

## EXST 7005

Fall 2010

# Lab #1: Introduction to Basic SAS Operations

---

**TA: Suping Zhang**

**Email: [szhang5@tigers.lsu.edu](mailto:szhang5@tigers.lsu.edu)**

**Office: Room 71A, Ag Admin Building**

### LAB POLICY

- Each lab report is worth 10 points. Usually, lab reports are due at the beginning of the next lab. If your lab report is late, you will lose 1 point for each day after the due time.
- For each lab report, please make your name, section number and due date clear on the first page.
- Each lab report should conclude three parts: 1) your answers to the questions (when applicable), 2) SAS output (not all of it, only the parts which are directly related to your answers) and 3) SAS log.

### OVERVIEW OF SAS

In SAS there are three main windows: Program Editor, Log, and Output.

**Program Editor:** where you type in your SAS program.

**Log:** gives you information about possible errors in your SAS program.

**Output:** gives you a printout of what your SAS program did.

### SAS PROGRAM

A SAS program mainly consists of the following parts: optional step; data statement; proc steps;

#### OPTIONAL STEP

```
dm 'log;clear;output;clear';  
title "Exam1 grades";  
options nodate nocenter pageno=1 ls=78 ps=55;
```

1) The first statement above tells SAS to clear the log and output windows each time the SAS program is submitted.

2) The second statement adds a title to the top of every page of output.

3) The third statement allows you to specify options: a) The **ls** option changes the number of characters printed across the page, and can take on values between 64 and 256. b) the **ps** option changes the number of lines printed down the length of the page. The values shown above for these two options are typically good for 8.5 X 11" paper. c) **pageno=1** resets the starting page number at 1.

### DATA STEP

```
data GRADES;
input NAME $ EXAM1A EXAM1B;
DIFF= EXAM1A-EXAM1B;
datalines;
David 78 92
Ann 85 82
Carol 91 91
Susan 87 95
Kyle 82 88
;
run;
```

- 1) The **data** step begins with the **data** statement. The **data** statement tells SAS to create a new dataset named "GRADES."
- 2) The **input** statement specifies what variables are being input. The \$ sign indicates a character variable which is not needed for numerical variables.
- 3) A new variable "DIFF" is created by manipulating the existing variables. It will be stored in the dataset "GRADES."
- 4) "**datalines**" or "**cards**" indicates that the data is typed directly into the program editor. (Importing data from other locations will be discussed later.)
- 5) Once all of your data has been entered, you then type a semicolon on the next line.

### PROC STEPS

```
proc print data=GRADES;
run;

proc sort data=GRADES;
by NAME;
run;

proc univariate data=GRADES plot;
var EXAM1A;
run;
```

- 1) In SAS, every specific data analyzing procedure starts with "**proc**".

- 2) **“proc print”** statement simply prints out all the sorted data, including the new variable we have just created. **“proc sort”** statement will sort the data by the values of a certain variable. The default order is ascending, but it can be changed by entering **“descending”** in the **“by”** statement before the variable name. **“proc univariate”** statement is used to request a variety of statistics for summarizing the data distribution of each variable. **“plot”** at the end of this statement is used to request a normal plot; If you use **“var EXAM1A”**, it will only summarize this specific variable; otherwise all the variables will be analyzed.
- 3) It is important to specify which dataset you are using, especially when there are multiple datasets. Or SAS automatically uses the most recently used data set,

#### OTHERS

- SAS does not tell capital letter or small letter. You can use either one;
- Comment: Additionally, you may add comments anywhere in your program either by beginning the statement with an asterisk (\*) and ending it with a semicolon (;) or by beginning with /\* and ending with \*/. These comments may be thought of as marginal notes, and will appear in the log and editor, but not in the output.
- Shortcuts: **F4**: recalls text once in program editor; **F5**: directs you to the program editor; **F6**: directs you to the log; **F7**: directs you to the output; **F8** or the **“running man”**: submits your program; **Ctrl+E**: clears content in the current window.

#### Assignment (Due in the next week before lab)

The following dataset is drawn from a housing database of a mid-sized city in east Texas.

OBS	AGE	SIZE	PRICE	EXTERIOR
1	21	951	30000	frame
2	21	1036	39900	brick
3	7	676	46500	brick
4	6	1456	48600	other
5	51	1186	51500	brick
6	19	1456	56990	frame

- 1) Input the dataset into SAS. (2 points)
- 2) Print this dataset.
- 3) Create a new variable **“AverPrice”** which is the average price per sq ft for each house. (Hint: use **“/”** to divide.)
- 4) Sort the dataset in a **descending** order according to **“AverPrice.”**
- 5) Find the mean for the **PRICE** and draw a normal plot for it.