>>> Course & People

Refresher course: Numerical Matrix Analysis & Optimization

Lecturers: from the applied maths and computer science lab LJK

Jérôme Malick CNRS researcher

jerome.malick@
univ-grenoble-alpes.fr

Franck Iutzeler Assistant Professor

franck.iutzeler@
univ-grenoble-alpes.fr

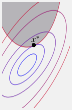
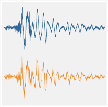
Students: $\sim$40 Master2 students

- Master Computer Sciences **MoSiG**
- Master of Applied Maths **MSIAM**
- Other Masters (MiSCIT, ORCO)

- From around the world: Europe, Russia, India, China, Africa, ...
- diverse origins and backgrounds: need for basic commun knowledge

Matrices and optimization are at the **heart of computational mathematics**

With applications everywhere, e.g.

Machine Learning	Energy Management	Signal/Image Processing
		

Mix between **theory** (meaning of a problem; existence/uniqueness of solution; math properties) and **practice** (Computability, speed, and use of standard libraries to solve numerically these problems.)

This short course focuses on **matrix analysis and optimization in action**

with exercises inspired from:

- Google PageRank, Image processing
- Machine learning applications (regression, classification)

This course is not a standard course on linear algebra or optimization

- not a math course

  basic knowledge is assumed (take a look to your undergraduate courses)
- not an algorithmic course

  basic programming skills are expected (check-out Python tutorials)

This course is

- a review of basics of matrix analysis – from numerical perspective
- a short overview of numerical optimization
- includes quick recalls from matrix calculus and differential calculus

Schedule: a quite dense week (before the other courses start)

Tutorials goals:

- to manipulate some of the notions on simple examples
  (no fancy maths and no computations – they will be done on machines)
- prepare the labs sessions...

Practical sessions goals:

- matrices and optimization in action for learning and ranking
- In Python using Jupyter notebooks (work by groups of 1 or 2)
- Warm-up: go through the Python Basics and NumPy introduction

  `https://github.com/iutzeler/refresher-course`

Course 1 on matrix analysis

1. Basics on matrices
    - Matrices and operations between matrices
    - Operations on matrices : tranpose, trace, determinant
    - Special matrices (triangular, symmetric, orthogonal, invertible, SDP)
    - Decomposition : (P)LU, QR

2. Linear systems
    - Invertible systems, linear least-squares, linear least-norm
    - Easy systems for special matrices (triangular, orthogonal,...)
    - Solving systems : by factorization, by iterative methods, by optimization
    - Practical considerations (preconditioning, software,...)

3. Spectral decompostions
    - Eigenvalues : real, complex, spectral radius
    - Eigenvalue decomposition, geometric interpretation
    - Singular value decomposition : SVD, compact SVD, link with eigenvalues

\+ Note on matrix norms : standard norms, induced/operator norms, connection with spectral radius

Course 2 on numerical optimization

1. Introduction : what is optimization ?
   - Optimization problems : definitions, exemples, first properties
   - How to solve an optimization problems : exact/approximate solutions, difficult/impossible in general, "easy" for linear... and convex problems
   - A classification: cvx/non-cvx, smooth/non-smooth, stochastic/deterministic

2. Convexity and optimization
   - Convex sets and functions, exemples
   - Convex optimization problems : global solutions, convex set of solutions
   - Recognizing convexity: definition, convexity-preserving operations, Hessian
   - In practice : modeling, interface, algorithms, experience

3. Simple algorithm for a simple problem : gradient algorithm
   - Unconstrained convex differentiable problems, optimality conditions
   - Gradient algorithm with 4 ingredients of all algorithms
   - Study : convergence theorem vs numerical experiments
   - Beyond gradient : acceleration, 2nd order, Newton

\+ Notes:
   - Recalls on derivatives : gradient, Hessian, chain rule, examples
   - Application to classification : geometrical/statistical problems, optim. models

This course introduces **material** for several courses, among them:

- "Efficient methods in optimization"
  (on convex analysis & complexity and convergence of algorithms)
- "Convex and distributed optimization" (for large-scale applications & big data)
- 3 courses on machine learning !
- PDEs, inverse problems, stats...

Useful Links:

- Python/Numpy's documentation
  `http://docs.scipy.org/doc/numpy-1.11.0/reference/`
- Stephen's Boyd website (check the courses, quizzes, and exercises)
  `http://web.stanford.edu/~boyd/`

Main References:

- Horn, R. & Johnson, C.: *Matrix analysis*.
- Boyd, S. & Vandenberghe, L.: *Convex optimization*.
- Ciarlet, Ph.: *Introduction à l'analyse numérique et l'optimisation*.
- Hiriart-Urruty J.-B., Lemaréchal C.: *Fundamentals of convex analysis*.