

Statistical Methods and Computing, STAT:2010/4200

Instructor: Cowles

Lab 3

Feb. 19, 2016

1 Using formats to get SAS to print something other than the values a variable actually contains

We will be using the billionaire dataset again today. Its variable called `region` contains abbreviations (“A” for Asia, “E” for Europe, etc.). If we want SAS to print out the complete words instead of the abbreviations, so that tables and graphs are more understandable, we need to run a “proc format” before the data step. The data step must then refer to the formats defined in the format procedure.

The `formchar` option is needed when printing tables as we will be doing later in the lab. You can copy it in from the file `formchar.sas` in the Datasets listing.

```
options linesize = 75
formchar = "|----|+|----+|-/\<>*" ;
```

```
proc format ;
value $regfmt 'A' = 'Asia' 'E' = 'Europe' 'M' = 'Middle East'
              'O' = 'Other' 'U' = 'US' ;
run ;
```

Because the original data values in the `region` variable are characters rather than numbers, we have to use a dollar sign as the first character in the name of the format.

Note the format statement in the data step below. It tells SAS to apply the format you have defined here to a particular variable. When you use the format statement in a data step, you must put a period at the end of the format name.

2 Using labels to get SAS to print more descriptive variable names

```
data billion ;
input wealth age region $ ;
format region $regfmt. ;
label wealth = 'Wealth in Billion $'
      age = 'Age in Years' ;
datalines ;
< copy and paste data in here>
;
run ;
```

Now enter and run the following code to see how the formats and labels affect the output of the “print” and “freq” procedures.

```
proc print data = billion (obs = 20);
run ;

proc print label data = billion (obs = 20);
run ;

proc freq data = billion ;
```

```
tables region ;
run ;
```

3 Formats for numeric variables

Formats can also be used to group numeric data. Add a line to your format procedure and change one line in the data step as follows:

```
proc format ;
value $regfmt 'A' = 'Asia' 'E' = 'Europe' 'M' = 'Middle East'
              'O' = 'Other' 'U' = 'US' ;
value amtfmt low<5 = '<5' 5-<10 = '5-<10' 10-<20 = '10-<20' 20-high = '20+' ;
run ;

data billion ;
input wealth age region $ ;
format region $regfmt. wealth amtfmt. ;
label wealth = 'Wealth in Billion $'
      age = 'Age in Years' ;
datalines ;
<data>
;
run ;
```

To see the effect of adding this format:

```
proc print data = billion ;
run ;

proc freq data = billion ;
tables wealth ;
run ;
```

4 Proc means

Gulanick (*Heart and Lung*, 1991) studied patients who were recovering from heart surgery. She was interested in whether different combinations of supervised exercise or teaching would affect patients’ self-efficacy (or confidence) to perform physical activity.

Patients were randomly assigned to one of three groups. Group 1 received teaching, treadmill exercise testing, and exercise training three times per week. Group 2 received only teaching and exercise testing. Group 3 received only routine care without supervised exercise or teaching. After 4 weeks, each patient was scored on self-efficacy.

Self-efficacy was measured on a continuous scale and scores were assumed to be distributed normally in each of the populations of interest. Her results are in the dataset “gulanick.dat.” We wish to produce a table that shows the number of observations and the mean and standard deviation of scores within each of the three groups.

```
proc format ;
value grpfmt 1 = 'Teaching and Training' 2 = 'Teaching' 3 = 'Neither' ;
run ;
```

```

data gulan ;
input score group ;
format group grpfmt. ;
datalines ;
< copy data in here>
;
run ;

```

We can use *proc means* to get various summary statistics in a more compact format than *proc univariate* provides. The default statistics provided are

- n = number of observations
- mean
- std dev = standard deviation
- minimum
- maximum

```

proc means data = gulan ;
var score ;
where group eq 1 ; * restricts to those records with group = 1 ;
run ;

```

The MEANS Procedure

Analysis Variable : score

N	Mean	Std Dev	Minimum	Maximum
11	126.8181818	24.2520852	100.0000000	170.0000000

5 Confidence intervals

We can request other descriptive statistics by specifying them as part of the *proc means* statement. One that you will need soon is the *confidence interval* for the mean. Putting “clm alpha = .05” on the end of the *proc means* statement produces a 95% confidence interval.

