HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Summer Term 2017

Project 4 – "Nature-Inspired Algorithms"

https://hpi.de/en/friedrich/teaching/ss17/natinsalg.html

In this project, we will optimize a real-world application using one of the nature-inspired algorithms discussed in class. The goal of the application is to maximize the diversity of a student allocation into collaborative teams.

The data for this project is real data provided to us by the HPI School of Design Thinking. You are given as input a table stored in CSV format in a file named `'project4.csv'`. This table has 321 rows and five columns. Each row corresponds to a student and the columns are as follows:

```
<hash> <Sex> <Discipline> <Nationality> <Semester>
```

The `<hash>` field contains a cryptographic hash of the student's name (for privacy reasons). The sex field contains `'m'` for male and `'f'` for female. the `Discipline` field contains one of the following seven entries: `'Business'`, `'Creative Disciplines'`, `'Engineering'`, `'Humanities'`, `'Life Sciences'`, `'Media'` or `'Social Sciences'`.

The `Nationality` field contains one of 37 nationalities, depending on the self-reported nationality of the student. The `Semester` field contains the semester in which that student was enrolled. This is stored as a code that indicates the semester and year. For example, the students in Winter 2015 semester have `WT-15` (for Winter Term), and the students enrolled in this semester have the code `ST-17`.

Assignment. Given the data described above, you are asked to compute allocations of students to teams of a given size that maximize diversity within teams and across teams. A *semester teaming* is an allocation of students to teams in a semester.

Each semester contains 80 unique* students except for Summer 2017, which contains 81. For each semester, you will assign 16 teams of five members each (for Summer 2017, you will create 15 teams of five members and one team of six members). A semester teaming is stored as a table with 321 rows and three columns, where each row is as follows:

```
<hash> <team> <semester>
```

where `<hash>` contains the unique student hash, `<team>` is a number in $\{1, \ldots, 16\}$ assigned to that student for that semester, and `<semester>` is the semester code. The count of rows with any given team number and semester must be five (or possible six for one of the team numbers in `ST-17`).

*Note that there is exactly one repeated hash in the data: `ab378874b45dc35e0f4bc2e410686d6b`. This is not a hash collision, but corresponds to the same student who was enrolled both in the Winter Term of 2015 and the Summer Term of 2016. This will not impact the results.

Teaming 1. (Arbitrary teaming). In the first part of the project, you will compute a simple non-optimized semester teaming. This requires no optimization, only a way of managing and formatting the data and serves as a warm-up and sanity check for later work.

Produce a table with an arbitrary semester teaming in a 321×3 table formatted as described above. Be sure to check the correctness of your data, i.e., make sure that the `<hash>` and `<semester>` match and that the above constraints on team size for each semester are respected. Store this table in a file called `'teaming1.out'`.

Teaming 2. (Intra-team diversity). In this part, you will create a semester teaming like Teaming 1, except now you will optimize a measure of diversity within teams. A *diverse* solution prefers to have different nationalities and different disciplines within a team, as well as a good gender balance within the team.

Design your own diversity measure for a semester teaming. Your metric should take into account the `Sex`, `Discipline` and `Nationality` fields.

To perform the optimization, you can use any algorithm we discussed in class. You can formulate it as a single-objective or multi-objective optimization problem, depending on your diversity metric. Use your algorithm to produce a semester teaming that optimizes diversity. This should be stored in a 321×3 table in the same format of Teaming 1 and written to a file named `'teaming2.out'`.

Teaming 3. (Inter-team diversity). We now compute a new semester teaming, except we want to add a fourth diversity objective (in addition to sex, discipline and nationality). The objective is to avoid assigning two people in a team who *were already assigned the same team in Teaming 2*. Every pair x, y of students that are in the same team (and semester) in Teaming 2 and Teaming 3 are called a *collision*. We thus want to optimize *inter-team* diversity by creating Teaming 3 to minimize collisions with Teaming 2. Modify your objective function to capture this, but keep in mind that we still want to preserve *intra-term* diversity in Teaming 3. Use the same optimization algorithm as you used to create Teaming 2. Again, store your results in a 321×3 table and write it to a file named `'teaming3.out'`.

Teaming 4. (Double inter-team diversity). Create a new semester teaming that has intra-team diversity (with respect to sex, discipline and nationality), but also minimizes collisions with *both* Teaming 2 and Teaming 3. Write this to a file named `'teaming4.out'`

Report. Prepare a short (5-10 page) report to discuss your results. Be sure to cover the following points.

(a) Describe the intra-team diversity measure you came up with for Teaming 2. Do you think it accurately captures the stated objective? What is the value of the measure for your solution to Teaming 2? What is the intra-team diversity value for your non-optimized Teaming 1? Discuss your findings.

(b) Describe the inter-team diversity measure you came up with for Teaming 3 and Teaming 4. What are the values for the teamings? How many collisions with Teaming 2 does your solution for Teaming 3 contain? How many collisions with Teaming 2 and Teaming 3 does your solution for Teaming 4 contain?

(c) Describe the optimization algorithm you chose and discuss why you chose it. What parameters did you use? How many iterations did you run it?

(d) Discuss anything pertinent to the project. Did you learn anything new?

Important!

Each team submits **THREE** files. The below example assumes a team with three members where `LastnameX` denotes the last (family) name of a team member.

1. A zip file named `LastnameA-LastnameB-LastnameC-OUTPUT.zip` that contains the four teaming output files: `'teaming1.out'`, `'teaming2.out'`, `'teaming3.out'` and `'teaming4.out'`.

2. A zip file named `LastnameA-LastnameB-LastnameC-CODE.zip` that contains your code;

3. A pdf file, named `LastnameA-LastnameB-LastnameC-REPORT.pdf` that contains the report outlined above.