Teaching Pattern Recognition: A Multidisciplinary Experience

Shadnaz Asgari and Burkhard Englert
California State University, Long Beach

Abstract
The solution to many open problems in science and engineering requires approaches that are multidisciplinary in nature. Therefore, state-of-the-art education needs to prepare prospective scientists and engineers to not only explore the boundaries within their own disciplines, but to also understand the basics of other disciplines. Accomplishing this important mission requires careful planning, selection of appropriate topics, and development of realistic educational objectives to promote cooperation and integration between students with various backgrounds.

Aiming for such a goal, in the spring of 2013, a graduate level course on "pattern recognition" was piloted in the Computer Engineering and Computer Science department of California State University, Long Beach. The course was offered under the name "CECS 590-Special Topics in Computer Science" and several graduate students from various backgrounds (Biology, Mathematics and Computer Science) were enrolled. Throughout the semester, students learned about different machine learning techniques and algorithms, and implemented multidisciplinary projects which required the application of those methods in order to solve real-world problems in biomedical science.

In this paper we present the results of our pilot offering. We provide details about the course objectives, structure, assessment tools and outcomes. We also discuss some of the challenges confronted in teaching such a multidisciplinary class and the approaches undertaken to address those issues. Hence, the presented results of our pilot offering could be of value to other multidisciplinary educators.

Introduction

In the era of global competition to create innovative value, multidisciplinary collaboration has become an essential element for the successful delivery of a product. Hence, employment of multidisciplinary teams; consisting of members with different professional backgrounds working together towards a common goal, has become an industry trend[1] . This recently adopted trajectory accurately reflects the realities of the twenty-first-century: any sustainable solution to the problems humanity is currently facing requires an integrated and interactive mix of sciences, engineering, social sciences, and humanities[2]. As a result, modern education needs to prepare future scientists and engineers to not only explore the boundaries within their own disciplines, but to also understand the basics of other fields.

The Accreditation Board of Engineering and Technology (ABET) acknowledges the importance of multidisciplinary education and explicitly supports it[3]. In fact, the 2013-2014 criteria for accrediting Engineering programs requires the programs to insure that prospective engineers hold "an ability to function on multi-disciplinary teams" [4]. One plausible approach to accomplish this mission is through the curricular concept of integrating or connecting school subject areas. The integrative or multidisciplinary curricular approach related to technology education seeks to help students learn and appreciate the relevancy of how school subjects are tied together and how each subject builds on the other[5]. This can be achieved by designing multi-disciplinary courses where students are exposed to selected key theories and techniques of various scientific disciplines that are jointly applied to a field of study or professional practice.

Although the importance of multi-disciplinary education is well acknowledged and its integration into traditional education is often sought, significant barriers exist for the execution of supporting activities. Examples of such impediments may include increases in teaching loads, curricular restrictions, the need to attract course participants from outside the sponsoring department, the need for the course instructor to possess some level of expertise in more than one field of study, the difficulty of instructing a heterogeneous student body, prejudice and time constraints[3,6,7]. The administrative and cultural barriers to multidisciplinary education can be ameliorated with proper planning, while the instructional barriers can be addressed through careful selection of appropriate topics, and development of realistic educational objectives that promote cooperation and integration among students with various backgrounds. A pilot offering of a multidisciplinary course provides an excellent opportunity to identify additional relevant challenges and find appropriate resolutions to address those issues.

The current paper presents the result of such a pilot offering at Computer Engineering and Computer Science Department of California State University, Long Beach (CSULB). In the spring 2013 semester, a graduate level course on "pattern recognition" was offered under the name "CECS 590-Special Topics in Computer Science" and several graduate students from various backgrounds were enrolled. Over the semester, students learned about different machine learning techniques and algorithms, and implemented data analysis projects with an emphasis on biomedical applications. The rest of this paper is organized as follows. In Section 2 we discuss the details of the course offering including course objectives, structure and assessment tools. Section 3 discusses our pilot evaluation and its results. Finally, in Section 4 we conclude the paper with a brief summary of confronted challenges and the approaches undertaken to address them.

