Project II for Fall 2003 Synthesis and Analysis of SVM / GRNN oracle (This is an example of an assigned project.)

Assignment date: November 3, 2003

(4.1.2) Objectives: The objectives of this project (for both graduate and upper level undergraduate students) ,as described in the detailed instructions below, are to synthesize the existing software package regarding its structure and organization and to use existing software package SVM kernels to establish the CAD benign / malignant accuracy,. The graduate onsite and offsite distance learning students were required not only to understand the theory of new spline kernels but also to up grade the CAD package by implementing and evaluating the CAD performance of these kernels. Specifically, the assigned objectives were:

- (a) synthesize the integrated SVM / GRNN oracle design, (which contains the integration, training and implementation of the theory described in class) and document the results of its use.

- (b) use this integrated package to compute the outputs different Support Vector Machine kernel designs against one another with the objective of improving the overall Az index, the partial AZ, as well as the specificities and PPVs at 95%,96%,97% and 100% sensitivities.

- (c) *Graduate and offsite students only*: And (c) implements the SVM spline kernel(s) (as discussed in class and in the material posted on the class web page) and use the outputs of these kernels as one of the inputs to the GRNN oracle.

(4.1.3) Documentation Requirements and Report Outline

This section summarizes the topics and content that both the graduate and upper level undergraduate students are required to develop to successfully complete this project.

For objective (a) above, describe and document in your report:

1. the existing structure chart describing how the program is organized
2. the Differential Evolution (DE) method used for Oracle training.
3. the specific GRNN oracle design and operation using selected gate variables (how experiments were designed). {Hint you may want to try : (a) age as a gate variable, (b) mass margin as a gate variable, and (c) mass margin and age together as gate variables.}.I also would suggest you use other gate variables from the reduced set of BIRADS indicators..
4. how the program is used ,with examples and include a users manual (as an appendix).
5. describe how to use the graphical packages.

For objective (b) above, describe and document in your report:

1. How you arrived at the DE population size and number of generations used
2. How the k and α values were selected
3. How critical is the population size and how the population size was selected
4. Plot histograms of all MOPs for all critical experiments as well as plot all the ROC Az and partial ROC Az curves for all critical experiments
5. Any other pertinent information required to understand, use and explain program results.

Graduate and offsite graduate students only complete this section to satisfy objective © above.

For objective above (c) describe and document in your report (graduate and offsite students only):

1. How the spline kernel(s) were implemented (did you use both second and third order splines.
2. How and why you chose the knots and intervals
3. Include documentation for all code written to implement the splines.
4. (*Optional and for extra credit*) Implement and describe how you implemented the Cheney and Kincaid method of computing cubic splines using the Langrangian Interpolating polynomials.

Finally all students should include detailed users manual that delineate how to run the program with examples as well as a CD that contains a soft copy of the documented report, presentation, C++ source code, object and executable files.

(4.1.4) Report Content

The report should be written to the following table of contents**,** which outlines how the results of the research should be presented. Observe that the graduate students are required to complete a separate research section on the use of spline kernels.

Required Report Table of Contents

1. Numbered Table of Contents
2. Summary of Report Content
3. Part a. Focuses on overall system design and GRNN oracle

3.1 Explain Structure Chart (graduate students will have modified structure chart to include spline work)

3.2 Summary of DE used for oracle training (here you *summarize* the differential evolution *process*, independent of a your specific application)

3.3 How you used GRNN oracle (including graphical packages) , selected gate variables and results. Big section here as it includes items 3 and 5. (Users, item 4, manual described in the appendix)

4. Focuses on specifics of training the oracle using differential evolution, the measures of performance used and performance results.

4.1 How experimentally did you select the DE population size and number of generations used.

4.2 How did you experimentally select the k and α values

4.3 How critical is the population size to system performance?

4.4 Section which contains tables and histograms of systems performance as well as figures of ROC Az and partial Az indices.

