Algorithmic Machine Learning

Introduction to the course

Prof. Pietro Michiardi

Objectives / learning outcomes of the course

- **Gain hands-on experience in real-life Data Science projects**

- **Use knowledge acquired in other courses: math and CS**

- **Develop a methodology to address challenges such as:**
  - Data preparation
  - Data exploration
  - Algorithm / model selection
  - Experimental evaluation and validation

Notebooks, not lectures!

- **Essentially, there will be no traditional lectures**
  - Knowledge from introduction to machine learning is a requisite
  - Knowledge from distributed computing is strongly suggested
  - Taking the Advanced Statistical Inference course is a BIG plus

- **Laboratories to learn and practice**
  - Guided Notebooks
  - Challenge Notebooks
  - Industrial Notebooks

Guided Notebooks

- **A self-contained studying and development environment**
  - Contains text, reference material, code, questions, …
- **A precious guide (through questions) to attack a data science problem**
  - Data exploration
  - Data preparation
  - Algorithm / model selection
  - Experimental evaluation and validation
- **Weight = 1 for the computation of the final grade**

Challenge Notebooks

- **Everything starts with a well defined problem statement**
  - It is up to you to use and adapt the methodology from guided notebooks
- **Students are supposed to**
  - Define a viable approach
  - Use techniques and models learned in MALIS and ASI
  - Eventually use distributed programming libraries
- **Winning the challenge**
  - Groups will be ranked based on a well defined performance metric, e.g. MSE
  - The top 3 groups will receive bonus points: 3 for 1st rank, 2 for 2nd rank, 1 for 3rd rank
- **Weight = 2 for the computation of the final grade**

Industrial Notebooks

- **These labs are MANDATORY**
  - Students will be guided through these notebooks, through a series of questions as done by operational data scientists

- **Topics covered**
  - Not seen in any of the classes (currently)
  - Require studying on your own

- **Weight = 1 for the computation of the final grade**

How to be a successful student

- **Do not underestimate this course!**
  - Be independent and dare to explore, and expand your Guided Notebooks
  - Study or revise the theory
  - Follow links on the Guided Notebooks
  - Lookup for references from this introductory slide deck

- **Discuss with TAs!**
  - Prepare your question, come up already with a plausible answer
  - Ask for advice, ask for references, for links, …
  - Ask if the quality of your work meets grading requirements (see next)

How to be a successful student

- **Is this a course about algorithm design?**
  - Standard libraries of machine learning algorithms implemented in an efficient way
  - Algorithmic concepts discussed in the Notebooks
  - Optional, advanced approaches are more than welcome!

- **Does this course make me a Data Scientist?**
  - No, it's the whole track, not a single "hacking" course
  - Aim at "learning the hard way" and put into practice theoretical concepts

- **Do I need to know how to program?**
  - Yes, and this is mandatory
  - We will use Python

Grading

- **Five main items, a bonus for challenges**
    1. Code quality
    2. Code efficiency
    3. Quality of data analysis and depth
    4. Quality of answers to questions
    5. Correctness
    - Rank (for challenges)
- **In practice**
    - Each item (except rank) brings up to 4 points
    - Sum of all points gives grade
- **Final grade: weighted sum of notebooks grades**

Concepts from the Notebooks

Material that you are supposed to study or revise on your own

Recommender algorithms

- **Textbook material**
  - *"Mining of Massive Datasets"*, by Jure Leskovec, Anand Rajaraman, Jeff Ullman, Stanford University
    http://www.mmds.org/

  ➔ Focus on chapter 9
  - *"Implicit Feedback for Inferring User Preference: A Bibliography"*,
    by Diane Kelly and Jaime Teevan

- **Research articles**
  - *"Matrix Completion and Low-Rank SVD viaFast Alternating Least Squares"*,
    by Trevor Hastie, Rahul Mazumder, Jason D. Lee, Reza Zadeh

