BST 262

Purpose of This Course

Big data is everywhere, from omics and health policy to environmental health. Every single aspect of the health sciences is being transformed. However, it is hard to navigate and critically assess tools and techniques in such a fast-moving big data panorama. In this course, we will give a critical presentation of theoretical approaches and software implementations of tools to collect, store and process data at scale. The goal is not just to learn recipes to manipulate big data but learn how to reason in terms of big data, from software design and tool selection to implementation, optimization and maintenance.

Course Description

The course fulfills a core course requirement for the Master of Science in Health Data Science. It is an intermediate course in data science and an introductory course in big data. The course is also open to students interested in processing large amounts of data.

The course will combine lectures and presentations by domain experts. Examples will be used throughout the lectures, and students are encouraged to use their laptop in class.

We will see how big data changes several aspects of data science (for instance, data management, software development and visualization) and how we can leverage dedicated tools to work with big data efficiently.

Learning outcomes

At the completion of the course, students will be able to:

1. assess the strengths and limitations of a variety of approaches to manage and compute with big data,
2. design, implement and optimize software to deal with big data,
3. understand and use best practices in software development.

The course will also give students opportunities to expand their data science portfolio.

Pre-requisites

We expect students to be familiar with Unix command lines tools (as in http://swcarpentry.github.io/shell-novice/) and have experience with a scripting language such as R or Python (at the level of BST 260 and BST 261).

Credits

2.5

Instructor Information

Instructor Name:	Christine Choirat
Instructor's Title:	Research Scientist
Instructor's Department association:	Biostatistics
Instructor's email:	cchoirat@iq.harvard.edu
Phone:	617-496-5097

Office Hours: By appointment, office TBD in the Biostatistics department

Office Address: IQSS, 1737 Cambridge Street
CGIS Knafel Building
Office K323
Cambridge, MA 02138

Teaching Assistants: Qian Di (qiandi@mail.harvard.edu), Ben Sabath (mbsabath@hsph.harvard.edu)
Office hours: TBD

Course Structure

This course assumes substantial and informed student participation. General discussion of theory and practice is encouraged. At a minimum, being informed requires class attendance, completion of assigned readings and homework, and attention to data science news. Class attendance and participation are strongly encouraged, but not mandatory.

Grading, Progress and Performance

Grades will be based on two mandatory problem sets. Each problem set will correspond to 50% (= 50 points) of the final grade. The first problem set will be available by the end of week 3 and the second problem set by the end of week 6. Students will be required to submit problem set solutions within two weeks. Grades, and feedback when appropriate, will be returned two weeks after submission.

Students will submit a markdown document that combines commented code for data analysis and detailed and structured explanations of the algorithms and software tools that were used.

Each assessment will have 4 to 6 questions focused on specific aspects of the course.

Assessment 1 will address the material covered in weeks 1-3, for example, how to:

- turn an R script into an R package,
- run benchmarks and improve code performance,
- estimate time series at scale.

Assessment 2 will be more challenging. It will assume that the material of weeks 1-3 is mastered and will use, among others, examples from:

- large-scale spatial databases (visualization and linkages),
- JVM big data platforms.

We will provide an evaluation grid with each assignment. Grades will take into account the following aspects.

- Is the code clearly written and suitably commented?
- Is the code correct? Is it efficient? Is it scalable?
- Is there a critical discussion of the chosen approach, its strengths and potential limitations?

Late or make-up assignments are not accepted except in case of **major** events. It will be possible to obtain extra-credit (at most 5 points) by contributing to an open-source big data project (either by direct code contributions or by opening meaningful issues on GitHub, contributing to an issue thread or providing a useful answer to a Slashdot question).

Texts and Reading Materials

Required text: none

Class notes: will be available on GitHub and/or Canvas each week and will cover the topics discussed in class. Class notes will be required readings.

Recommended books: Specific chapters of these books will be recommended readings for specific modules.