5. (*Graduate and offsite students only complete this section*) Spline Kernel and GRNN oracle performance

5.1 Which spline kernels did you implement (1st order, 2nd order and/or 3rd order)

5.2 How many knots used and why?

5.3 Documentation of spline source code

5.4 (*Optional and for extra credit*) Implementation of the Cheney and Kincaid method of computing cubic splines using Langrangian interpolation polynomials

6. Conclusions

7. Recommendations

8. Appendices

8.1 Users manual with examples

8.2 Soft copy, which includes documented report and presentation, source code, object and executable files.

(4.1.5) Presentation Outline

All onsite student groups were required to give a formal presentation of at least 20 minutes. However, some of these presentations have lasted for an hour, their length depending upon the originality of their work as well as the new results which some of the groups sometimes develop.

Content of your presentation

- Structure chart of system (for graduate and offsite students revised system including splines)
- Summary of DE method of training GRNN oracle
- Summary of GRNN oracle performance with SVM kernel inputs (graduate and offsite students include spline kernels also). Include performance of individual SVM kernels for comparison
- Conclusions
- Recommendations

(5.) Example Project Report Results from an Offsite Distance Learning Student Douglas M. Reitz

A complete report for this project cannot be included in this paper bacause of its length. However, this section does some contain report content abstracted from the project report of an offsite distanct learning student named Douglas M. Reitz. Only one distance learning final project report is illustrated for consistancy in presentation. Exerps are repeated here because reports, particurally of graduate and off site distance learning studetnts, sometimes exceed 100 pages. The information in the remainder of section 5 is abstracted directly from Doug's report [14].

CS562 GRNN Oracle Project II for Fall 2003
Synthesis and Analysis of SVM/ GRNN Oracle
Student: Douglas M. Reitz
Date: 01/17/2004

SUMMARY OF REPORT CONTENT

This report describes the results of Project 2. Project consisted of inputting the SVM outputs provided from Project 1 (mixed Duke USF data input set) into the GRNN oracle and comparing the results. The project 1 SVM outputs used here are the ones achieved using the Add, Mult, S2000, and Spline kernels. The GRNN results are presented using the following combinations of gate variables: age; mass margin (MM); age and MM; and all 7 BIRADS indicators.

For all experiments Differential Evolution (DE) was used to train the GRNN. Section 3.2 contains a summary of the DE process independent of the software application.

In summary the experiments here showed that the GRNN with all of the gate variable combinations performed better than any of the individual SVM kernels alone when considering performance at or below 98% sensitivity. The GRNN performance exceeded the individual SVM kernels in the following measures of performance Az, Az90, and spec & ppv at 98% sensitivity and below. The spec 98% performance improved significantly (7-8%) over any of the SVM kernels alone. When considering performance at 100% sensitivity the GRNN results fell below that of some of the individual SVM results.