Teaching Pattern Recognition

Pattern Recognition is a branch of machine learning, one of the fastest growing multidisciplinary fields at the intersection of computer science and mathematics, and concerns the construction and study of systems that can learn from data. A core objective of a learner is to generalize knowledge gained from the experience[8]. Generalization in this context is the ability of a learning machine to perform correctly on new, unseen data samples or tasks[9]. For example, a pattern recognition system could be trained on email messages to learn distinguishing between spam and non-spam messages. After learning, it can then be used to classify new email messages into spam and non-spam folders[10]. Pattern Recognition has a broad range of applications spanning from computational neuroscience and medical diagnosis to stock market analysis and consumer behavior prediction.

A major objective of biomedical pattern recognition is discovering the patterns in the biomedical data that are associated with the onset and/or prognosis of a specific disease (identification of biomarkers). As the size, complexity, and variety of data sets resulting from biomedical research continue to increase at an exponential rate, so does the need to educate future biomedical scientists and engineers to learn about sophisticated algorithms and techniques which could be applied to automatically identify the biomarkers. In response to this need, we piloted a 3-unit graduate level course on "pattern recognition" with emphasis on biomedical applications. In what follows, we provide a detailed description of the organization of the course.

Course objectives, pre-requisites and textbooks. For our “Pattern Recognition” class, the course objectives were defined as

- To understand and apply methods for preprocessing, feature extraction, and feature selection to multivariate data.
- To develop prototype pattern recognition algorithms that can be used to study algorithm behavior and performance against real-world multivariate data.

The course pre-requisites were set as

- Basic familiarity with probability and statistics
- Basic familiarity with linear algebra
- Basic familiarity with computer programming

No textbook was mandated, as the instructor planned to cover different topics from different sources. But the following books were recommended:

- R. Duda, P. Hart, D. Stork, "Pattern Classification", second edition, 2000.
- T. Hastie, R. Tibshurani, and J.H. Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer Series in Statistics, 2001.

Recruitment of a heterogeneous student body. To recruit a diverse body of students, in addition to the Computer Science department, the course offering was advertised in the Mathematics department and the Biology department by sending a prepared course flyer to the Mathematics and Biology faculty members who were involved in multidisciplinary research and by asking them to share it with the potentially interested students. The CECS 590 course requires instructor consent for registration. Based on the pre-requisite requirements, the instructor (first author) authorized the registration of seven students (four students from computer science, two students from mathematics and one student from biology) and the auditing of another student (from biology).

Course topics. Aiming at balancing theory and practice, the instructor covered the mathematical as well as the heuristic aspects of various pattern recognition techniques and algorithms throughout the semester. The topics covered in the course were:

- Decision Theory (Likelihood Ratio, Naïve Bayes Classification, Belief Networks)
- Maximum-Likelihood and Bayesian Parameter Estimation
- Dimension Reduction (Principal Component Analysis, Linear Discriminant Analysis)
- Dynamic Bayesian Network (Hidden Markov Model)
- Nonparametric Techniques (Probabilistic Neural Network, Nearest Neighbor Classification)
- Linear Discriminant Functions (Perceptron, Linear Programing)

Course learning assessment. Homework was designed for each topic to insure that the students learn the theory. Quizzes were taken every three weeks to assess the students’ learning of the materials throughout the semester. Our course had one paper-and pencil midterm and a final exam. The exams consisted of a variety of questions requiring employment of algorithms covered in the course to solve a problem. In addition, 3 programming projects with Matlab implementation were assigned to exemplify the concepts. Most of the programming assignments were real-world problems in biomedical science and mainly pertaining to the instructor’s previous or ongoing research projects in the area of

Computational Physiology. The most comprehensive programming assignment was assigned during the last few weeks of the class and required employment of various methods that students had learned throughout the semester. The goal of the project was the design of an efficient (in terms of computational complexity) and accurate (in terms of the algorithm's sensitivity and specificity) method which automatically distinguishes valid physiological pulsatile signals from the ones contaminated with noise and artifacts. As noise is a constantly existing problem in any application that involves information extraction from clinical recordings of physiological signals, designing an accurate and efficient method to address this issue is of significant practical value to the clinicians[11]. For the project, the students were asked to apply different dimensionality reduction methods (e.g. Principal Component Analysis, Linear Discriminant Analysis) to a dataset of more than 14,000 arterial blood pressure pulses and compare the results of valid pulse recognition using different classification techniques (e.g. parametric Bayesian, probabilistic neural network, nearest neighbor).

