OR 750 – Computational Optimization – Syllabus (Fall 2018)

Instructor: Dr. Ran Ji
Office: Nguyen Engineering Building, Room 2210
Email: rji2@gmu.edu
Phone: (703) 993-5993

Office Hours: Thursdays 3:00 – 5:00 PM or by appointment

Class Place And Time: Innovation Hall 139
Wednesday 4:30 – 7:10 PM

Prerequisites: This course is suitable for students with some basic knowledge in mathematical programming, optimization, and some basic programming experience. No specific courses are required. Basic knowledge on linear programming is preferred.

Course Description And Objectives: This course focuses the description, formulation, and implementation of efficient computational algorithms/methods to solve large-scale optimization problems, with various applications in machine learning, financial portfolio optimization, supply chain optimization, facility location and capacity allocation, power system operations with renewable energy, etc. These problems can take various forms, such as deterministic vs. stochastic; continuous vs. integer; convex vs. non-convex, etc.

The course aims to provide students guidance to effectively formulate the mathematical optimization problems, and to derive reformulations/algorithms to efficiently solve the data-driven large-scale optimization problems. The course will acquaint the students with the programming languages, academic and commercial optimization solvers.

The course is broadly composed of two parts:

(1) The first part focuses mainly on ***convex optimization***. The basic concepts and theory of convex programming (SDP, SOCP) will be covered. Implementation of the computational methods (reformulations, relaxation, linearization) will be conducted using Matlab and Disciplined Convex Programming package CVX.
(2) The second part is devoted to tractable modeling of important classes of optimization problems, including ***mixed-integer programming (MIP), stochastic and chance constrained programming, and distributionally robust optimization problems***. Computationally efficient methods (reformulation, relaxation, decomposition, approximation) will be covered.

Upon completion of the class, students should be able to:

(1) detect and exploit the properties of mathematical optimization problems
(2) derive computationally tractable reformulations and propose/design computationally efficient algorithmic methods to solve large-scale optimization problems;
(3) develop programming and coding skills to implement the reformulation and algorithmic methods, and to interact with off-the-shelf optimization solvers with useful settings/options;

Note: The list of topics is tentative and is subject to update.

Topics And Schedule:

Lecture 1: Introduction
- Course Setup
- Overview of Optimization Problems and Computational Solvers
- Introduction to AMPL

Lecture 2: Basics of Convexity – Sets and Functions
- Basics Convexity Concepts
- Convex Sets, Convex Functions, Saddle Points

Lecture 3: Convex Optimization I
- Revisit Linear Programming
- Least Squares, Quadratic Programming, Quadratically Constrained Programming
- Second Order Cone Programming (SOCP)
- Introduction to Matlab-CVX

Lecture 4: Convex Optimization II
- Geometric Programming (GP)
- Semidefinite Programming (SDP)
- Implementation with Matlab-CVX

Lecture 5: Convex Duality and Optimal Conditions
- Dual Cones, Conic Duality Theorems
- Lagrange Duality, Optimality Conditions (KKT Conditions)

Lecture 6: Applications: Convex Optimization Perspectives on Machine Learning Models
- Regularization: Robust Linear Regression, Lasso
- Classification: Support Vector Machine

Lecture 7: Mixed-Integer Linear Optimization Models
- Computationally Efficient Mixed-Integer Linear Optimization Problems
- Implementation using AMPL (or Python-Pyomo) with different solvers

Lecture 8: Mixed-Integer Non-linear Optimization Models
- Computationally Efficient Mixed-Integer Non-linear Optimization Problems
- McCormick Relaxation, Reformulation Linearization Technique (RLT)

Lecture 9: Chance-Constrained Programming Models
- Classical Stochastic Programming Method via Distributional Information
- Data-Driven Combinatorial Optimization Method (Threshold Boolean Method)
- Application: Value-at-Risk measure, Portfolio Optimization Problem

Lecture 10: Distributionally Robust Optimization (DRO) – Part I
- Overview of Optimization under Uncertainty: Robust Optimization, Stochastic Optimization, Distributionally Robust Optimization
- Distributionally Robust Optimization with Moment-Based Ambiguity

- Convex Duality Reformulation for DRO

Lecture 11: Distributionally Robust Optimization (DRO) – Part II

- Distributionally Robust Optimization with Statistical Distance-Based Ambiguity
- Phi-divergence, Total Variation, Wasserstein Distance
- Computationally Tractable Reformulations/Approximations for DRO

Lecture 12: Distributionally Robust Optimization (DRO) – Part III

- DRO Perspectives on Machine Learning Models: DRO-Linear/Logistic Regression, Lasso, SVM, Wassertein Generative Adversarial Networks (Wasserstein GAN), etc.
- DRO Applications: Portfolio Optimization, Power System with Renewable Energy, etc.