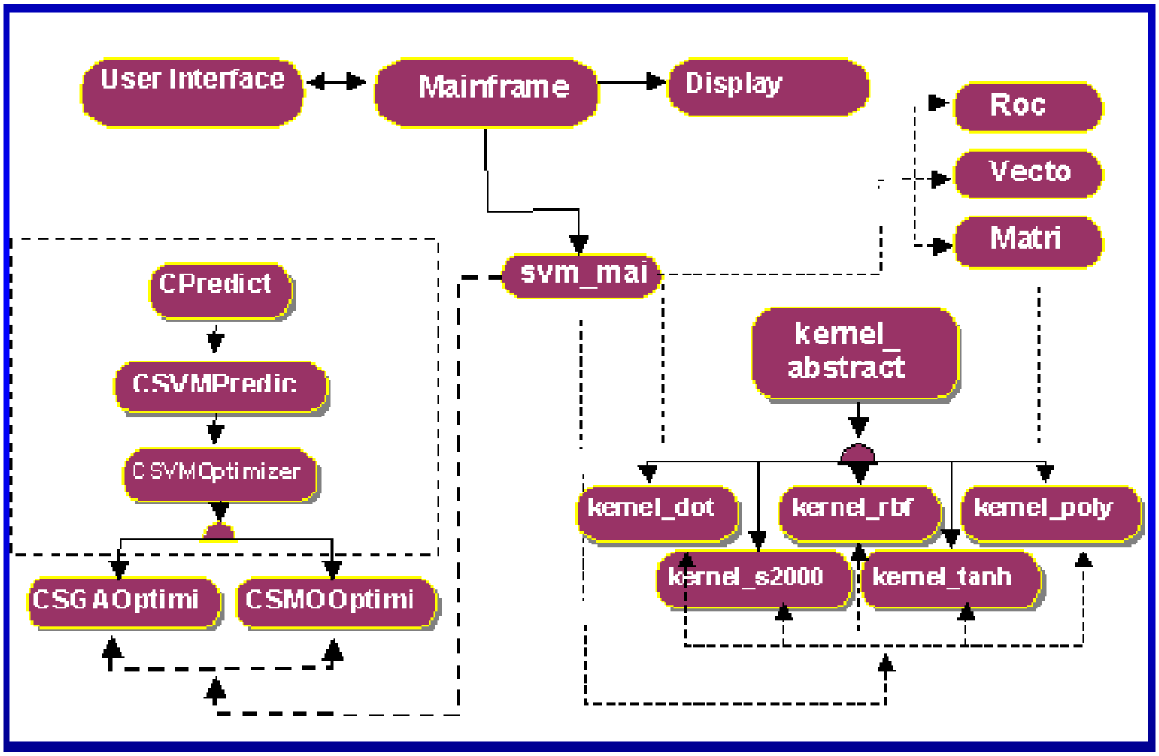
The selection of gate variables and CR for the GRNN had little effect on the results. However, one surprising result was that the best GRNN performance over all the MOPs was with all 7 indicators used as gate variables and CR=0.9.

The remainder of the report provides some background and describes in detail how the experiments were conducted and the results obtained.

OVERVIEW DESCRIPTION OF GRNN ORACLE SOFTWARE PACKAGE SYSTEM DESIGN

The GRNN software package was used and no changes were made to the provided GRNN software. The EP_SVM_GUI software is structured as shown and described below.

STRUCTURE CHART



Picture taken from Web Page presentation

As described in the Project1 report, 3 classes were added: kernel_add, kernel_mult, and kernel_spline. All three of them were implementations of kernel_abstract just like the existing kernels shown above (kernel_dot, kernel_rbf, kernel_poly, etc.). In like manner a new class kernel_nspline3 was allocated for implementation of a natural 3rd order spline kernel. The development of this kernel was not completed and is therefore not mentioned further in this report.

SUMMARY OF THE DIFFERENTIAL EVOLUTION (DE) PROCESS USED FOR ORACLE TRAINING

As described in the lecture notes the DE process is summarized as follows:

1. The population is initialized randomly.
2. The primary parent (P1) is deterministically chosen.
3. A second parent (P2) is randomly chosen.
4. The first differential (d1) is chosen.
5. The second differential (d2) is chosen.

6. The second parent is mutated as follows to produce a p2’: p2’ = p2 + K(d1-d2) where K is a predetermined mutation factor.
7. A random number (alpha) is generated.
8. If alpha is less than a predetermined crossover constant (CR) p2’ could replace p1.
9. If the fitness of p2’ exceeds p1, keep p2’. Otherwise replace p1 with p2’.

HOW THE GRNN ORACLE WAS USED FOR THIS PROJECT

The add, mult, spline, and s2000 kernel output results from Project 1 were used as inputs into the GRNN. The GRNN was executed with 2 different crossover constants (0.6 and 0.9) and the results of the GRNN output were created.