Evaluating the Efficacy of Teaching

To assess the efficacy of teaching, we use two different measures: students' grades; and students' teaching evaluation results. Figure 1 shows the boxplots of students' grades (normalized to 100) for various exams in the chronological order taken throughout the semester. As the magenta dashed line on the plot shows, average student grades improved over the semester. This observation can be explained as follows: One of the main challenges of teaching a multidisciplinary class with a heterogeneous student body is that the prior experiences and knowledge of the students with respect to the subject of the course are not aligned. For our pattern recognition class, although all the students had (at least) basic familiarity with probability and statistics, and linear algebra, some of them had taken the pre-requisite courses more than one year before the class when they were undergraduate students. These students had more difficulty at the beginning of the semester and needed additional assistance to catch up with the rest of the class. To address this issue, the instructor asked the students to actively attend office hours when more time could be spent on reviewing the pre-requisites and/or any other topic which the students had difficulty to understand. In addition, the instructor made extensive efforts during the first few weeks of the class to adapt and/or adjust course content, teaching techniques and learning activities to align interests, knowledge and foci of students. We believe that these efforts have contributed to the gradual improvement of the performance of the class throughout the semester.

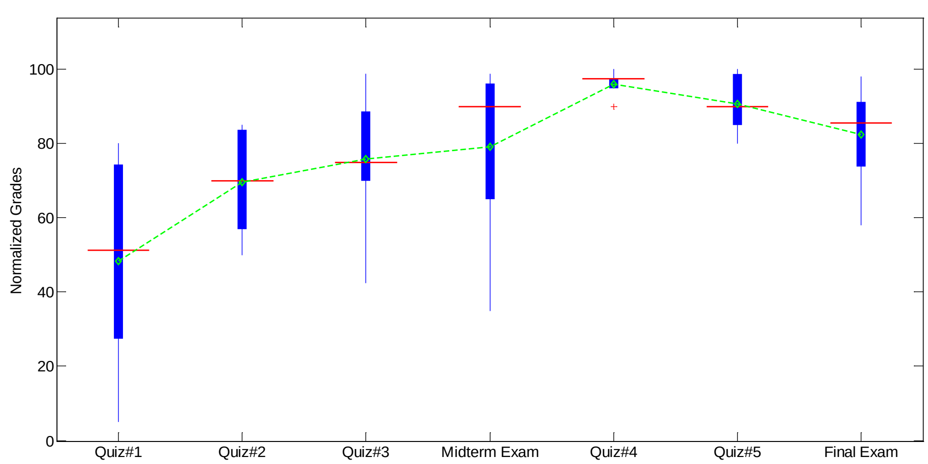

Figure 1. Box plot of students' normalized grades for various exams. The magenta dashed line connects the average grade of different exams

Figure 2 presents the boxplot of the student's grades for three computer programming assignments. Similarly we observe that throughout the semester programming performance of the class improved. Although all the students had basic familiarity with programming, some of them had not implemented any project in Matlab before. Hence, the first computer assignment was challenging for them. Given this observation, the instructor tried to address this issue by providing additional assistance outside of the class and also through email communication with the students. She also defined a simple bonus project before the second assignment to provide an additional opportunity for the students to improve their Matlab programming skills before the second computer assignment. Furthermore, she allowed the students to work in groups of two for the remaining two assignments. These modifications could have contributed to the improvement of the class performance in computer programming.

