HPL Cohort Creation Tool

A step-by-step guide

This guide will provide an overview of the strategies employed in the HPL Cohort Creation Tool (CCT) as well as step-by-step instructions on how to use it. The code for the cohort maker can be found here.

General Idea and Strategy

The goal of the CCT is to automate the creation of cohorts that will produce high-quality experiences during the HPL course. The assumption is that maximizing some element of diversity (years of employment, sector, program, etc) will result in groups that produce high-quality work and foster enriching learning experiences.

To investigate this assumption, the following design was used:

	Size	Makeup
Cohort 1	30	*"Balanced" - the distribution of programs within the cohort match the distribution across the entire group. See below for more details*
Cohort 2	30	*Balanced*
Cohort 3	30	*Balanced*
Cohort 4	30	*Balanced*
Cohort 5	15	*Maximized similarity - students are grouped together based on a calculated similarity metric. See below for more details*
Cohort 6	15	*Mixed similarity - students are mixed in their similarities based on a calculated similarity metric. See below for more details*
Cohort 7	15	*Maximized similarity*
Cohort 8	15	*Mixed similarity*

Description of cohort makeup

Balanced
"Balanced" cohorts can be loosely thought of as control groups. These groups are based on the program distribution of all the HPL registrants. First, the program ratios of the entire group were calculated (eg, ".15 of the registrants are in AIE"). Then, students are randomly sorted into smaller groups (in this case of size 30) with the same ratios. That is, if a histogram were plotted of the distribution of programs within a cohort, its shape would match a similar histogram of the entire registrant group. The only instance where this principal strayed was in the case of "lone" students:

the cohort algorithm avoids having cohorts in which only one student represents an entire program. Therefore, due to the initial distribution of programs, there may be cohorts in which a smaller program is absent, in order to keep the small numbers of students together.

Similarity-based groups
Similarity is calculated based on cosine similarity. Students are represented as vectors, containing information about their program, years of employment, sector, and a 'meta-group' determined by K-means clustering. These vectors are used in the cosine similarity calculation, which produces a similarity matrix. To create the groups, a 'seed' student is chosen randomly, and the rest of the students are ordered in a list based on their similarity to the seed. For similar cohorts, we take the first X students in the ordered list. For dissimilar cohorts, we take the *last* X students from the list. For mixed cohorts, we take X/2 from the top of the list, and X/2 from the bottom. Once a cohort is created, a new seed student is chosen, and the process begins again.

Step-by-Step Instructions

This is as specific as possible. However, with each iteration of the course, things will obviously change. Please refer to the code as you read.

Preparing

1. The code in github is in a Jupyter Notebook. This is an "Integrated Development Environment" (IDE) for Python. The easiest way to install the Jupyter platform on your computer is to install Anaconda - this is a data science platform that will install Python, Jupyter, and a host of other applications.

Creating the cohorts

1. Collect similarity data based on interest survey.
    a. For the Summer 2018 pilot, incoming students were sent an interest survey which asked their name, years of employment, sector, and program. Because it will be required for all students to take the course in 2019, a different approach to collecting this data will most likely be taken.

2. Merge similarity data with the final registrant list.
    a. In 2018, an export from my.harvard was used as the final registrant list. There were field differences between the similarity data and the my.harvard export, which needed to be addressed before merging. See comments in the code for more information.
    b. When importing data into the notebook, make sure you have the correct file path: "`info = pd.read_csv('info copy.txt')`" means that the file 'info copy.txt' is in the same folder as the notebook itself. If it were in a folder named 'data', for example, the Python command would look something like: `info = pd.read_csv('data/info copy.txt')`

3. Subset the merged dataset based on your cohort plan (in 2018, our plan was the table on pg 1). Our plan had two-thirds of the data in balanced groups, hence the command `balanced = data.sample(frac=.66).`

4. Use the balanced groups function to create balanced groups from the subsetted 'balanced' data.

5. The leftover data can be used to test similarity. Use the similarity function to create those cohorts.

6. Merge the new cohort data into the original merged data frame.

7. Export the final data frame to a .csv file with the command: `name_of_your_data_frame.to_csv('destination_file_path/filename.csv')`