Computational modeling in the data science era

Name : Mahak

Phd Scholar ,Department of Computer Science,KUK

Abstract

Our teaching experience, we formed a set of design principles for an integrative course. We independently implemented these principles in two public research universities, in Canada and the US, for a course targeting graduate students and upper-division undergraduates. We discuss and contrast these implementations, and suggest ways in which the teaching of computational science can continue to be revised going forward.

Keywords: Course content; Data analytics; Simulations

• Introduction

Computational modeling has been taught for several decades: for example, the well-known PhD in Computational Sciences and Informatics at George Mason University was created in 1992 [3]. The ways in which computational modeling needs to be taught changes with societal demands and innovations. In the early 2000s, data mining was increasingly combined with computational modeling, for example in the Computational and Statistical Learning PhD introduced at Carnegie Mellon University in 2001 [1]. Since the late 2000s, a shift has operated toward 'data science'. The growing societal needs for trained data scientists have been emphasized by several large bodies

In this paper, we detail the design of a modelling and simulation course targeting upper-division undergraduates and graduate students. Our focus is to update our course content to include more data science topics. The course was independently designed by the two authors, who were strategic hires in data science at two different North American universities, thus allowing to compare design and experiences. In Section 2 we review how courses and programs at a variety of institutions have addressed computational modeling and/or data science. In Section 3 we introduce the two designs chosen for our course. As for many data science programs [1, 2, 3], we found that having a project was essential. A discussion on how such projects were set-up and managed, as well as selected examples, are given in Section 4. Finally, we discuss the next update of course material going forward in Section 5 and concluding remarks. In this paper, we detail the design of a modelling and simulation course targeting upper-division undergraduates and graduate students. Our focus is to update our course content to include more data science topics. The course was independently designed by the two authors, who were strategic hires in data science at two different t North American universities, thus allowing to compare design and experiences. In Section 2 we review how courses and programs at a variety of institutions have addressed computational modeling and/or data science. In Section 3 we introduce the two designs chosen for our course. As for many data science programs [1, 2, 3], we found that having a project was essential. A discussion on how such projects were set-up and managed, as well as selected examples, are given in Section 4. Finally, we discuss the next update of course material going forward in Section 5 and concluding remarks.

Content of modeling & data science courses/programs

➢ Modelling and simulation

Generic modeling and simulation courses typically include the modeling process, and techniques for continuous (system dynamics, systems of deferential equations) or discrete models (cellular automata, agent-based models). Additional topics range from hybrid models to experimental design or large-scale simulations. The lists of topics for three courses are provided in Table 1. The first two courses are two instances of a graduate-level course that we collegially taught at Simon Fraser University, Canada [12], and that influenced the design of the two versions presented in section 3. For comparison, the last course is an undergraduate course

Table 1: Content for three courses on modelling and simulation

<table>
<tr><td>MATH800 (Fall'08)</td><td>MATH800 (Fall'11)</td><td>COSC/MATH 201 (Fall'11)</td></tr>
<tr><td colspan="3">Modeling process</td></tr>
<tr><td>Modelling social networks</td><td>Topological data analysis</td><td>High-performance computing</td></tr>
<tr><td>Spatial analysis in criminology</td><td rowspan="3">Fuzzy cognitive maps</td><td>Computational error</td></tr>
<tr><td>Crime modelling
Issues of data manipulation</td><td>Simulation techniques</td></tr>
<tr><td>Health promotion in prisons</td><td>Empirical models</td></tr>
<tr><td colspan="3">Cellular automata</td></tr>
<tr><td>Computational criminology</td><td rowspan="2">Complex networks</td><td>Monte Carlo simulations</td></tr>
<tr><td>Stat. model for homelessness
Urban dynamics</td><td>Interactive visualization</td></tr>
<tr><td>Complex systems and obesity</td><td colspan="2">Agent-based modeling</td></tr>
<tr><td>Operation research</td><td colspan="2">System dynamics</td></tr>
<tr><td>Queuing theory</td><td>Queuing theory</td><td></td></tr>
</table>

• Integrated computational modeling and data science

Teaching such model building skills is thus essential. To illustrate that point, imagine that a data scientist is meeting with a decision-maker to discuss findings regarding the data. The decision-maker may not only be interested in the properties of the data at present: there is likely to be questions about how data came to be the way it is, or what interventions can change the situation going forward. Answering such questions requires models to generate theories, or conduct what-if analyses. That is, the toolbox of a data scientist should not be limited to regression analysis and data mining but should also include discrete and continuous models. Critically, modeling should not be seen as an advanced/optional course for data scientists or a separate track within a larger program. Rather, the two need to be integrated, and that is commonly not the case [3]. In the words of Finkelstein, "what is surprising is that science largely looks at data and models separately, and as a result we miss the principal challenge

