Introduction to Data Science as a Pathway to Further Study in Computing

Michael Guerzhoy

Center for Statistics and Machine Learning, Princeton University
Dept. of Statistical Sciences, University of Toronto
Li Ka Shing Knowledge Institute, St. Michael's Hospital

Work in Progress!

Introduction

Many institutions have recently introduced *Introduction to Data Science* (I2DS) courses that involve a substantial programming component and do not require CS1 as a pre-requisite. Programming and computational thinking are central to the emerging discipline of data science: there is overlap between traditional CS1 courses and Introduction to DS.

Partly because of the evident societal significance of data science and because data science does not have the problematic reputation of computer science, I2DS courses can attract new and diverse audiences that may not have been interested in taking CS1.

We explore I2DS as a possible alternative path into computing. We would like to address the following questions:

- What are the learning goals in I2DS that involve programming and/or computational thinking?
- How generalizable are the problems students solve in I2DS to what students would encounter in the future?
- Is it feasible for students to pursue a data science sequence rather than CS1-CS2 and be prepared for a career that uses data science?
- To what extent can a pathway through a data science sequence diversify the population of students who graduate from degree programs in computer science and data science?

We are surveying I2DS courses offered at the post-secondary level, with a particular focus on courses that use *R* and the *tidyverse* libraries.

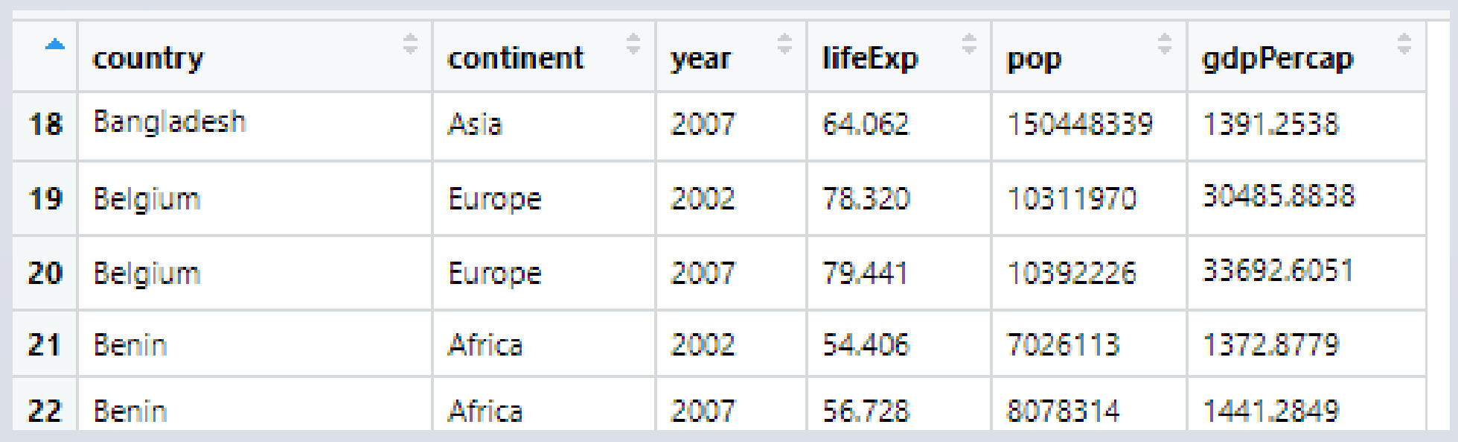
Introduction to Data Science

Introduction to Data Science (I2DS) courses have started appearing in the 2010s. A selection of course webpages with course materials available is at the Course Webpage Wiki [1].

I2DS courses generally cover a combination of traditional introductory statistics (e.g., t-tests and linear regression), basic programming concepts (variables, conditionals, and functions), simulation-based inference, dataframe manipulation ("data wrangling") using packages like the `tidyverse`/`Pandas`, and data visualization. We describe the unique aspects of I2DS courses that are relevant to computational thinking.

Manipulating dataframes

A dataframe is a 2D table. The type of the elements within each column of the dataframe is always the same. Libraries such as `dplyr` (part of `tidyverse`) and Pandas provide a set of operations on dataframes.



	country	continent	year	lifeExp	pop	gdpPercap
18	Bangladesh	Asia	2007	64.062	150448339	1391.2538
19	Belgium	Europe	2002	78.320	10311970	30485.8838
20	Belgium	Europe	2007	79.441	10392226	33692.6051
21	Benin	Africa	2002	54.406	7026113	1372.8779
22	Benin	Africa	2007	56.728	8078314	1441.2849

The following `dplyr` code is a typical example from early *I2DS*:

```
# Compute the number of the distinct countries in Africa that appear in gapminder
gapminder %>% filter(continent == "Africa") # filter out non-Africa rows
          %>% select(country)               # only take the country column
          %>% n_distinct                    # compute num. of distinct elemenents
```

Note that `x %>% h(a) %>% g(b, c) %>% f` is the same as `f(g(h(x, a), b, c))`.

