BCSCXXX: Robot Learning-II : Machine Learning

Lecturer:
Office Hours: Thursdays from 1 to 3 PM. Office: Room xxx. Phone number: xxxx-xxx-xxx.

Course Summary

Course Title	**Robot Learning-II : Machine Learning**		
Course Code	**BCSCXXX**		
Credit Units	4		
Work load	Lectures 2h/week	Tutorial 2 h/week	Lab 2 h/week
Pre-requisite(s)	None		
Study Programme	BIT		
Level	3st Year		
Semester	1		
Mode of Teaching	Lectures + Tutorial + Practical		
Course Description	This course will focus on Machine Learning (ML) to tackle real world examples with mathematical tools and implementation of such algorithms in the robotics lab. The course will also discuss recent applications of machine learning, such as to robotic control, data mining, autonomous navigation, bioinformatics, speech recognition, and text and web data processing. Topics include: supervised learning (generative/discriminative learning, parametric/non-parametric learning, neural networks, support vector machines); unsupervised learning (clustering, dimensionality reduction, kernel methods); learning theory (bias/variance tradeoffs; VC theory; large margins); reinforcement learning and adaptive control.		

Course Objectives
The main goal of the course is to learn the basic tools to implement Machine Learning algorithms and use them in Robotics applications.

Learning Outcomes

Upon completion of this course, the student will be able to:

- Understand the basic concepts and principles of Machine Learning (ML).
- Understand the basic concepts and principles of ML & Robot programming.
- Able to recognize the real world Machine Learning problems.
- Design, write, and test ML algorithms to implement solutions to given problems
- Able to implement an ML algorithm to a robot in the lab
- Develop program competencies
- To be able to explain the principles, practices and scope of ML & robot implementation

Prerequisite(s)
A computer programming course preferably Java and/or Python

Co-requisite(s) and Concurrent Prerequisite(s)
None

Equivalent(s)
To be determined

Delivery Methods

- Lectures and Group Discussions

- Hands on training, design and implementation in Robotics Lab
- Technical Workshops, Conferences and seminars
- Videos
- Presentations of student developed applications
- Team works

Faculty of Science and Technology's Expectations

Students are required to participate fully and engage in lively, respectful debates and hands-on Java programming practicals. Much of the in-class work will involve critical thinking and cooperative learning that stress individual responsibility and collaborative approaches to Java programming. All lab work involve hands-on coding. Students are required to take part and make a meaningful contribution to their learning and to take part in technical workshops, conferences and seminars. Your teacher will be a facilitator to you as you go through the course content. Whether you go through all the course content or not with your teacher, it is your responsibility to ensure you go through all the content either alone or as a study group. This is important because test and exam questions will come from any of the content detailed in this course outline.

In this class, we believe that sometimes you can lead the group and we expect your best. We expect you to understand and implement ML algorithms and implement what you learned, read the books, take notes and go back over your notes and code after each class. Arrive early to class, pay attention, ask questions, and work with others. We expect you to turn your cell phones off before class. Don't come in late or leave early unless absolutely necessary. We teach not only through lecture and discussion but with example. We use the readings, workshops, conferences, seminars and guest speakers as additional help. We love to teach and we want you to leave the class feeling special, ready to go and more aware of how to use your skills to make the world a better place.

Class Policies

- Late submission of assignments will not be accepted. All the assignments given must be completed on time. Penalties for any form of cheating or plagiarism are severe. Written work submitted must be a student's own. All sources of information used must be identified
- Students are responsible for both the information given in class and the readings from the text book or supplemental resources. If absent when materials are distributed, it is the responsibility of that student to get the information from another student
- All mobile phones must be turned off or put on silence mode during lectures
- Students must always be on time for classes, no habit of late coming will be entertained
- Students are required to stay for the entire duration of the lecture
- Attendance will always be monitored and students are required to have a class attendance record of at least (75%) attendance. Students with class attendance of less than 75% will be barred from writing examinations. Attendance is defined as coming to class on time and leaving after the class is over. Coming on time and leaving before the class is over or coming to class just before the class is over is NOT considered attendance
- Although occasional absence may be unavoidable, it in no way excuses you from meeting the requirements of the course
- If you have a prior commitment, it must be approved in advance with your professor and the work for that week will have to be made up
- If you have an emergency, the material can be made up within the next class period. It is your responsibility to notify the professor and the faculty administrator, and it is your responsibility to set up a strategy for completion
- The student is responsible for the material discussed and the assignments given on the day of his/her absence

<u>Must do Assignments</u>

1. Write JAVA/ Python Programs to implement ML algorithms.
2. Pick a robot from the lab or design one for an ML implementation
3. Write ML related algorithms to sharpen programming skills
4. Find working example programs and implement it
5. Able to read and understand an existing code related to the course content

NOTE
A Guest lecture on ML will be conducted plus a practical and project guidance.

Course Details

Week 1: Basic Concepts

Understanding the concepts of the subject, expectations and the projects

Assignment
Search about the problems suitable for the course, find case studies and present in the class

Week 2 : Supervised Learning

Assignment
Design a concept from the material that can work for a robot to develop an algorithm and implement in Java.

Week 3 : Logistic regression perceptron

Assignment
Assignments and lab work

Week 4 : Generative learning algorithms

Assignment
Assignments and lab work

Week 5 : Break

Week 6: Support vector machines

Assignment
Assignments and lab work

Week 7: Course work I + Test 1

Week 8: Learning Algorithms

Assignment
Implementation: Algorithm, psudo code and Java code with a robotics example.

Week 9 : Deep Learning

Neural Networks, Vectorization

Assignment
Implementation: Algorithm, psudo code and Java code with a robot example.

Week 10: Unsupervised Learning

Clustering. K-means, Mixture of Gaussians. Factor analysis.

Assignment

Each student to write a simple program that demonstrates how to apply learning.

Week 11: PCA (Principal components analysis).

Assignment
Assignments and lab work

Week 12: Coursework II & Presentations

Week 13: ICA (Independent components analysis).

Assignment
Assignments and lab work

Week 14: Reinforcement learning and control

Assignment
Assignments and lab work

Week 15: Case Studies

Discussion on ML examples

Week 16: Project presentations

Assignment
Each team will present ML algorithms implemented from around the globe and implement a small demo related to the subject.

Week 17: GUEST LECTURE

Assessment and Criteria

- **Course & Lab Work** **55%**
  - Class Presentation 05%
  - Attendance 05%
  - Assignment 15%
  - Implementations 30%
- **Exams** **45%**
  - Midterm 20
  - Final 25
- **Total** **100%**

Reference Readings:

1. UNDERSTANDING MACHINE LEARNING, From Theory to Algorithms, Shai Shalev-Shwartz, Shai Ben-David

2. Building Machine Learning Systems with Python, Richard, Coelho.

3. Russell and Norvig. Artificial Intelligence: A Modern Approach.

References for programming:

1. Herbert Schildt .,Java The Complete Reference. McGraw Hill.
2. Riccardo Flask .,Java for Beginners. 2nd Edition.
3. E. BalaGuruSamy., Programming with Java 3e Primer. The McGraw-Hill Companies.
4. Simon Kendal., Object Oriented Programming using Java. 1st Edition.
5. Laura Lemay, Charles L.Perkins., teach Yourself Java in 21 days.