Lecture 13: Large-Scale Optimization Algorithms (if time permitted)

- Smooth: Gradient Descent Method, Accelerated Gradient Descent Method.
- Nonsmooth: Subgradient Method
- Stochastic Approximation: Stochastic Gradient Descent (SGD)

Lecture 14: Student Project Presentation

Software: AMPL with different solvers and Matlab-CVX will be the primary software/programming language adopted in this course.

AMPL algebraic/mathematical modeling language (see blackboard)
Academic/Commercial optimization solvers: Cplex, Gurobi, Knitro, Minos, Snopt, Bonmin, Ipopt, etc. (see blackboard)
Matlab (available on VSE computers, or via GMU Virtual Computing Lab)
CVX: Matlab package for Disciplined Convex Programming (see blackboard)
Python with Pyomo package may also be covered (see blackboard).

Textbooks: *The course materials will be adapted from several textbooks and some research papers. The books listed below are suggested but NOT required.*

- Boyd, Stephen, and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004. [Available Online]
- Bertsekas, Dimitri P. *Convex Optimization Algorithms*. Belmont: Athena Scientific, 2015. [Link]
- Ben-Tal, Aharon, and Arkadi Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, Engineering Applications*, 2015. [Available Online]
- Fourer, Robert, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*, 2002. Duxbury Press. [Avaiable Online]
- Grant, Michael, Stephen Boyd, and Yinyu Ye. *CVX: Matlab Software for Disciplined Convex Programming*, 2008. [Link]
- *Some relevant research papers will be distributed on blackboard*

Grading Components and Policy:

The course grade will be based on individual homework assignments, an oral presentation of a selected research paper, a term project (including both presentation and technical report). Each grading component is described below.

Assignments:	45%
Paper Presentation:	10%
Term Project:	40%
Class Participation:	5%

The +/- letter grades will be used.
The instructor reserves the rights to make minor modifications in the percentages related to the number and difficulty of the homework given.

Homework Assignments:
The homework will be assigned on a bi-weekly basis, resulting in around a total of six assignments. Each assignment generally contains two types of questions, the "handwriting" questions to enhance the conceptual understanding and mathematical programming skills, the "coding/programming" questions to practice the computational algorithms and methods to solve real-life problems with synthetic or real dataset. The homework will be always posted on blackboard and will be due in two weeks. The "handwriting" questions will be turned in at the beginning of class, while the "coding" questions will be compressed in a '.zip' file and submitted via blackboard. Students are encouraged to get familiar with the Latex software to type your answers.

The discussions among students are welcome, however the homework must be submitted individually. Any detection of duplicate submissions (both handwriting and coding questions) will invoke an honor code violation.

Research Paper Presentation:
Each student will pick a recent research paper of interest on a specific topic within the theme of this class. The goal of the research paper presentation is to present the key ideas, models, solution methods and results of the paper at an appropriate technical level (conference-type oral presentation). With good understanding, the students are expected to make their own slides and present the paper in their own words to the peers.

Term Project:
Each student will propose (or pick) a specific topic of interest with appropriate applications to work on a term project. The student is expected to understand the background of optimization problems, design/propose solution method(s), implement them and evaluate them on data sets. The entire project would consist of a project proposal in the mid of the semester, an oral presentation, and a written technical report. The final report is expected to take the format of a conference/journal paper.

Class Website:	**Blackboard**: http://mymson.gmu.edu Click on the Courses tab in the green area (top right-of-center) and then on the OR-750-006 (Fall 2018) link when the course list column appears. The left column menus include the following: **Syllabus**: Class syllabus, schedule, course overview information. **Content**: Links to weekly modules with lecture slides, data files and external readings. **Assignments**: Homework information, data and guidance. Solutions to the Assignments. Links to submit assignments. **Projects**: Project information. Links to submit the proposals and final reports. **Software**: Some resources and materials for used software are available here. **My Grades**: This is the place to check on your grades. **Discussion Board**: At least one discussion board will be open to support communication among students.

General Material

Academic Integrity
Mason is an Honor Code university; please see University Catalog **(**http://oai.gmu.edu/the-mason-honor-code-2/**)** for a full description of the code and the honor committee process. The principle of academic integrity is taken very seriously and violations are treated gravely.

Mason Email Accounts
Students must use their MasonLive email account to receive important University information, including the messages related to this class. See Mason Live (http://masonlive.com) for more information.

Office of Disability Services
If you are a student with a disability and you need academic accommodations, please see me and contact the Office of Disability Services (ODS) at (703) 993-2474. All academic accommodations must be arranged through the ODS (http://ods.gmu.edu).

Writing Center: Robinson Hall A114. Phone: (703) 993-1200. Webpage: http://writingcenter.gmu.edu

University Libraries: “Ask a Librarian”. Webpage: http://library.gmu.edu/mudge/IM/IMRef.html

University Policies: The University Catalog (http://catalog.gmu.edu) is the central resource for university policies affecting student, faculty, and staff conduct in university academic affairs. Other policies are available at University Policy (http://universitypolicy.gmu.edu). All members of the university community are responsible for knowing and following established policies.