SCOOP: Project guidelines

Emanuele Di Buccio and Michele Schimd

April 19, 2018

The project consists of the development of a C# project for the analysis of *data* through the usage of data analysis tools and algorithms (*e.g.*, data mining, machine learning).

The project is **mandatory** and must be carried on in groups of $2-3$ persons, it will be presented during the exam session but must be **submitted 5 work days before the presentation**.

Each group must submit the entire code (working and compiling) along with all the necessary to run it (libraries, data, ...). The code represents only a part of the project work, each group is required to submit a report (as a `pdf` file) which contains all details about the problem, the library and/or algorithms used and results obtained (see section below for details on final report).

Evaluation

The final grade on the project (which impacts the final grade of the whole exam), is (roughly) based on the following proportion

Report 35%

Presentation and discussion 30%

Code and implementation 35%

Each of these part **must** be submitted / given in order to pass the final exam (for example projects with report but not code will be considered failed).

Goal

The goal of the project is to make students work and familiarize with the following concepts (explained during lectures).

Programming with the target language C# as well as with the scripting language VBA Visual Basic Application (optional but may give higher grade).

Software development and engineering. Students are supposed to design the entire software from scratch but could use libraries (other than the .NET ones) for most of the heavily algorithmic parts (*e.g.*, libraries can be used for machine learning techniques).

Modeling and design Core part of the project must be the design and implementation of data analysis techniques for: loading and filtering (preprocessing) of data, design or application of known data analysis techniques to the specific problem.

Presentation The work must be presented by the entire group during the discussion sessions. All the group members should know the entire work carried out within the project and should be able to answer questions on all the project parts.

Final report

The final report must be a `pdf` document no longer than 10 pages (single column, 10pt) discussing the following aspects:

Problem The real world problem (for example forecasting of market trends) must be clearly explained keeping in mind that the reader may not be an expert of the subject.

Data representation Real world problems usually do not immediately translates into usable data. Moreover data downloaded from the Internet (usually) contains much more information then needed and even *noise* (*e.g.*, outliers). How all the data needed is represented in terms of the software must be clearly explained in the final report (possibly justifying all choices like, for example, why trees has been used instead of arrays).

Statistical (mathematical) model Usually data analysis resorts to statistical techniques to extract information out of data. There is no *universal* method to solve every possible statistical problem (*i.e.*, no free-lunch theorem). Which method is used and why it has been chosen must be indicated the report giving appropriate references (*e.g.*, paper(s) that successfully applied such techniques).

Software This part should provide an overview of the software components and should make explicit how the software has been designed in order to address the considered problem. Provide a concise description of the classes constituting the software, their interaction, and their role in the solution of the problem.

Evaluation and Results After having implemented the software it must be run on appropriate data (possibly real data obtained from the web). The report must contain only relevant result and experiment (remember you have a very limited amount of pages) **without discarding "bad" results**. We are not asking you to come up with publishable data (better if you will though...). We need to test your ability to develop software using libraries as well as designing part of the software.

Conclusions The report must contain a final paragraph ($1 - 1/2$ page) where the results are discussed and possibly suggesting what could have been done better. For example in the conclusion part a group may notice that having not filtered (or more aggressively filtered) data could have been reduced the execution time and/or increased the overall performance of the software. This part can also include remarks on the limitation of the proposed implementation, limitations or issues of the adopted third party libraries, and issues addressed during the project development.