The first step in this process was to create appropriate inputs for the GRNN oracle from Project 1. This required re-executing the EP_SVM_GUI as detailed in Project 1 to generate the prediction file results for each of the kernels used in this project. The only change from the Project 1 procedures was to select the generation of prediction files. Once the prediction files were generated, a new input file was created for each of the gate variable scenarios. The files are created as described in section 8.1.1. In this project the following gate variable combinations were used (age, MM, age & MM, all 7 BIRADS indicators). Please note that the 1st and 2nd BIRADS indicators (columns) in the provided data set are being considered age and MM respectively. In the event that those columns do not represent those indicators, the term age should be taken to mean the indicator in the first column, and the term MM should be taken to mean the indicator in the second column. The age, MM, and age & MM gate variables were chosen based on the suggestion included in the assignment. Based on the experience of others, these two indicators work well when chosen as gate variables. The last combination chosen was to use all of the 7 indicators in the dataset as gate variables for comparison.

Once the input files were created, the EP_SVM_GUI software package was used to generate the 5 fold training and validation files to be used for input into the GRNN. During this process it was discovered that the provided EP_SVM_GUI software was not able to produce the 5fold files. After spending time debugging this issue, the EP_SVM_GUI software was modified to correctly generate these files. All of the modified EP_SVM_GUI software is included and/or its location is specified in section 8.2. The description of how the input files are generated is included in section 8.1.2. For this project the following input files were created:

1st_gate_	files for age gate variable
2nd_gate_	files for MM gate variable
1st_and_2nd_gate	files for age and MM gate variables
all_gate	files for all of the 7 BIRADS indicators as gate variables

Using the input files, the GRNN was executed for each of the 5 folds for each of the gate variable combinations with crossover constants of 0.9 and 0.6. The output of executing the GRNN was report files (.rep) for each fold and gate variable/crossover constant combination. Section 8.1.3 describes how these .rep files were generated in detail. For all of the experiments the mutation factor, population size, and number of generations were each set to a constant value.

Once the report (.rep) files were generated for each fold, gate variable, crossover constant combination, the corresponding .az files were generated using the roc_gui feature in the GRNN software.

These files which are the final output of the GRNN software are used to evaluate the results of all the experiments.

TRAINING THE GRNN AND SYSTEM PERFORMANCE

SELECTING POPULATION SIZE AND NUMBER OF GENERATIONS

A population size of 30 was used for all of the experiments in this project. This population size was determined by the recommendations of the previous Lut et. al project. Per the GRNN implementation comments from the lecture notes, the population size should be 5 to 10 times the number of degrees of freedom in the input vector. For these experiments a size of 30 was deemed to adequately meet these recommendations/requirements.

Ten generations were used for all of the experiments in this project. Increasing the number of generations increases the number of levels through which the differential evolution process can identify a more accurate solution. However as the number of generations increases the computational time required to execute the GRNN experiments increases. Ten was chosen as a balance between these two factors.

SELECTING K AND SIGMA (ALPHA)

The mutation factor (K) for all of the experiments in the project was set to 0.5. This factor determines the amount by which the second parent is perturbed when creating p2'. For a summary of how K affects the DE process see the above description. A value of 0.5 was chosen based on the implementation comments in the lecture notes and previous projects. Studies and experiments with adjusting this parameter were not conducted in this project.

A sigma range of 0-25 was chosen for all of the experiments in this project. This value is the range in which the GRRN can scale the input parameters. This range was selected based on previous experiments.

POPULATION SIZE CRITICALITY TO SYSTEM PERFORMANCE

The larger the population size the more accurate results can be expected. However, there are diminishing returns as the computational time increases. Each population member is used to represent the distribution of input samples. Therefore if this number is too low, GRNN results will be less accurate.

Results for different SVM Kernels

Average Combined Results Table for SVMs Used As Inputs to GRNN

	S2000	Add	Mult	Spline
Az	**0.8170388**	0.810415	0.813558	0.7966992
Az90	0.3010476	0.3014496	**0.3138928**	0.2746192
100% spec	0.03802296	**0.07293676**	0.00847991	0.04351522
100% ppv	0.4775874	**0.4867668**	0.469693	0.4790228
98% spec	0.16184116	**0.1745396**	0.15830644	0.16735224
98% ppv	0.5083938	**0.5114496**	0.5067704	0.5092162
97% spec	0.2484878	0.2412362	**0.2523292**	0.2327052
97% ppv	0.532105	0.5301294	**0.5340502**	0.5272194
95% spec	0.3407654	0.3447238	**0.3575432**	0.2988802
95% ppv	0.5596056	0.5610136	**0.5660278**	0.544725

New Topic: SPLINE KERNEL AND GRNN ORACLE PERFORMANCE

SPLINE KERNEL IMPLEMENTED

The spline kernel implemented for this project was the 1st order spline function. This kernel was added to the EP_SVM_GUI software package as described in the project 1 report.