Simulation-Based Inference

Statistical inference can be performed by generating simulated datasets, using re-sampling (as in the Bootrstrap), or using generative probabilistic models. We provide an illustration of using generative probabilistic models for the problem of computing a p-value for the hypothesis that there is no difference between the means of two populations, when we have the two samples `Sample1` and `Sample2` of sizes resp. `N1` and `N2` from the populations. We assume that the population distributions are $N(\theta_1, 1)$ and $N(\theta_2, 1)$ for some unknown θ_1 and θ_2.

```
$\hat{\theta}_1$ <- mean(Sample1)
$\hat{\theta}_2$ <- mean(Sample2)
$\hat{\theta}$ <- mean(Sample 1 + Sample 2)
diff <- $\hat{\theta}_1$ - $\hat{\theta}_2$
for i = 1..K
  FakeSample1[i] <- N1 numbers sampled from $N(\hat{\theta},1)$
  FakeSample2[i] <- N2 numbers sampled from $N(\hat{\theta},1)$
  FakeDiff[i] <- mean(FakeSample1[i]) - mean(FakeSample2[i])
Pval <- #{|FakeDiff| > |diff|}/K
```

I2DS and CS1: Content

There is clearly substantial overlap between the content of CS1 and the content of I2DS: both courses require students to understand, devise, and implement moderately complex algorithms. Can I2DS be viewed as a "CS1 with tables"?

Characterizing the overlap, even for concrete implementations of I2DS and CS1, is challenging. Language-independent assessment is a difficult problem [2], even within CS1. We propose to address the question by analyzing the curricula and learning goals:

- What are the 3-4 most difficult algorithms students are expected to understand? How complex are they?
  - The most complex algorithms in most offerings of I2DS seem comparable to e.g. quicksort
- How varied are the problems that students are expected to be able to solve? To what extent are students expected to simply pattern-match to solve problems? How stereotyped is the code students write?
  - Difficult to formalize

I2DS as a Pathway to Computing

Anecdotally, a nontrivial number of the students who complete I2DS but not CS1 continue to write code.

- At Princeton University, students can take SML310 – Research projects in Data Science, where they learn Python and PyTorch. Students without CS1 have successfully completed the course and gone on to write code in industry
- Large-scale surveys are needed to understand the university and industry careers of students who take I2DS
  - Collaborators wanted!

I2DS and Diversity

- Hypothesis: I2DS should attract a more diverse population than CS1
- Large-scale surveys needed

SML201: Introduction to Data Science

- Aimed mainly at students in the life and social sciences
- Inference-based simulation and dataframe manipulation
  - Similar to most data science courses
- Students expected to be able to implement simulation-based inference
- Programming skills built up manipulating dataframes, applied to inference and cross-validation
  - Inspired by functional CS1s
  - Challenge: coming up with varied and interesting problems based only on dataframe manipulation
- All repeated computation done with map (`sapply` in R) and `dplyr`'s `summarize`
- 50% programming, 50% theory

"Conclusions"

Introduction to Data Science (I2DS) is a course that is in many ways similar to CS1, but I2DS courses have a different emphasis and attract somewhat different audiences than CS1. A substantial number of students start their computing careers in I2DS rather than CS1, so it is important for the CSEd community to understand I2DS.

We are analyzing I2DS courses to understand their relationship to CS1 and plan to develop surveys to understand students' educational and industrial career paths after I2DS.

Open Questions

- Comparing one implementation of I2DS to one implementation of CS1 seems pointless. Comparing all implementations of I2DS to all implementations of CS1 seems impossible. Can this be resolved?
  - Most I2DS curricula are currently substantially similar
  - A huge variety of CS1 courses
- How can we measure to what extent the code students write is stereotyped?

References

[1] *Introduction to Data Science* on the *Course Webpage Wiiki*
http://guerzhoy.princeton.edu/courses_online/Introduction_to_Data_Science

[2] Allison Elliott Tew and Mark Guzdial. "Developing a validated assessment of fundamental CS1 concepts." In *Proc. SIGCSE 2010*