1. Chacon S. and Straub B. (2014). Pro Git. Apress. https://git-scm.com/book/en/v2
2. Wickham H. (2014). Advanced R. Chapman & Hall/CRC. http://adv-r.had.co.nz/
3. Wickham H. (2015). R packages. O'Reilly. http://r-pkgs.had.co.nz/
4. Wickham H. and Grolemund G. (2016). R for Data Science. O'Reilly. http://r4ds.had.co.nz/
5. Lim A. and Tjhi W. (2015). R High Performance Programming, Packt Publishing. https://www.packtpub.com/application-development/r-high-performance-programming
6. Eddelbuettel D. (2013). Seamless R and C++ Integration with Rcpp. Springer. http://www.springer.com/us/book/9781461468677

Session by Session Detail

Week 1 - Basic tools

Lecture 1. Unix scripting, make
Readings: https://swcarpentry.github.io/shell-novice/

Lecture 2. Version control: Git and GitHub (guest lecture: Ista Zhan)
Readings: Chacon S. and Straub B. (2014), Chapters 1, 2, 6.

Week 2 - Creating and maintaining R packages

Lecture 3. Rationale, package structure, available tools
Readings: Wickham H. (2015)

Lecture 4. Basics of software engineering: unit testing, continuous integration, code coverage
Readings: Class notes

Week 3 - Software optimization

Lecture 5. Measuring performance: profiling and benchmarking tools
Readings: Wickham H. (2014), Chapters "Performance", "Profiling", "Memory"

Lecture 6. Improving performance: an introduction to C/C++, Rcpp
Readings: Wickham H. (2014), Chapters "Rcpp", "R's C interface". Eddelbuettel D. (2013), Chapters 1, 2, 3, 4, 8.

Week 4 – Databases

Lecture 7. Overview of SQL (SQLite, PostgreSQL) and noSQL databases (HBase, MongoDB, Cassandra, BigTable, …)
Readings: Class notes

Lecture 8. R database interfaces (in particular through dplyr)
Readings: Class notes

Week 5 - Analyzing data that does not fit in memory

Lecture 9. Pure R solutions (sampling, ff and ffbase, other interpreters)
JVM solutions (h20, Spark)
Readings: Class notes

Lecture 10. An introduction to parallel computing; clusters and cloud computing.
"Divide and Conquer" (MapReduce approaches)
Readings: Class notes

Week 6 – Visualization

Lecture 11. Principles of visualization (guest lecture: James Honaker)
Readings: Class notes

Lecture 12. Maps and GIS: principles of GIS, using R as a GIS, PostGIS
Readings: Class notes

Weeks 7 & 8 - Guest lectures (order and precise schedule TBD)

Lecture 13. Software project management (Danny Brooke)

Lecture 14. R and Spark (Ellen Kraffmiller and Robert Treacy)

Lecture 15. Advanced GIS and remote sensing (TBD)

Lecture 16. Cluster architecture (William J. Horka)

Harvard Chan Policies and Expectations

Inclusivity Statement

Diversity and inclusiveness are fundamental to public health education and practice. It is a requirement that you have an open mind and respect differences of all kinds. I share responsibility with you for creating a learning climate that is hospitable to all perspectives and cultures; please contact me if you have any concerns or suggestions.

Academic Integrity

Harvard University provides students with clear guidelines regarding academic standards and behavior. All work submitted to meet course requirements is expected to be a student's own work. In the preparation of work submitted to meet course requirements, students should always take great care to distinguish their own ideas and knowledge from information derived from sources. Please refer to policy in the student handbook for details on attributing credit and for doing independent work when required by the instructor.

Accommodations for Students with Disabilities

Harvard University provides academic accommodations to students with disabilities. Any requests for academic accommodations should ideally be made before the first week of the semester, except for unusual circumstances, so arrangements can be made. Students must register with the Local Disability Coordinator in the Office for Student Affairs to verify their eligibility for appropriate accommodations. Contact the OSA **studentaffairs@hsph.harvard.edu** in all cases, including temporary disabilities.

Course Evaluations

Constructive feedback from students is a valuable resource for improving teaching. The feedback should be specific, focused and respectful. It should also address aspects of the course and teaching that are positive as well as those which need improvement.

Completion of the evaluation is a requirement for each course. Your grade will not be available until you submit the evaluation. In addition, registration for future terms will be blocked until you have completed evaluations for courses in prior terms.