To FREQ, Perchance to MEANS

ay, there's the rub!

Christopher Bost

Choose the right procedure

Run PROC FREQ when…

- variables are categorical
- small number of levels
- numeric or character variables

Run PROC MEANS when…

- variables are continuous
- large number of levels
- numeric variables only

Challenges

- Determine which variables are categorical and continuous
- Type long lists of variable names
  - TABLES statement or VAR statement
- Process is **tedious** and **error prone**

What's in it for you?

You will learn how to

1. Determine the number of levels of each variable
2. Determine the type of each variable
3. Store variable lists in macro variables based on the number of levels and type
4. Use variable lists stored in macro variables with PROC FREQ and PROC MEANS

Sample data set

PRINT of data set SASHELP.CLASS

Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5
6	James	M	12	57.3	83.0
7	Jane	F	12	59.8	84.5
8	Janet	F	15	62.5	112.5
9	Jeffrey	M	13	62.5	84.0
10	John	M	12	59.0	99.5
11	Joyce	F	11	51.3	50.5
12	Judy	F	14	64.3	90.0
13	Louise	F	12	56.3	77.0
14	Mary	F	15	66.5	112.0
15	Philip	M	16	72.0	150.0
16	Robert	M	12	64.8	128.0
17	Ronald	M	15	67.0	133.0
18	Thomas	M	11	57.5	85.0
19	William	M	15	66.5	112.0

FREQ **Sex** **Age**

or

MEANS? **Height** **Weight**

What's in it for you?

You will learn how to

1. Determine the number of levels of each variable
2. Determine the type of each variable
3. Store variable lists in macro variables based on the number of levels and type
4. Use variable lists stored in macro variables with PROC FREQ and PROC MEANS

1. Determine the number of levels

```
proc freq data=sashelp.class nlevels;
tables _all_ /noprint;
run;
```

Results:

```
The FREQ Procedure

Number of Variable Levels

Variable      Levels
--------------------
Name              19
Sex                2
Age                6
Height            17
Weight            15
```

Save number of levels to data set

```
ods trace on/listing;
proc freq data=sashelp.class nlevels;
tables _all_ /noprint;
run;
ods trace off;
```

Results:

```
Output Added:
-------------
Name:         NLevels
Template:     Base.Freq.NLevels
Path:         Freq.NLevels
-------------
```

Save number of levels to data set

```
ods output nlevels=nlevelsds;
proc freq data=sashelp.class nlevels;
tables _all_ /noprint;
run;

proc print data=nlevelsds;
run;
```

Results:

Obs	TableVar	NLevels
1	Name	19
2	Sex	2
3	Age	6
4	Height	17
5	Weight	15

Rules

- 10 or fewer levels → PROC FREQ
- More than 10 levels → PROC MEANS
- What about NAME?
  - Need to know variable **type** as well as number of levels to determine which PROC to run

What's in it for you?

You will learn how to

1. Determine the number of levels of each variable
2. Determine the type of each variable
3. Store variable lists in macro variables based on the number of levels and type
4. Use variable lists stored in macro variables with PROC FREQ and PROC MEANS

2. Determine each variable type

```
proc sql;
select name,type
from dictionary.columns
where libname='SASHELP' and memname='CLASS'; *uppercase;
quit;
```

Results:

```
Column Name                    Column Type
-------------------------------------
Name                           char
Sex                            char
Age                            num
Height                         num
Weight                         num
```

Save variable type+ to data set

```
proc sql;
create table meta as
select name,type,nlevels
from dictionary.columns,nlevelsds
where libname='SASHELP' and memname='CLASS' and
      name=tablevar;
quit;

proc print data=meta;
run;
```

Save variable type+ to data set

Results:

Obs	name	type	NLevels
1	Name	char	19
2	Sex	char	2
3	Age	num	6
4	Height	num	17
5	Weight	num	15

What's in it for you?