– the articulation of modelling and experimentation" [14]. There have been plans to develop curriculum integrating computational modeling and data science. For example, Embry-Riddle Aeronautical University planned to add a data science track to its computational program [17]. The program in Computational and Data Sciences at George Mason University is built on that integration. Of the 6 required computational and data sciences core courses, 2 are modeling and simulations courses, and 1 is an introductory course covering scientific discovery via simulations and data analysis [3]. The balance of content in this program has inspired the design of our course

Our course designs

- Design principles

The context for the course development is as follows. The two authors have a background in computational modeling and simulation. In Fall 2015, they were part of strategic hires in data science in the computer science departments of two public research universities: Lakehead University ('LU') in Ontario, Canada, and Northern Illinois University ('NIU') in Illinois, USA. The two departments deliver master degrees but not PhD. In both cases, no classes were currently taught in modeling and simulation. However, classes touching on data science were introduced *at the same time* as we delivered the course. At LU, the class on big data was taught in parallel, while at NIU classes on data visualization and information retrieval were taught in the same semester. We thus had to develop a course on computational modeling and data science that would be directly accessible upon admission to the program, while minimizing partial overlaps with other courses (whose content was in a state of flux). While we did not expect a qualified teaching assistant to be available for such new courses, this did not affect our design principles.

Design principles for a course on computational modeling and data science

(1) Students should first be provided an overview of the field (in line with Table 1). Then, the course should include at least 3 modeling techniques, exposing students to both individual-level models and aggregate models.

(2) Students should be familiarized with one programming environment for modeling and one programming environment for data analytics. These environments should use the object-oriented and imperative paradigms that students are most familiar with. Since no prior exposure to either data science or simulation is assumed, the introduction to these environments will have to start from basic syntax.

(3) Students should be actively engaged in interdisciplinary projects, as is generally the case in computational and/or data science programs [3, 17, 1] and as we previously recommended [9, 10].

(4) Students should be exposed to seminal and recent research papers on modeling and data science. Students should critically evaluate these papers to find shortcomings, and form teams to provide solutions combining data science and modeling.

- **Implementation**

Structural-equation modeling was an optional module, which was also taught. The design principles were implemented as follows:

(1) Three modeling techniques are used: 1 individual-level technique (Cellular Automata) and 2 aggregate techniques (Compartmental Models, and Fuzzy Cognitive Maps). These three techniques were selected or extensive research experience by the instructor [18, 13].

(2) Students were introduced to R and Matlab. Since fuzzy logic was an important modeling tool in the course, Matlab was chosen as it provides a fuzzy toolbox. R was selected as it is commonly used for data analytics (alongside Python).

(3) Students worked in 3 teams of 2 and 1 'team' of 1 on projects. They were pointed to data repositories in week 1 and asked at the same time to provide a survey of the literature (via google scholar) on a problem of their choice. Their survey was discussed in class, following by discussions about the model and finally using the data.

(4) All students read 13 research papers, which were debated in class. In addition, teams performed of varying amount of readings for the specific needs of their project. A LaTeX report summarized each project.

Table 2: Course content for the LU implementation

Module	Theme	Name
1		Principles of modeling and simulation
2	Programming	Programming with MATLAB
3		Programming with R
4	Data science	Data preprocessing
5	Modeling techniques	Cellular automata
6		Compartmental models
7		Fuzzy Logic
8		Fuzzy Cognitive Maps
9		Adaptive Neuro Fuzzy Inference System

Table 3: Course content for the NIU implementation

Module	Theme	Name	Content
1		Overview of modeling and data science	
2	Basic models	Basics of abstraction	Abstracting a problem Implementation in AnyLogic
3		Modeling infectious diseases	Principles of Compartmental Models The SIR model What makes a good model Mathematical notation
4		Network concepts	What is a network Finding important elements Modeling processes on networks Network properties
5		Modeling with net-works	Implementing a network model with AnyLogic
6		Agent-based models	What is an ABM Implementing ABMs with AnyLogic
6	Data science	Understanding the data for/from models	Data Mining in Python
7		Using data to populate and validate models	Initializing from ind. distributions Initializing from individual cases Validating at the aggregate level
8		Cleaning data	Data cleaning and wrangling in Python
9		Acquiring data	Building Selenium scripts Web crawlers Collection on humans and ethics

