

## PROJECT DESCRIPTION – SOFTWARE OPTION

This version of the project consists of designing, coding, testing, and documenting statistical software. Alternatively, you can also add functionality to existing statistical software or add functionality to non-statistical software that would be useful to statisticians. I strongly encourage you to undertake projects in teams of two or three students but, in such cases, the project should be more ambitious. You can choose whichever programming language is most appropriate for the project including low level languages (Fortran, C, C++, Java, etc.) as well as high level languages (R, Ruby, Matlab, Python, SAS, VIM, BASH, etc.). My recommendation would be to use a high level language if you are not already familiar with low level languages.

The scope of the project is wide open. As you think about what project you might choose, consider some *examples* of the types of projects that might be appropriate:

- An R package that:
  - Performs a statistical analysis (preferably one that is not currently implemented). See the CRAN website for a list of packages done in the past, or
  - Produces HTML and LaTeX tables from objects in R. See the xtable package.
- A Ruby package that:
  - Implements some statistical methodology such as linear models and/or generalized linear models, etc., or
  - Provides functionality useful for statisticians, or
  - Provides for interactive data analysis similar to that of R.
- Software that integrates two environments. See, for example:
  - Software that integrates R and Python. See RPy, for example, or
  - Software that integrates R and Ruby (either one way or both ways), or
  - Software that facilitates the use of R in KDE, Gnome, or Microsoft Windows.
- A library written in C to perform some statistical calculation:
  - A library (written in whatever language) for multivariate distributions providing for the evaluation of the density and realizing random vectors and/or matrices.

There is a tendency to embark on a project that is too ambitious. I would rather see a straight forward project done well than a grandiose project that is not fully completed. Also, projects that are unique are preferred to those that merely reimplement existing software.

The project consists of four components:

- The initial report is a brief description of what you are going to do and how you propose to do it. You should indicate the software language or environments, what input will be required, and what output will be available. Providing pseudo-code or a proof of concept is encouraged. The report should be written in  $\LaTeX$ . Commit your initial report (both LaTeX and PDF versions), the original article if there is one, and any necessary supporting files to the Subversion repository (specifically into “students/UIN/project/initial-report”) by the start of class on October 18. (For those in teams, all files should be submitted under the “team leaders” area of the repository, with all other team members simply committing a “README.TXT” file indicating the UIN of their team leader.) Proportion of project grade: 15%.
- The final written report must include the code with documentation for both the end user and a programmer you may wish to modify/adapt your code. Also provide the results of test runs that you made. The report should be written in  $\LaTeX$ . Commit your final report (both LaTeX and PDF versions) and any necessary supporting files (that are not already in the initial report) to the Subversion repository (specifically into “students/UIN/project/final-report”) by the start of class on December 7. (For those in teams, all files should

be submitted under the “team leaders” area of the repository, with all other team members simply committing a “README.TXT” file indicating the UIN of their team leader.) Proportion of project grade: 45%.

- The final oral report presented to the class during the final weeks of semester. Commit an electronic version of your oral report to the Subversion repository (specifically into “students/UIN/project/oral-report”) by the start of class on the day of your presentation. (For those in teams, all files should be submitted under the “team leaders” area of the repository, with all other team members simply committing a “README.TXT” file indicating the UIN of their team leader.) Proportion of project grade: 25%.
- Finally, attendance at the oral reports of your colleagues is required. Proportion of project grade: 15%.