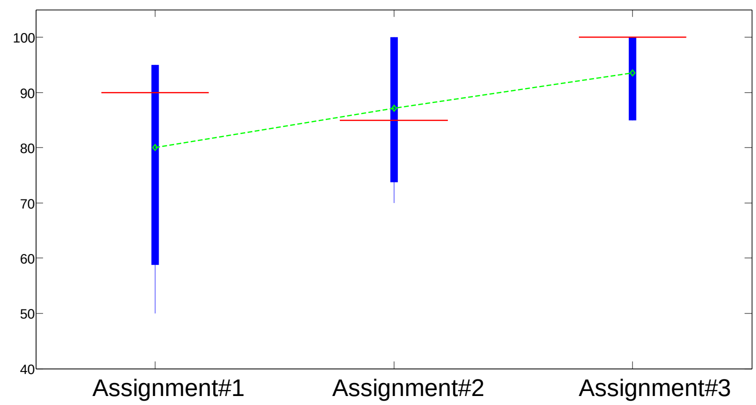

Figure 2. Box plot of students' normalized grades for programming assignments. The magenta dashed line connects the average grade of different assignments

Figure 3 shows the distribution of the final course grades. To calculate the final course grade, the lowest quiz score was not considered, and each remaining quiz accounted for 6% of the final grade. The midterm and final exam had 20% and 26% weight, respectively. Programming assignments accounted for 20% and class attendance for 10% of the total grade. As seen from figure 3, 43% of students received a score of over 90 (an "A"), while 29% of the students received a score in the range of 80-90 (a "B"). Therefore, a total of 72% of the students in the class displayed good or excellent performance in learning the course materials.

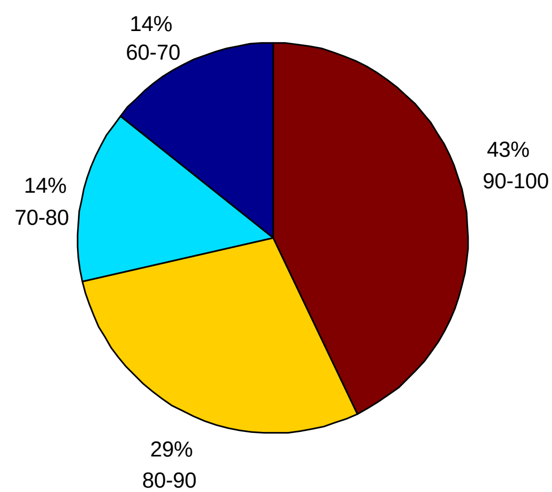

Figure 3. Distribution of course final grades normalized to 100

As a second measure for assessment of teaching efficacy, we exploit the results of CSULB's Student Perception of Teaching (SPOT) survey conducted on the last day of instruction. This anonymous survey consists of 9 multiple-choice and 3 open-ended questions as shown in Table 1. The possible answers for the multiple-choice questions are: 1: strongly disagree; 2: moderately disagree; 3: slightly disagree; 4: slightly agree; 5: moderately agree; and 6: strongly agree.

Table 1. Multiple-choice and open-ended questions of SPOT survey.

Multiple-Choice Questions
Class Time was used efficiently.
Concepts were presented in a manner that helped me learn.
Assignments contributed to my learning.
The instructor responded respectfully to student questions and viewpoints.
The instructor was effective at teaching the subject matter in this course.
The instructor communicates well.
Graded assignments were returned promptly.
The grading criteria for this course were clearly defined.
The instructor was available during the office hours.
Open-Ended Questions
What contributed most to your learning in this course?
Did anything interfere with your learning in this course?
What suggestions or recommendations do you think would help your instructor prepare to teach this course again?

Figure 4 shows the results of students' responses to the multiple-choice questions. The comments that students made in response to open-ended questions are summarized in the Appendix.

