STAT8510 – Programming for Data Analysis (SAS Section)

Lifang Zhang, MD. MS

Phone: 706-721-4453

Email: lizhang@augusta.edu

Text Book: ***The Little SAS Book*** *by LD Delwiche and SJ Slaughter.*

Grading:	Homework	40%
	Hands-On Attendance/Participation	20%
	Programming Project	40%

Grades will be assigned on the following scale: 90-100=A, 80-89=B, 70-79=C, 60-69=D, and <60=F.

Class notes and examples, hands on, and Homework assignments will be uploaded into D2L

What SAS Can Do?

SAS can do: Statistical analysis; Data manipulating; Graphics; Online data entry;….

Usage of SAS: SAS programmers are needed in all areas.

1, Medical and public health (examples),

2. Financial company (banks, accountings),

3. Pharmaceutical company.

4, Educational Institute.

5, Power Company.

6, Fire house.

7, Industry research…..

SAS and R

	R vs. SAS	
	R	SAS
Basic Structure of Commands	keyword (essential command(s), options)	PROC Keyword; Commands; Run;
Case Sensitive	Yes	No
Commons	# comments	*comments; or /*comments*/
Semi-colons at end of text	No	Yes
Assignments	x<-2*3	x=2*3
How to submit commands	Hit Enter or Return to run each lines of code	Click on "Run" to run all program or Highlight code, click on "Run"

SAS and R

Pharmaceutical companies require use SAS for program analysis and report, use R for design study. Because SAS has essentially the "full faith and credit of the SAS Institute" which has a pretty solid track record. R is Free.

Some packages in R are amazing. Some packages are not. You have to go find them, evaluate them, and even then there's some leap-of-faith issues in that the package is only as good as the guy writing it. It's hard to trust that.

SAS Windowing Environment

There are five basic SAS windows: The Results and Explorer windows, and three programming windows: Editor, Log, and Output. There are also many other SAS windows that may use for task such as getting help; system options….

Editor – You use this window to type in, edit, and submit SAS programs. The default editor is the Enhanced Editor. The Enhanced Editor is syntax sensitive and color codes your programs making it easier to read and find mistakes. The Enhanced Editor also allows you to collapse and expand the various steps in your program.

-Continue next page

SAS Windowing Environment

- **Log** – The Log window contains notes about your SAS session, and after you summit a SAS program, any notes, error, or warnings associated with your program as well as your program statements themselves will appear in the log window.
- **Output** – Any printable output will appear in Output window.
- **Results** – The results tree lists each part of your results in an outline form.
- **Explorer** – The explorer window gives you easy access to your SAS file and libraries.

The SAS Language

SAS programs is a sequence of statements executed in order.
SAS Statements have certain rules to follow:

1. Every SAS statement ends with a semicolon!
2. SAS statements can be in upper- or lowercase.
3. Statements can continue on the next line.
4. Statement can be on the same line as other statements.
5. Statement can start in any column.

SAS Comments

Comments is to make sure your programs are more understandable. It doesn't matter what you put in your comments – SAS doesn't look at it! There are two styles of comments you can use:

1. Start with an asterisk (*) and ends with a semicolon.

2. Start with a slash asterisk (/*) and ends with an asterisk slash (*/).

SAS Data sets

Variables and observations:

SAS data consist of variables and observations. SAS data sets are also called tables, observations are called rows, and variables are called columns or fields.

SAS Data types:

There are only two data types for SAS: Numeric and Character.

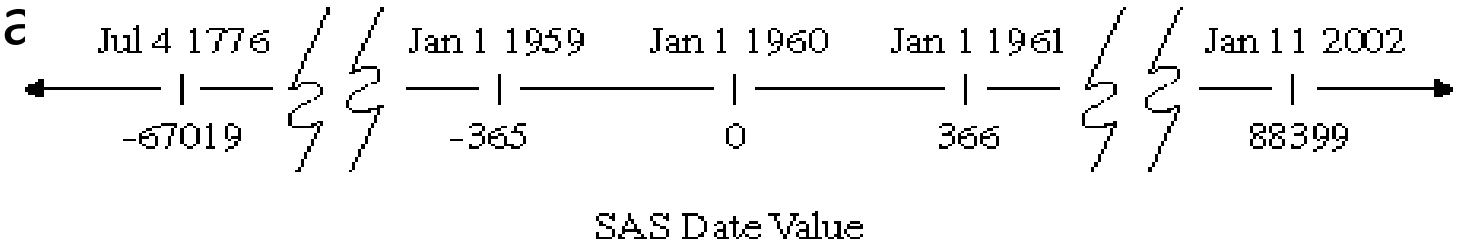
Numeric variables are numbers that can be added and subtracted, and can have any number of decimal places, and can be contain plus sign (positive), minus sign (negative), decimal points…

SAS Data sets *- Continue*

Character variables are everything else. Can ZIP code be a Character?