- **Advanced readings**
  - *"Probabilistic Models for Data Combination in Recommender Systems"*,
    by Sinead Williamson and Zoubin Ghahramani
  - *"Generalized Low Rank Models"*,
    by Madeleine Udell, Corinne Horn, Reza Zadeh, Stephen Boyd

Monte Carlo simulation

- **Basic material**
  - “Monte Carlo Simulation Tutorial”,
    https://www.solver.com/monte-carlo-simulation-example
  - *“The Monte Carlo Method”,* WikiPedia
  https://en.wikipedia.org/wiki/Monte_Carlo_method
  - *“An Introduction to Statistical Learning”*,
    by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani
  ➔Chapter 2 and Chapter 3, *“Linear Models”*
  - *“Kernel Density Estimation”*, WikiPedia
  https://en.wikipedia.org/wiki/Kernel_density_estimation

- **Advanced readings**
  - *“An Introduction to Statistical Learning”*,
    by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani
  ➔Chapter 7, *“Moving beyond linearity”*
  - *“Backtesting Value-at-Risk Models”*, by Kansantaloustiede et al

Challenge: a practical regression problem

- **Textbook material**
  - *"An Introduction to Statistical Learning"*,
    by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani

  ➔Chapter 2, Chapter 3, *"Linear Models"*

  ➔Chapter 8, "Tree-based Methods"

- **Advanced readings**
  - *"An Introduction to Statistical Learning"*,
    by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani

  ➔Chapter 9, "Support Vector Machines"

  - *"Gradient Boosting"*, WikiPedia
    https://en.wikipedia.org/wiki/Gradient_boosting
  - *"XGBoost: A Scalable Tree Boosting System"*,
    https://arxiv.org/abs/1603.02754
  - A video tutorial on XGBoost,

  https://campus.datacamp.com/courses/extreme-gradient-boosting-with-xgboost/

SAFRAN laboratory: time series data

- **Textbook material**
  - *“Introduction to Time Series and Forecasting”*,
    by Peter J. Brockwell Richard A. Davis
  - *“Time series analysis”*,
    by Jan Grandell
  - *“An Introductory Study on Time Series Modeling and Forecasting”*,
    by Ratnadip Adhikari, R. K. Agrawal https://arxiv.org/abs/1302.6613

- **Advanced readings**
  - *“Bayesian Time Series Learning with Gaussian Processes”*,
    by Roger Frigola

  http://www.rogerfrigola.com/doc/thesis.pdf

SAP laboratory: reinforcement learning

- **Textbook material**
  - *"Reinforcement Learning: An Introduction"*,
    by Richard S. Sutton and Andrew G. Barto

- **Websites / Blogs**
  - https://gym.openai.com/

- **Advanced Readings**
  - *"Playing Atari with Deep Reinforcement Learning"*,
    by Volodymyr Mnih, et. al.
    https://arxiv.org/abs/1312.5602
  - *"Deep Reinforcement Learning: An Overview"*,
    by Yuxi Li
    https://arxiv.org/abs/1701.07274

Calendar and timings

Rules for the laboratories

- **Each laboratory has dedicated Q/A slots**
  - Each notebook is granted 2 slots
  - TAs will answer questions related to the specific notebook in its slots

- **Deadlines**
  - Know your deadlines! Each notebooks has a specific one

- **Presence**
  - MANDATORY for industrial notebooks
  - Warmly suggested for all other notebooks, otherwise you won’t have the possibility to ask questions

Schedule of the laboratories

- **Recommender algorithms**
  - March 16, 23
  - Deadline: March 29th at 23h59m59s

- **Monte Carlo simulation**
  - March 30, April 6
  - Deadline: April 12th at 23h59m59s

- **Challenge**
  - April 13, 20, 27, May 4th
  - Deadline: May 17th at 23h59m59s

➔ **Discussion lecture about challenge: May 18th**

Schedule of the laboratories

- **SAFRAN**
  - May 25, June 1
  - Deadline: at the end of each laboratory

- **SAP**
  - June 8, 15
  - Deadline: June 15th at 23h59m59s