Project Report guidelines

The project reports should demonstrate several different features of either PROC SQL, macro variables and macro language, or some combination of all three. The reports can also use methods learned from STAT 540, particularly those that can enhance the appearance of output (E.g., ODS, PROC REPORT, PROC TABULATE, and STYLE options). The Chapter 11/12 bridge presentation could serve as a reasonable guide for the project, though I would anticipate enhancement in the output, as well as possible enhancements to the macro code.

Though the project will likely focus on data management, it is certainly reasonable to include analyses. In fact, we have included straightforward analyses throughout the course and more complex analyses are certainly possible. I downplay analytical issues here, because they could lead to an overly complex project.

Project Selection

Of course, some students' choices of project topic are arbitrary, but when possible I encourage students to choose a topic directly relevant to their personal/work/professional interests. This can mean revisiting old data sets/analyses, or applying methods from this class to current/future projects. Given the focus of this class on data management, almost any project involving a large data set could be suitably adapted to the techniques we've learned in this course. This is quite different from the projects in most of my other statistics courses, in which the match between the student's specific needs and the course material can be problematic.

Though class data sets are not eligible for the project, the size of the data sets should give you some idea of the scope of the data you choose for your project. It is not necessary to have a huge number of independent variables, though it would generally be a good idea to have at least one of each of the following: a numeric variable, a classification variable, an ID variable, a date/time variable, and a character variable. You will find you will want a variety of variables to adapt techniques you have learned in class.

Past student projects have included:

- Use macros to read in and stack multiple Excel worksheets containing traffic county data for each SC county in a single file, then match merge the file with another data set in PROC SQL, then export all the files separately as Excel files.

- Use PROC SQL and macros to format, analyze, and create reports on the amount of energy used by SCE&G customers.

- Use macros and data management tools to analyze rates of primary ACL (anterior cruciate ligament) reconstruction and revision completed by surgeons in South Carolina.

- Adapt a macro used to assess model fit for nested hierarchical linear models so that it assesses model fit for non-nested hierarchical linear models.

- Designing an end-of-course exam.
- Re-organize EEG data into a manageable set of tables for exploratory data analysis (the data appeared as a series of subtables in a single Excel workbook).
- Create macros using PROC SQL to automate summary reports for survey response data.
- Combine School Climate Survey data from multiple schools into a single data set
- Write a macro to simulate dice rolls for role-playing games.
- Analysis of MATH 111 scores at USC-Beaufort based on high-school preparation
- Analysis of leading causes of death in SC counties

Introduction

The first part of your report can include your motivation for the project (as suggested by the preceding discussion). It is perfectly fine for the discussion here to be personal or anecdotal. Do not feel you have to leave context out of the report. The first part of your report should also introduce the data, methods, or coding problems that precipitated your interest in your project. It should also discuss your ultimate goal, as initially envisioned (though this may evolve–see below).

You have the option of introducing specific code or data sets at this point; the discussion of a specific example can help explain why you became interested in a method or application in the first place. Use your discretion.

Software demonstration/coding

Development of complex programs is sequential and evolutionary. Be sure to break down complicated procedures into their constituent steps, and to demonstrate the steps with subsets of PROC SQL code, macro code or supplementary code. It would be reasonable to take the same approach to each of your separate steps as we take with the class exercises. At each step, explain whether the code worked the way you anticipated. Identify areas of improvement and next steps. Discuss obstacles and don't worry about including mistakes. The focus of the project is process, process, process...

You will find that your ultimate aim may evolve as your project proceeds–it will be important to document and discuss major obstacles and how you resolve them. If you find just the right solution to a particular problem, that's great. It's also perfectly reasonable to come up against a seemingly intractable programming problem, or one that threatens to overtake much of your effort on the project, and make some modifications to the project. Often these modifications involve a change in your goal, possibly by simplifying code and output. Or you may redesign your data set so that it is more flexible or tractable for the problem at hand. Be straightforward and honest about these obstacles in your write-up.

At the end of the process, you should have a complete set of code that you demonstrate on your full data set, along with the product of the code, whether it is a PROC SQL report, data sets or tables, ODS output, etc.

Be sure to write a conclusion summarizing your project and reflecting on what you learned. Include suggestions for improvement or enhancements to your final product, and discuss their feasibility.