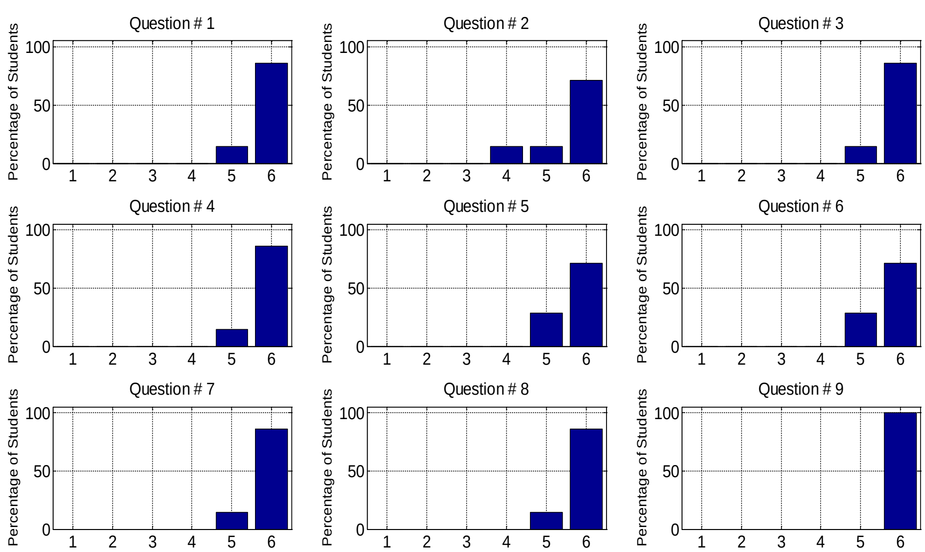

Figure 4. Student responses to the multiple-choice questions of the CSULB SPOT survey

From Figure 4 we observe that at least 70% of the students strongly agreed with all the statements of the SPOT survey. More specifically, about 85% of the students strongly agreed that the class time was used efficiently (question #1). As mentioned earlier, our “pattern recognition” class was offered as a standard computer science course named “CECS 590-Special Topics in Computer Science”. This course has only 2.5 hours contact hours per week without any lab session. Hence, the instructor of “pattern recognition” had to carefully plan the course flow and manage the limited class time to cover both the theory and the computer programming aspect of each technique. For example, there was one topic (decision trees) that the instructor had originally planned to cover, but due to time constraints at the end of the semester, she decided to skip that topic and not compromise the quality of teaching with the quantity of the materials covered. Overall, as evidenced by student responses to question #1, the instructor’s efforts to efficiently manage the limited class time have been effective.

For question #2, about 70% of the students strongly agreed that the concepts were presented in a manner that helped them understand, while 15% of them moderately agreed and the remaining 15% slightly agreed with this statement. This variability in the students’ responses could be explained by the fact that achieving a single teaching approach which optimally woks for every student in a heterogonous class is almost inconceivable. In our class, the students came from three different majors: computer science, mathematics and biology. While students majoring in mathematics may be more interested in learning the theoretical aspect of a technique, computer science or biology students would require a teaching approach mainly emphasizing the heuristic aspects of the discussed techniques. Therefore, keeping the perfect balance between theory and practice in such a multidisciplinary class can be really challenging.

About 85% of the students strongly agreed that the given course assignments have helped them in the enhancement of their learning experience (question #3). We believe that this success in delivering the learning objective grew out of the use of real-world examples and projects from biomedical science. Students’ comments (See Appendix) do substantiate this theory, as well. We also believe that assigning a comprehensive project at the end of the semester which required the application of different techniques learned throughout the semester could have concluded our efforts to enhance students learning.

To complete the evaluation of teaching efficacy for our “Pattern Recognition” class, we studied the students’ response to the remaining multiple-choice questions of SPOT: About 85% of the students strongly agreed with the statements of questions # 4, 7 and 8, while all the students strongly agreed with the statement of question #9. About 70% of the students strongly agreed and 30% of students moderately agreed that the course teaching has been effective (question #5).

Conclusion

As it is expected for teaching any multidisciplinary class, instructing “pattern recognition” to a group of students with various backgrounds was challenging. Although the group of participating students was carefully selected, fairly early in the course, we realized that the knowledge, experience and foci of students were not aligned. While students majoring in mathematics had a solid theoretical foundation for learning the concepts, they needed additional assistance (specially at the beginning of the semester) with respect to implementation of computer programming assignments. There were also few computer science and biology students who had no prior Matlab coding experience. Hence, these students experienced some difficulties with programming assignments, as well. As CECS 590 had no associated lab, we tried to address this issue by providing additional assistance outside of the class and also through email communications with the students. Furthermore, we defined a simple bonus project that helped the students to increase their programming skills.