You will learn how to

1. Determine the number of levels of each variable
2. Determine the type of each variable
3. Store variable lists in macro variables based on the number of levels and type
4. Use variable lists stored in macro variables with PROC FREQ and PROC MEANS

3. Store variable lists

```
proc sql;

title 'Variables to process with PROC FREQ';
select name
from meta
where nlevels <= 10;

title 'Variables to process with PROC MEANS';
select name
from meta
where nlevels > 10 and type='num'; *lowercase;

quit;
```

3. Store variable lists [cont.]

Results:

```
Variables to process with PROC FREQ

Column Name
--------------------------------
Sex
Age


Variables to process with PROC MEANS

Column Name
--------------------------------
Height
Weight
```

3. Store variable lists [cont.]

```
proc sql noprint;

select name into :FREQvars separated by ' '
from meta
where nlevels <= 10;

select name into :MEANSvars separated by ' '
from meta
where nlevels > 10 and type='num'; *lowercase;

quit;
```

Inspect values [optional]

```
%put FREQvars=&FREQvars;
%put MEANSvars=&MEANSvars;
```

Results [Log]:

```
FREQvars=Sex Age
MEANSvars=Height Weight
```

What's in it for you?

You will learn how to

1. Determine the number of levels of each variable
2. Determine the type of each variable
3. Store variable lists in macro variables based on the number of levels and type
4. Use variable lists stored in macro variables with PROC FREQ and PROC MEANS

4. Use variable lists

```
proc freq data=sashelp.class;
tables &FREQvars;
run;

proc means data=sashelp.class;
var &MEANSvars;
run;
```

4. Use variable lists [cont.]

Sex	Frequency	Percent	Cumulative Frequency	Cumulative Percent
F	9	47.37	9	47.37
M	10	52.63	19	100.00

Age	Frequency	Percent	Cumulative Frequency	Cumulative Percent
11	2	10.53	2	10.53
12	5	26.32	7	36.84
13	3	15.79	10	52.63
14	4	21.05	14	73.68
15	4	21.05	18	94.74
16	1	5.26	19	100.00

Variable	N	Mean	Std Dev	Minimum	Maximum
Height	19	62.3368421	5.1270752	51.3000000	72.0000000
Weight	19	100.0263158	22.7739335	50.5000000	150.0000000

Rules

Variable Type	Number of Variable Levels	
	10 or fewer	**More than 10**
Character	PROC FREQ	PROC PRINT
Numeric	PROC FREQ	PROC MEANS

Character variables with many levels [NAME]?
→ PROC PRINT

Character vars with many levels

```
proc sql;
select name into :PRINTvars separated by ' '
from meta
where nlevels > 10 and type='char'; *lowercase;
quit;

proc print data=sashelp.class(obs=5); *adjust N as needed;
var &PRINTvars;
run;
```

Results:

Obs	Name
1	Alfred
2	Alice
3	Barbara
4	Carol
5	Henry

Final program

```
ods output nlevels=nlevelsds;
proc freq data=sashelp.class nlevels;
tables _all_/noprint;
run;

proc sql noprint;
create table meta as
select name,type,nlevels
from dictionary.columns,nlevelsds
where libname='SASHELP' and memname='CLASS'
      and name=tablevar;

*store names of all variables with
 NLEVELS <= 10 in macro variable FREQvars;
select name into :FREQvars separated by ' '
from meta
where nlevels <= 10;

*store names of numeric variables with
 NLEVELS > 10 in macro variable MEANSvars;
select name into :MEANSvars separated by ' '
from meta
where nlevels > 10 and type='num';

*store names of character variables with
 NLEVELS > 10 in macro variable PRINTvars;
select name into :PRINTvars separated by ' '
from meta
where nlevels > 10 and type='char';

quit;

proc freq data=sashelp.class;
tables &FREQvars;
run;

proc means data=sashelp.class;
var &MEANSvars;
run;

proc print data=sashelp.class(obs=5);
var &PRINTvars;
run;
```

Conclusion

- The PROC FREQ option NLEVELS counts the number of levels of each variable.
- The Output Delivery System can save this metadata to a SAS data set.
- PROC SQL can check the number of levels and variable type and create macro variables that store respective lists of variables on which to run PROC FREQ and PROC MEANS.
- The process can be automated with a macro.

Contact information

Comments and questions are valued and encouraged.

Christopher J. Bost
MDRC
16 East 34th Street
New York, NY 10016
(212) 340-8613
christopher.bost@mdrc.org
chrisbost@gmail.com