PROJECT REPORT

Py-School-Match

Ashmean Kaur Sran

Abstract

The school choice problem is formulated as a one-sided or a two-sided matching problem. Py-school-match aims to solve the problem of matching students with schools according to a pre-defined set of criteria. It is an open-source Python package that implements multiple matching algorithms in order to assign students to schools. It is specifically designed to allocate students to schools by evaluating different settings and algorithms, without the need to adapt or develop a complete solution. It is a better way to match prospective students with the schools they want to apply for.

Keywords: students; schools; criteria; matching

Introduction

Matching students with schools plays an integral part in shaping future of the young minds all around the world. A typical school choice problem consists of a set of students and schools. Students rank schools in the order of their preferences. Since schools have limited capacities, admissions to popular schools are usually regulated via assignment priorities. In many countries where schools cannot discriminate among student applicants (no entrance exams, no interviews, no previous grades examination, etc.), there is always one question that cannot be answered directly: How to assign students fairly and efficiently? Over the years, there have been many proposed solutions to this problem. These vary from simple "lotteries" (random priority assignment) to more complex graph algorithms. Because each option has its own strengths and weaknesses, and given that in the real world there is an extra layer of requirements (quotas, special conditions, ranks of preference, etc.) it is vital to analyze and simulate every available option. The correct selection of the algorithm

Correspondence: aksran@mun.ca

Faculty of Business Administration, Memorial University of Newfoundland

Full list of author information is available at the end of the article

can have serious effects on efficiency and fairness, as it has been exemplified by studies conducted in school systems from Boston [1] and New York [2].

A matching is pseudo-stable if it does not violate priorities at stabilityconstrained schools. A pseudo-stable matching is stable if there is no student-school pair $(i; s)$ such that student i prefers s to her assignment, she is eligible for s and s has an empty seat. A pseudo-stable matching is student optimal stable if there is no pseudo-stable matching that every student weakly prefers to it and some strictly prefer.

Method

Py-school-match [3] is a Python library that implements multiple matching algorithms and aims to ease the process of choosing the best alternative for each school system. It allows researchers to simply specify the country's requirements or conditions, and then run interchangeably the different algorithms to compare their results. What makes py-school-match different from other libraries is that it is specifically created to be used in the student-to-school assignment problem. Another distinctive characteristic is that it allows the use of quotas, priorities, capacities, among

others, without much effort. py-school-match provides

multiple algorithms ready to use:

- Top Trading Cycles (TTC)
- Deferred acceptance with multiple tie-breaking (DAMTB)
- Deferred acceptance with single tie-breaking (DASTB)
- Stable improvement cycles (SIC)
- Deferred Acceptance with multiple tie-breaking, plus stable cycles (MSIC)
- Deferred Acceptance with single tie-breaking, plus non-stable cycles (NSIC)

Procedure

We have worked on the Stable improvement cycles (SIC) algorithm for assigning seats to a list of students based on their school preference and their nationality acceptance quota in those schools. The procedure has been explained as follows:

- The package will read the data file of name of students with their preferences for schools and their nationality status. It will also read the data file of school names and the seats available in each school.
- It will use the SIC algorithm and assign schools to the students based on their preference of schools and the characteristics(nationality status).
- The users can change the quota percentage as per their school guidelines for assigning students to the schools. According to the percentage of quota set by the user, the ratio of students assigned to school and not assigned to schools will fluctuate.
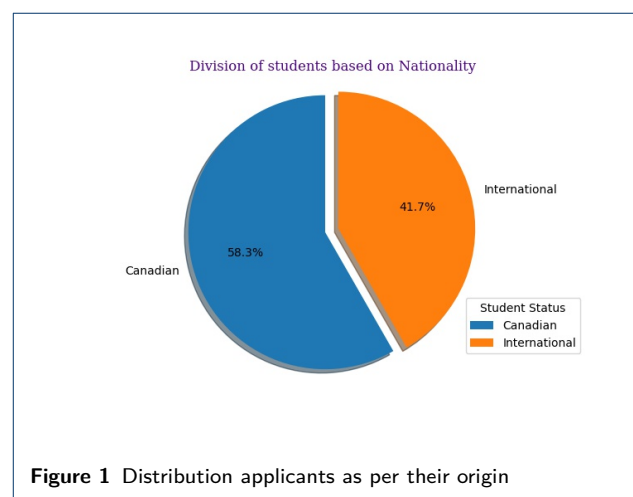
Results

In this section, a list of outputs generated by using the py-school-match package are mentioned.

1. The py-school-match package will generate a .csv file showing list of students who got the schools assigned based on the quota of nationality characteristic along with the list of students who did not get assigned to any school by using the package.

2. The output will also include representation of number of students who were assigned and not assigned to schools with the breakdown of their nationality.

3. Another representation will show the total number of students based on the difference of nationality.

Project Novelty

Py-school-match python library has been updated to include names of students and schools in the results. Running the package will generate a list of students who got assigned or not assigned to schools. The original package was only printing students ids that gets assigned to schools ids. With a few changes in the package, a .csv file gets generated which shows names of the students and schools that are assigned along with the students who did not get assigned to any school.
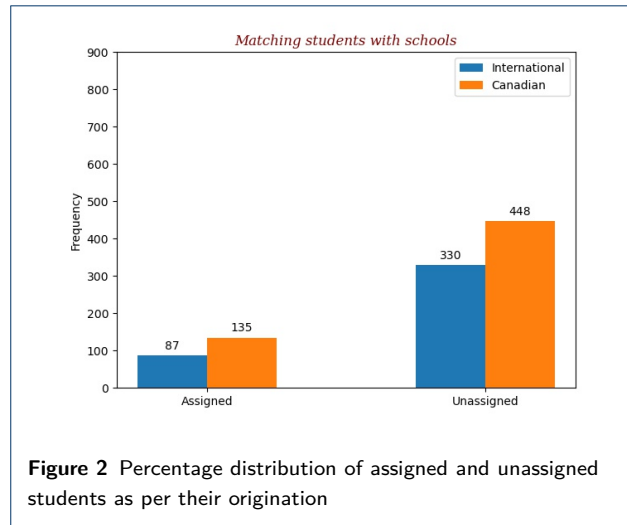
Figures and Tables

Figure 1 Distribution applicants as per their origin

Figure 2 Percentage distribution of assigned and unassigned students as per their origination

Table 1 Output file showing assigned/non-assigned schools to students

Student Name	Assigned School	Nationality
Niels Travers	New Mexico State University	CA
Chet Enderby	William and Catherine Booth College	CA
Grove Caustan	None	INT
Lutero Attock	British Columbia Open University	CA
Shay Naerup	None	INT
Birk Camamile	None	INT

References

1. Abdulkadiroğlu, Atila, Parag A. Pathak, Alvin E. Roth, and Tayfun Sönmez. *"The Boston Public School Match." American Economic Review, 95 (2): 368-371*. DOI: 10.1257/000282805774669637(2005).

2. Abdulkadiroğlu, Atila, Parag A. Pathak, and Alvin E. Roth. *"The New York City High School Match."* American Economic Review, 95 (2): 364-367, DOI: 10.1257/000282805774670167 (2005)

3. Garizio, Iacopo. *Py-school-match: Matching algorithms to assign students to schools*. Journal of Open Source Software, Opt. Express 4(34), 1111, https://doi.org/10.21105/joss.01111 (2019).