For a few students the lack of sufficient knowledge in probability and statistics was another hindering factor. Leveraging on the additional contact time provided during office hours, the instructor assisted those students to catch up with the rest of the class by reviewing the related pre-requisite materials and answering their questions in details. The use of real-world examples or projects related to biomedical science substantially enhanced students' learning, as evidenced by their comments. Hence, the result of our teaching experience is in agreement with the accepted belief that project-based learning can be an efficient teaching approach for multidisciplinary classes. We also learned that in order to further enhance the learning experience in a multidisciplinary class, the instructor needs to have a careful but flexible plan in terms of course content, teaching techniques and learning activities which allows for continuous adjustments of strategies depending on the class dynamics or educational needs of the students. Although the inclusion of some level of flexibility in teaching of any course is necessary, the need for adaptive teaching is more significant in a multidisciplinary class, because alignment of interests, knowledge and foci of a heterogeneous student body requires more considerable effort.

In summary, our pilot offering of "Pattern Recognition" has been a challenging but rewarding experience. As our next step, we plan to leverage on our current experience in teaching such multidisciplinary class and develop and offer a course on "Applied Data Analysis" which is tailored for a heterogeneous Engineering student body.

Bibliography

1. Joachim S, Petra BS. Thinking in design teams-an analysis of team communication. Design Studies 2002;23:473-96.
2. Williams R. The Challenge of Multidisciplinary Education for Undergradutes. MIT Faculty NewsLetter 2007;19.
3. Hendrickson CT. Encouraging Multi-Disciplinary Education and Inter-disciplinary Research. Third National Congress on Civil Engineering Education; 2001; Houston,Texas: American Society of Civil Engineers. p. 1-5.
4. Accrediation Board for Engineering and Technology (ABET). *Retrieved on March 3, 2014* at http://www.abet.org/.
5. Wicklein RC, Schell JW. Case Studies of Multidisciplinary Approaches to Integrating Mathematics, Science and Technology Education Technology Education 1995;6.
6. Dekhtyar A, Goodman AL, Montana A. Teaching Bioinformatics in Concert: an Interdisciplinary Collaborative Project-based Experience. 2013 American Society for Engineering Education Pacific Southwest Conference; 2013; Riverside: American Soeciety of Engineering Education. p. 147-68.
7. Tang HH, Hsiao E. The advantages and disadvantages of multidisciplinary collaboration in design education. 5th International Congress of International Association of Societies of Design Research (IASDR). Tokyo, Japan.2013:1-10.
8. Bishop CM. Pattern Recognition and Machine Learning (Information Science and Statistics): Springer-Verlag New York, Inc.; 2006.
9. Mohri M, Rostamizadeh A, Talwalkar A. Foundations of Machine Learning: The MIT Press; 2012.
10. Maldonado S, L'Huillier G. SVM-Based Feature Selection and Classification for Email Filtering. In: Latorre Carmona P, Sánchez JS, Fred ALN, eds. Pattern Recognition - Applications and Methods: Springer Berlin Heidelberg; 2013:135-48.
11. Asgari S, Bergsneider M, Hu X. A robust approach toward recognizing valid arterial-blood-pressure pulses. IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society 2010;14:166-72.

Appendix

Here we include the students' responses to open-ended questions of the course SOPT survey. Please note that not all students answered these questions.

Question: What contributed most to your learning in this course?
"Good execution of real-world examples and real-world projects."
"All the applicable methods that are useful for classification."

"Lectures that were presented clearly."
"The examples in class and the programming assignments."
"Lecture and quizzes were the most helpful. Programming exercises were insightful, but also very tedious at times and didn't help that much."
"The assignments that we had. Instructor tried to make sure that all students understand the materials."
"Examples and problems in class."

Question: Did anything interfere with your learning in this course?
"None."
"None."
"None."
"Lacking mathematical foundation necessary for the course."
"The programming assignments. It was hard to know if my code was correct or wrong, or what might have gone wrong."

Question: What suggestions or recommendations do you think would help your instructor prepare to teach this course again?
"Maybe more homework and projects."
"I sometimes felt that it was hard to both keep up with the concepts that were being explained while taking notes at the same time. Perhaps notes can be written a little more sparingly and concepts can be elaborated on a little bit more from time to time before moving to the next part of a lecture."
"I think the homeworks were great, I really liked the class and I learned a lot."
"Maybe have some "example" data we could test our code on w/ associated "example" answers to see if it was working correctly."