Course projects

Project design

The data science and/or computational modeling programs that we surveyed often require that students take courses in another discipline and do a project/capstone. For example, at the College of Charleston, data science students learned about molecular biology or psychology; doing a data science capstone was a core course [1]. Similarly, at George Mason University, students had concentration areas in physics, chemistry, and biology; their research project was optional [3]. Since we developed a course rather than a program, our research projects required a minimal exposure to another field. As both authors have ample experience in health informatics, projects were mostly geared toward the health sciences.

At LU, students were asked to present a research proposal consisting of: an objective, a discussion of recent research and identification of a knowledge gap, links to dataset (if required), and a potential conference/journal (to identify the intended readership). At NIU, projects were designed for the students, who instead started by a literature review. For all projects, using real data was mandatory. Indeed, "engaging students by using real data to address scientific questions in formal education settings is known to be an elective instructional approach" [2] and data science topics easily lend themselves to it.

Projects

There were four LU projects. One individual projected was a software benchmarking, while the others were performed in teams with the following subjects, techniques, and datasets:

- A study of cancer using structural equation modeling and fuzzy cognitive maps, based on the P53 Mutant Dataset from the University of California at Irvine (Figure 1(a)).
- An examination of nutritional status in the US based on fuzzy inference systems, using the National Health and Nutrition Examination Survey (NHANES) data.
- A predictive model of cardiovascular diseases using structural equation modeling and fuzzy cognitive maps, using the Canadian Community Health Survey (CCHS) (Figure 1(b)).

The six NIU projects varied from reviews to pilot projects on new topics or follow-up of previously developed models. It was decided that, to emphasize that students were dealing with real-world data, they should have access as much as possible to those directly involved with the data. Local non-governmental organizations were thus approached for the 3 months preceding the start of classes, with the order to share their data and expertise in return for help with making sense of the data. One organization accepted to partner, provided their anonym zed individual-level data with explanations, and agreed to regularly meet with students who would take on the project. Similarly, researchers in other fields were approached and one accepted to discuss with students the results on their data analysis on gerontology data.

5 Discussion and conclusion

Data and models should be looked at together [14]. Revising a course on computational mod-eling and simulation during the data science era provides a much needed incentive to work toward that integration. In this paper, we introduced our design principles for a course on computational modeling and data science, and discussed two independent implementations of these principles at two public research universities. Since both implementations had to com-bine data science and modeling topics, it was expected that neither would be classified as only

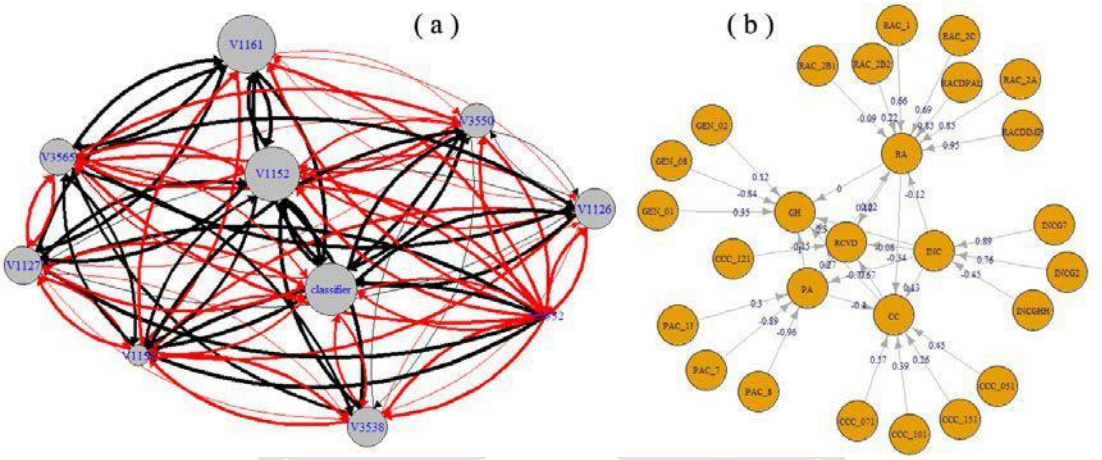

Figure 1: Two models developed in the LU projects. A fuzzy cognitive map (FCM) was built to model the transcriptional activity (i.e., active/inactive) of the tumor protein p53 based on the DNA mutations in the Mutant Dataset from the University of California at Irvine (a). A structural equation model (SEM) was built to relate the risk of cardiovascular disease (noted RCVD) to variables from the Canadian Community Health Survey, such as self-perceived health or having high blood pressure (b).

