Course guides
270659 - OTDM - Optimization Techniques for Data Mining

Last modified: 12/07/2021

Unit in charge: Barcelona School of Informatics
Teaching unit: 715 - EIO - Department of Statistics and Operations Research.

Degree: MASTER'S DEGREE IN INNOVATION AND RESEARCH IN INFORMATICS (Syllabus 2012). (Optional subject).

Academic year: 2021 **ECTS Credits:** 6.0 **Languages:** English

LECTURER

Coordinating lecturer: JORDI CASTRO PÉREZ

Others: Primer quadrimestre:
JORDI CASTRO PÉREZ - 10
CRISTINA CORCHERO GARCIA - 10
FRANCISCO JAVIER HEREDIA CERVERA - 10

PRIOR SKILLS

Basic background in linear algebra, calculus, and programming languages is needed for the course.

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
CEC1. Ability to apply scientific methodologies in the study and analysis of phenomena and systems in any field of Information Technology as well as in the conception, design and implementation of innovative and original computing solutions.
CEC2. Capacity for mathematical modelling, calculation and experimental design in engineering technology centres and business, particularly in research and innovation in all areas of Computer Science.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE3.3. Capability to understand the computational requirements of problems from non-informatics disciplines and to make significant contributions in multidisciplinary teams that use computing.
CEE5.3. Capability to work in interdisciplinary engineering services teams and, provided the necessary domain experience, capability to work autonomously in specific service systems.

Generical:
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CG3. Capacity for mathematical modeling, calculation and experimental designing in technology and companies engineering centers, particularly in research and innovation in all areas of Computer Science.

Transversal:
CTR2. SUSTAINABILITY AND SOCIAL COMMITMENT : Capability to know and understand the complexity of the typical economic and social phenomena of the welfare society. Capacity for being able to analyze and assess the social and environmental impact.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.

Basic:
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

TEACHING METHODOLOGY

The students will have available all the course material.

About two thirds of lecture time will be devoted to optimization algorithms and their properties, and the rest will be for presenting and solving exercises and problems

Lab sessions (using AMPL) will be devoted to the solution of some data mining applications.

LEARNING OBJECTIVES OF THE SUBJECT

1. Discerning what is an optimization problem and its type and having a basic knowledge of optimization algorithms
2. Formulating optimization problems and representing them through a modeling language
3.Choosing an adequate solver type for a given problem
4.Using publicly available and commercial solvers. Interpreting results from solvers and communicating in writing results from optimization

STUDY LOAD

Type	Hours	Percentage
Self study	96,0	64.00
Hours large group	54,0	36.00

Total learning time: 150 h

CONTENTS

Unconstrained Optimization

Description:
Optimality conditions. Convexity. Descent directions.
Line search. Acceptability of step sizes.
General minimization algorithm.
Gradient method. Rate of convergence.
Newton's method. Factorizations to ensure convergence.
Weighted least squares.
Introduction to AMPL. The Neos solver site.

Constrained Optimization and Support Vector Machines.

Description:
- Introduction to Support Vector Macines (SVM)
- KKT Optimality conditions of constrained optimization. Optimality conditions of SVM.
- Duality in Optimization. The dual of the SVM.

Integer Programming

Description:
- Modelling problems with binary variables.
- The branch and bound algorithm for integer programming
- Gomory's cutting planes algorithm.
- Minimal spanning tree problem and algorithms of Kruskal and Prim.

ACTIVITIES

Unconstrained Optimization

Description:
Optimality conditions. Convexity. Descent directions.
Line search. Acceptability of step sizes.
General minimization algorithm.
Gradient method. Rate of convergence.
Newton's method. Factorizations to ensure convergence.
Weighted least squares.
Introduction to AMPL. The Neos solver site.

Specific objectives:
1, 2, 3, 4

Related competencies :
CG3. Capacity for mathematical modeling, calculation and experimental designing in technology and companies engineering centers, particularly in research and innovation in all areas of Computer Science.
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CEE5.3. Capability to work in interdisciplinary engineering services teams and, provided the necessary domain experience, capability to work autonomously in specific service systems.
CEE3.3. Capability to understand the computational requirements of problems from non-informatics disciplines and to make significant contributions in multidisciplinary teams that use computing.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEC1. Ability to apply scientific methodologies in the study and analysis of phenomena and systems in any field of Information Technology as well as in the conception, design and implementation of innovative and original computing solutions.
CEC2. Capacity for mathematical modelling, calculation and experimental design in engineering technology centres and business, particularly in research and innovation in all areas of Computer Science.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR2. SUSTAINABILITY AND SOCIAL COMMITMENT : Capability to know and understand the complexity of the typical economic and social phenomena of the welfare society. Capacity for being able to analyze and assess the social and environmental impact.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 52h 18m
Theory classes: 17h 18m
Self study: 35h