Note: There was also a significant amount of time spent on trying to come up with a design that would incorporate a natural 3rd order spline function into the EP_SVM_GUI software. Two approaches were attempted. However neither was successful. The author of this project was not able to come up with a design that would allow inserting a natural cubic spline into the EP_SVM_GUI without redesigning and rewriting significant portions of the EP_SVM_GUI software. The biggest item that could not be reconciled was figuring out how to determine knots and therefore the spline kernel constants for each interval, while the kernel function was only seeing two input vectors at a time. Nspline3 is a non-operational partially implemented (basic structure only) class that is a remnant of this effort.

SPLINE KERNEL KNOTS

The 1st order spline function ports very well into the existing EP_SVM_GUI software and only requires 2 knots for implementation. These equated to being the u and v input vectors used in the current EP_SVM_GUI design.

SPLINE KERNEL SOURCE CODE DOCUMENTATION

The 1st order spline function was implemented as a new class (kernel_spline) in the EP_SVM_GUI software. Like the other kernels in the EP_SVM_GUI, it is an implementation of the kernel_abstract class. The significant change from the other kernels is its calcK function which was taken from equation 10 on page 9 of Steve R. Gunn and J.S. Kandola's paper entitled 'Structural Modelling with Sparse Kernels'. The equation as implemented is kspline(u,v)=uv + 1/2(u+v)min(u,v) - 1/6(min(u,v))^3

DISCUSSION OF CHENEY AND KINCAID METHOD OF COMPUTING CUBIC SPLINES USING LAGRANGIAN INTERPOLATION POLYNOMIALS

The Cheney and Kincaid method of computing cubic splines using Lagrangian interpolation polynomials is an alternative to solving a large number (4n) of simultaneous equations. With the Cheney and Kincaid method the system of equations can be reduced to n-1 thereby, greatly reducing the computational cost. This is accomplished by requiring the 2nd order derivatives at the end knots to be zero, making this a natural spline. As a result this reduces the number of unknowns. The method also utilizes the fact that the 2nd derivatives at the knots must be connected by a straight. With these restraints and observations the number of unknowns is reduced to solving n-1 equations.

Douglas M. Reitz Project II CONCLUSIONS

Use of the GRNN Oracle using EP_SVM_GUI and the parameter inputs specified in this report offers some improvement over using the EP_SVM_GUI results alone when sensitivities at or below 98% are desired. Performance improvement at the 97% sensitivity was very striking with the GRNN spec results reaching 0.3 and exceeding the best input SVM results by approximately .05. GRNN performance improvement at 98% was also significant with spec reaching .24-.25 and exceeding the best input SVM performance by ~.07.

If sensitivities above 98% are desired, use of the GRNN with these parameters offers no improvement over the input Add SVM kernel. However, it did perform better than some of the input SVM kernels and may be useful for a more “balanced” result. The results here were closer to the Add SVM kernel and avoided the low performance of the Mult kernel.

Douglas M. Reitz Project II RECOMMENDATIONS

In light of the surprising performance of the GRNN using CR=0.9 and all 7 indicators as gate variables. I would recommend conducting further experiments with these same inputs and parameters using larger population sizes and number of generations.

The GRNN results here also show that the GRNN is able to combine the individual SVM results into a solution that able to produce results above those of all of the input SVM kernels when sensitivities at or below 98% are acceptable. When operating at higher sensitivities the GRNN was able to smooth the differences between the SVMs and provide results between the worst SVM result and the best SVM result. Therefore, I would recommend that it be considered further for use as a diagnostic tool in this application.