In our projects, students generally have to find datasets either as the primary object of study or to complement data that is already provided to them. They are pointed to sources such as HealthData.gov, physionet.org, or the UK Data Service. While identifying or navigating repositories is relatively straightforward, finding datasets useful for modeling purposes is much harder. We believe that there is a tendency to capture all quantifiable information, archive it with little data quality assurance, and leave it to the data analyst 'as is'. This can be particularly problematic for students who can be tempted to download and use any dataset matching the right keywords, spend a lot of time on cleaning it, and then (possibly) realize that it has little value. A student reported that it was 'painful' to find the right dataset, as it took 10 to 15 hours. To ensure some standards in the data, we recommended to only use repositories from governments or public agencies, and that datasets were previously used for publications in reputable journals.

Early feedback from the completed class suggests that students appreciated having few lectures on programming or data cleaning, and more time to discuss readings and their projects. At the same time, doing the research for a project that required both modeling and data science skills took much more work than they anticipated when they signed for the course. A potential solution would be to spread the content of the course over two courses, but we would advocate against splitting the course into data analytics on the one side and modeling on the other as it would run contrary to integrating these synergistic topics.

References

[1] P. Anderson; J. Bowring; R. McCauley; G. Pothering (2014) C. Starr. An undergraduate degree in data science: curriculum and a decade of implementation experience. *SIGCSE*, pages 145–150.

[2] K. Borne; S. Jacoby; K. Carney, A. Connolly, dT Eastman; M. Raddick, J. Tyson, ; J. Wallin. The revolution in astronomy education: data science for the masses. http://arxiv.org/abs/0909.3895.

[3] K. Borne; J. Wallin, ; R. Weige l (2009) . The new computational and data sciences undergraduate program at george mason university. In G. Allen, J. Nabrzyski, E. Seidel, G. van Albada, J. Don-garra, and P. Sloot, editors, *Computational Science - ICCS 2009*, volume 5545 of *Lecture Notes in Computer Science*, pages 74–83. Springer Berlin Heidelberg,

[4] A. Chatfield; V. Shlemoon, W. Redublado, ;F. Rahman.(2014) Data scientists as game changers in big data environments. *Proc. of the 25th Australasian Conference on Information System*.

[5] W. Cleveland. Data science: an action plan for expanding the technical areas of the field of statistics. *Statistical Analysis and Data Mining*, 7:414–417, 2014.

[6] Computing Research Association. Cyberinfrastructure for education and learning for the future: a vision and research agenda, 2005. http://archive.cra.org/reports/cyberinfrastructure.pdf.

[7] P. Giabbanelli, T. Torsney-Weird,; D. Finegood.(2011) Building a system dynamics model of indi-vidual energy balance related behaviour. *Canadian Journal of Diabetes*, 35(2):201..

[8] P. J. Giabbanelli. Why having in-person lectures when e-learning and podcasts are available? In *Proc. of the 14th Western Canadian Conference on Computing Education*, WCCCE '09, pages 42–44. ACM, 2009.

[9] P. J. Giabbanelli. Ingredients for student-centered learning in undergraduate computing science courses. In *Proc. of the Seventeenth Western Canadian Conference on Computing Education*, WCCCE '12, pages 7–11. ACM.

[10] P. J. Giabbanelli ; P. J. Jackson.(2015) Using visual analytics to support the integration of expert knowledge in the design of medical models and simulations. *Procedia Computer Science*, 51:755 – 764. International Conference On Computational Science (ICCS).

[11] P. J. Giabbanelli,; A. A. Reid; V. Dabbaghian.(2012) Interdisciplinary teaching and learning in computing science: Three years of experience in the mocssy program. In *Proc. of the Seventeenth Western Canadian Conference on Computing Education*, WCCCE '12, pages 47–51. ACM.

[12] V. K. Mago; R. Mehta; R. Woolrych (2012) E. I. Papageorgiou. Supporting meningitis diagnosis amongst infants and children through the use of fuzzy cognitive mapping. *BMC Medical Infor-matics and Decision Making*, 12(1):1–12.