What about date and time? How SAS Converts Calendar Dates to SAS Date Values?

Da ers;
da



SAS Data sets *- Continue*

SAS time value

is a value representing the number of seconds since midnight of the current day. SAS time values are between 0 and 86400.

12:00am = 0

1:00am = 60x60=3600 Sec

Missing Data:

Missing character data are represented by blanks, and missing numeric data are represented by a single period (.).

Rules for SAS names (Variable)

Names must be 32 characters or fewer in length.

Name must start with a letter or an underscore (_).

Name can contain only letters, numerals, or underscore. No %$#@*&^

Name can contain upper- and lowercase letters.

The Two Part of a SAS Program

SAS programs are constructed from two basic building blocks:

1, DATA step to create SAS data set.

2, PROC step for processing data and statistical analysis.

Once you start PROC, SAS assume that you have finished your data manipulation, and you can not do any data manipulating after PROC until you start a new DATA step.

DATA Entering

There three common ways to get your data into SAS:

1. Entering data directly into SAS data sets using INPUT Statement

2. Reading other software's data files (et. al. Excel data, Access data..) directly.

3. Converting other software's data files into SAS data sets using a data transformation software

Today, we will discuss the FIRST way of input data - **Entering data directly into SAS data sets**

DATA Entering

DATA name;

INFILE “path\filename”;(If data is in txt or any other form)

INPUT list of variable name ;

DATALINES (or CARDS); (If entering the data manually)

.

Entering data here

.

;

END; ** is not necessary.

DATA INPUT

1, LIST INPUT (Raw Data Separated by Space): If the values in your raw data file are all separated by at least one space, then use the list input. SAS automatically scans to the next non-blank field and start to read.

2, COLUMN INPUT (Data Separated by Columns): If raw data files do not have space (or other delimiters) between all the values – so the files can't be read using list input, column input can be used if each of the variable's values is always found in the same place in the data line. With column style input, SAS start reading in the exact column you specify.

. Space are not required between values

. missing value can be left blank

. character data can have space

. you can skip unwanted variables

You can use Column pointer **@n** to tell SAS to start reading a specific column.

DATA INPUT - *Continue*

3, FORMATTED INPUT: Sometime raw data are not straightforward numeric or character. For example, date, time….informats are useful when you have non-standard data.

What is standard numeric? – Numerals, decimal points, minus signs, E for scientific notation)

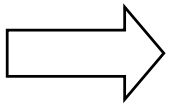
So numbers with embedded commas or dollar signs are example of non-standard data. Dates are perhaps the most common non-standard data. SAS will convent conventional forms of dates like 10-31-2003 or 31OCT03 into a number of days since January 1, 1960. This number is referred to as a SAS date value. (Why January 1, 1960? Who knows? Maybe 1960 is a good years for SAS founder?)

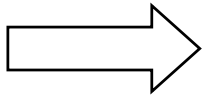
There are three general types of informats: character, numeric, and date.

Character	Numeric	Date
$ informat*w.*	informat*w.d*	informat*w.*

DATA INPUT - *Continue*

	Informat		Format	
Jan 01, 1960		0		Jan 10, 1960
01/12/1960		11		01/12/1960
02-13-1960		43		02/13/1960
$4,000		4000		$4,000
12:00 (12 at noon)		43200		12:00





DATA INPUT - *Continue*

4, MIXED INPUT: Each of the three major input styles has its own advantages. SAS is very flexible that you can mix and match any of the input styles for your own convenience.

5, Reading multiple Obs. per line: Normally SAS assumes that each line of raw data represents no more than one observation. When you have multiple observations per line, you can use double trailing at sign (@@) at the end of your INPUT statement.

DATA INPUT - *Continue*

6, Reading multiple lines of raw data per observation: A slash (/) and pound-n (#n) are called pointers. They are like road signs telling SAS, “Go this way.” To read more than one line of raw data for a single observation, you simply insert a slash into your INPUT statement when you want to skip to the next line of raw data. The #n line pointer performs the same action except that you specify the line number.

7, Reading Part of a Raw Data File: @ can also be used as line holder for other input.