(6.) 25 Minnute Video which accompanies this paper

This paper also has a 25 minnute video which demonstrates both graduate and upper level undergraduate students preesenting their Support Vector machine Project Reports**.**

(7.) Conclusions

This paper has demonstrated how a computational intelligence technology, specifically, a SVM / GRNN oracle hybrid, is taught to prepare students to work with not noly intelligent processing but also intelligent software packages in both local and distance learning enviroments. The specific example chosen to demonstrate this distance learning capability was the computer aided benign / malignant diagnosis of breast cancer using this hybrid. Using a data set of 2500 samples from six separate hospitals and cancer research instutions, it was shown that this hybrid provide accurate diagnostic results as measured by specificities and

Positive Predictive Values at the clinically relevant higher sensitivities. (See table of results in section 5 for specific values for different SVM kernels).

Bibliography

1. Boring, C.C., T. S. Squires, and T. Tong. (1993) Cancer Statistics. *Cancer Journal for Clinicians*, 43:7-26.

2. Strax, P. (1989) re that You Do Not Have Breast Cancer. St. Martin's, NY.

3. Lo, J. Y., J. A. Baker, P. J. Kornguth, J. D. Iglehart, and C. E. Floyd. (1997) Predicting Breast Cancer Invasion with Artificial Neural Networks on the Basis of Mammographic Features. *Radiology*, 203:159-163.

4. Floyd, C.E., J.Y. Lo, A.J. Yun, D.C. Sullivan, and P.J. Kornguth. (1994) Prediction of Breast Cancer Malignancy Using an Artificial Neural Network. *Cancer*, 74:2944-98.

5. Lo, J. Y., J. A. Baker, P. J. Kornguth, and C. E. Floyd, Jr. (1999) Effect of Patient History Data on the Prediction of Breast Cancer from Mammographic Findings with Artificial Neural Networks. *Acad. Radiology*, 6:10-15.

6. Fogel, D.B. (1995) Evolutionary Computation: Toward a New Philosophy of Machine Intelligence. IEEE Press, Piscataway, NJ.

7. Fogel, D.B., E.C. Wasson, and E.M. Boughton. (1995) Evolving Neural Networks for Detecting Breast Cancer. *Cancer Letters*. 96:49-53.

8. Fogel, D.B., E.C. Wasson, E.M. Boughton, and V.W. Porto. (1997) A Step Toward Computer-Assisted Mammography using Evolutionary Programming and Neural Networks. *Cancer Letters*, 119:93-97.

9. Fogel, D.B., E.C. Wasson, E.M. Boughton, and V.W. Porto. (1998) Evolving Artificial Neural Networks for Screening Features from Mammograms. *Artificial Intelligence in Medicine*. 14:317-326.

10. Land, W. H., T. Masters, J. Y. Lo, and D. McKee. (2000) Using Evolutionary Computation to Develop Neural Network Breast Cancer Benign/Malignant Classification Models. *4th World Conference on Systemics, Cybernetics and Informatics*, 10:343-7.

11. Fogel, D.B., E.C. Wasson, E.M. Boughton, V.W. Porto, and P.J. Angeline. (1998a) Linear and Neural Models for Classifying Breast Masses. *IEEE Trans. Medical Imaging*, 17(3):485-488.

12. Land, W. H., Jr., T. Masters, and J. Y. Lo. (2000a) Application of a New Evolutionary Programming/ Adaptive Boosting Hybrid to Breast Cancer Diagnosis. *IEEE Congress on Evolutionary Computation Proceedings*

13. Walker H. Land, Jr. , John Heine , Mark Embrechts ,Tom Smith, Robert Choma, and Lut Wong, "New Approach to Breast Cancer CAD Using Partial Least Squares (PLS) and Kernel-PLS (K-PLS)", MI 2005, (In Press)

14. Reitz, D.M., "CS562 GRNN Oracle Project II for Fall 2003, Synthesis and Analysis of SVM / GRNN Oracle", 1/17/2004.