Constrained Optimization and Support Vector Machines

Description:
- Introduction to Support Vector Macines (SVM)
- KKT Optimality conditions of constrained optimization. Optimality conditions of SVM.
- Duality in Optimization. The dual of the SVM.

Specific objectives:
1, 2, 3, 4

Related competencies :
CG3. Capacity for mathematical modeling, calculation and experimental designing in technology and companies engineering centers, particularly in research and innovation in all areas of Computer Science.
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CEE5.3. Capability to work in interdisciplinary engineering services teams and, provided the necessary domain experience, capability to work autonomously in specific service systems.
CEE3.3. Capability to understand the computational requirements of problems from non-informatics disciplines and to make significant contributions in multidisciplinary teams that use computing.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEC1. Ability to apply scientific methodologies in the study and analysis of phenomena and systems in any field of Information Technology as well as in the conception, design and implementation of innovative and original computing solutions.
CEC2. Capacity for mathematical modelling, calculation and experimental design in engineering technology centres and business, particularly in research and innovation in all areas of Computer Science.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR2. SUSTAINABILITY AND SOCIAL COMMITMENT : Capability to know and understand the complexity of the typical economic and social phenomena of the welfare society. Capacity for being able to analyze and assess the social and environmental impact.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 52h 24m
Theory classes: 17h 24m
Self study: 35h

Integer Programming

Description:
- Modelling problems with binary variables.
- The branch and bound algorithm for integer programming
- Gomory's cutting planes algorithm.
- Minimal spanning tree problem and algorithms of Kruskal and Prim

Specific objectives:
1, 2, 3, 4

Related competencies :
CG3. Capacity for mathematical modeling, calculation and experimental designing in technology and companies engineering centers, particularly in research and innovation in all areas of Computer Science.
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CEE5.3. Capability to work in interdisciplinary engineering services teams and, provided the necessary domain experience, capability to work autonomously in specific service systems.
CEE3.3. Capability to understand the computational requirements of problems from non-informatics disciplines and to make significant contributions in multidisciplinary teams that use computing.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEC1. Ability to apply scientific methodologies in the study and analysis of phenomena and systems in any field of Information Technology as well as in the conception, design and implementation of innovative and original computing solutions.
CEC2. Capacity for mathematical modelling, calculation and experimental design in engineering technology centres and business, particularly in research and innovation in all areas of Computer Science.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR2. SUSTAINABILITY AND SOCIAL COMMITMENT : Capability to know and understand the complexity of the typical economic and social phenomena of the welfare society. Capacity for being able to analyze and assess the social and environmental impact.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 52h 18m
Theory classes: 17h 18m
Self study: 35h

GRADING SYSTEM

- Theory (40%). There will be a short midterm exam based on practical questions for the first two parts of the course, and some individual numerical exercises for the third (and last) part.

- Practical assigments (60%). There will be 3 lab assignments, one for each part of the course, all of them with the same weight on the final mark.

Additional exercises provided during the lectures may be taken into consideration to decide or to boost the final mark.

BIBLIOGRAPHY

Basic:
- Nocedal, J.; Wright, S.J. Numerical optimization. 2nd ed. Springer Science+Business Media, 2006. ISBN 0387303030.
- Luenberger, D.G.; Ye, Y. Linear and nonlinear programming. 4th ed. Springer, 2016. ISBN 9783319188416.
- Wolsey, L.A. Integer programming. John Wiley & Sons, 1998. ISBN 0471283665.
- Fourer, R.; Gay, D.M.; Kernighan, B.W. AMPL: a modeling language for mathematical programming. 2nd ed. Thomson Brooks/Cole, 2003. ISBN 0534388094.
- Cristianini, N.; Shawe-Taylor, J. An introduction to support vector machines: and other kermel-based learning methods. Cambridge University Press, 2000. ISBN 0521780195.

RESOURCES

Hyperlink:
- http://www-eio.upc.es/teaching/ple/pfc_ing.html
- https://neos-server.org/neos/