The MEANS Procedure

Analysis Variable : score

N	Mean	Std Dev	Lower 95% CL for Mean	Upper 95% CL for Mean
11	126.8181818	24.2520852	110.5254093	143.1109543

6 Using proc tabulate to summarize the distributions of quantitative variables in different groups

```

proc tabulate data = gulan ;
class group ; * class statement identifies qualitative variables ;
var score ; * var statement identifies quantitative variables ;
tables group , score * (n mean std) ;
run ;

```

	score		
	N	Mean	Std
group			
Teaching and Training	11.00	126.82	24.25
Teaching	12.00	128.42	25.04
Neither	13.00	103.92	17.71

7 More on proc means

The following code will produce means for this dataset of the values in the variables “wealth” and “age.”

```

proc means data = billion n mean ;
var wealth age ;
title 'Average Age and Wealth of 1992 Billionaires' ;
run ;

```

The MEANS Procedure

Variable	Label	N	Mean
wealth	Wealth in Billion \$	233	2.6815451
age	Age in Years	225	64.0311111

Average Age and Wealth of 1992 Billionaires

Since our dataset contains an observation for every billionaire in the world in 1992, if the population of interest is billionaires in 1992, is this a population mean or a sample mean?

If we want a separate mean for each region, we must first make sure that the dataset is sorted in order by region, and then run “proc means” with an additional “by” statement.

```

proc sort ;
by region ;
run ;

```

```

proc means data = billion n mean ;
var wealth age ;
by region ;
title 'Average Age and Wealth of 1992 Billionaires' ;
title2 'By Region' ;
run ;

```

Average Age and Wealth of 1992 Billionaires 7
By Region 10:19 Friday, March 14, 2014

----- region=Asia -----

The MEANS Procedure

Variable	Label	N	Mean
wealth	Wealth in Billion \$	38	2.6210526
age	Age in Years	37	63.6486486

----- region=Europe -----

Variable	Label	N	Mean
wealth	Wealth in Billion \$	80	2.2075000
age	Age in Years	76	64.0263158

----- region=Middle East -----

Variable	Label	N	Mean
wealth	Wealth in Billion \$	22	4.2636364
age	Age in Years	22	64.2272727

----- region=Other -----

Variable	Label	N	Mean
wealth	Wealth in Billion \$	29	2.2344828
age	Age in Years	28	64.1428571

----- region=US -----

Variable	Label	N	Mean
wealth	Wealth in Billion \$	64	2.9687500

age Age in Years 62 64.1451613

```

proc means data = billion ;
class region ;
var wealth age ;
run ;

```

region	Obs	Variable	Label	Maximum
Asia	38	wealth	Wealth in Billion \$	14.0000000
		age	Age in Years	83.0000000
Europe	80	wealth	Wealth in Billion \$	11.7000000
		age	Age in Years	96.0000000
Middle East	22	wealth	Wealth in Billion \$	37.0000000
		age	Age in Years	102.0000000
Other	29	wealth	Wealth in Billion \$	6.2000000
		age	Age in Years	84.0000000
US	64	wealth	Wealth in Billion \$	24.0000000
		age	Age in Years	88.0000000

Proc tabulate for the billionaire data.

```
proc tabulate data = billion ;  
class region ;  
var age ;  
tables region, age * (mean stddev) ;  
run ;
```

	Age in Years	
	Mean	StdDev
region		
Asia	63.65	9.96
Europe	64.03	14.70
Middle East	64.23	19.53
Other	64.14	13.08
US	64.15	11.88