

Overview ○○○○	Informats ○○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Instream data, informats, formats, labels, and PROC FORMAT

Shannon Pileggi

STAT 330

STAT 330: Lecture 6 1 / 24

Overview ○○○○	Informats ○○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

OUTLINE

- Overview
- Informats
- Formats and Labels
- PROC FORMAT

STAT 330: Lecture 6 2 / 24

Overview ●○○○	Informats ○○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Instream data

One way to get data into SAS is to directly type raw data into the DATA step using DATALINES

- ▶ convenient for small set of data
- ▶ values separated by a space
- ▶ both **character** and **numeric** missing data values must be indicated by a period in DATALINES

STAT 330: Lecture 6 3 / 24

Overview ●●○○	Informats ○○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Standard versus nonstandard data

SAS can read *standard* data without any additional instruction

- ▶ character data is always standard (and requires \$)
- ▶ standard numeric values:
58, 67.23, 5.67E5, 00.99, 1.2E-2

Informats provide additional instruction for SAS to **read** in nonstandard data. **Formats** provide additional instruction for SAS to **display** nonstandard data.

- ▶ non-standard numeric values:
(23), \$67.23, 5,823, 1/12/2010, 12May2009

STAT 330: Lecture 6 4 / 24

Overview ○○●○	Informats ○○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Example Code

SAS Code

```
DATA work.class ;
  INPUT name $ GPA ;
  DATALINES;
  Bill  3.4
  Susan 2.7
  ;
RUN ;
```

SAS Code

- ▶ on the INPUT line list the variable names, with any informats *after* the variable name (e.g., \$ comes after name)
- ▶ DATALINES indicates that we are entering data
- ▶ the DATALINES statement must be the **last** statement in the data step.
- ▶ the semi-colon after the data should be on a line by itself

STAT 330: Lecture 6 5 / 24

Overview ○○○●	Informats ○○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Practice

SAS Code

```
DATA work.class ;
  INPUT name $ GPA ;
  DATALINES;
  Bill  3.4
  Susan 2.7
  ;
RUN ;
```

SAS Code

On your own: One at a time, try making

1. Bill's GPA missing
2. Susan's name missing

and verify your output.

STAT 330: Lecture 6 6 / 24

Overview ○○○○	Informats ●○○○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Overview

Informats

Formats and Labels

PROC FORMAT

STAT 330: Lecture 6 7 / 24

Overview ○○○○	Informats ○○●○○	Formats and Labels ○○○○○○	PROC FORMAT ○○○○○○○
------------------	--------------------	------------------------------	------------------------

Example informat for nonstandard data

SAS Code

```
DATA work.class ;
  INPUT name $ GPA dob MMDDYY10. ;
  DATALINES ;
  Bill  3.4 10/13/1995
  Susan 2.7 6/24/1993
  ;
RUN ;
```

SAS Code

On your own:

1. Identify the nonstandard data.
2. Identify the informat.
3. What does MMDDYY10. mean?

STAT 330: Lecture 6 8 / 24

Overview

Formats and Labels

PROC FORMAT

Informats

STAT 330: Lecture 6

9 / 24

Informats

Informats allow us to read formatted data. The general structure of an informat is:

Character: \$name_of_informatw.
 Numeric: name_of_informatw.d
 Date: name_of_informatw.

where

- ▶ w specifies the *complete* string width (including any \$ signs, commas, ...)
- ▶ d specifies the number of decimal places
- ▶ Search: SAS informats!

Overview

Formats and Labels

PROC FORMAT

Informats

STAT 330: Lecture 6

10 / 24

On your own:

SAS Code

```
DATA work.class;
  INPUT name $ GPA dob MMDDYY10. salary ?;
  DATALINES ;
  Bill 3.4 10/13/1995 $18,000
  Susan 2.7 6/24/1993 $535,000
  ;
  RUN ;
```

SAS Code

The COMMAw.d informat removes embedded characters for numeric data.

Which would be the correct specification for salary?

1. COMMA2.3
2. COMMA3.3
3. COMMA8.
4. COMMA.8

Overview

Formats and Labels

PROC FORMAT

Informats

STAT 330: Lecture 6

11 / 24

On your own:

SAS Code

```
DATA work.class;
  <OPTION 1>
  INPUT name $ GPA dob MMDDYY10. ;
  <OPTION 2>
  DATALINES ;
  Bill 3.4 10/13/1995
  Susan 2.7 6/24/1993
  <OPTION 3>
  ;
  <OPTION 4>
  RUN ;
```

SAS Code

Suppose we wanted to identify the day of the week on which they were born. Where should I insert `day=WEEKDAY(dob);` ?

Overview

Formats and Labels

PROC FORMAT

Informats

STAT 330: Lecture 6

12 / 24

Overview

Informats

Formats and Labels

PROC FORMAT

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

SAS dates

SAS Code

```

DATA work.class;
  INPUT name $ GPA dob MMDDYY10. ;
  day = WEEKDAY(dob);
  DATALINES ;
  Bill 3.4 10/13/1995
  Susan 2.7 6/24/1993
  ;
RUN ;

```

SAS Code

SAS stores dates as the number of days since January 1, 1960.

-2	Dec 30, 1959
-1	Dec 31, 1959
0	Jan 1, 1960
1	Jan 2, 1960
2	Jan 3, 1960
7	Jan 8, 1960

On your own: What is the interpretation of 13069?

Obs	name	GPA	dob	day
1	Bill	3.4	13069	6
2	Susan	2.7	12228	5

STAT 330: Lecture 6

13 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

SAS days

SAS Code

```

DATA work.class;
  INPUT name $ GPA dob MMDDYY10. ;
  day = WEEKDAY(dob);
  DATALINES ;
  Bill 3.4 10/13/1995
  Susan 2.7 6/24/1993
  ;
RUN ;

```

SAS Code

On your own: Examine the help file for the WEEKDAY informat. What does a value of 6 mean?

Obs	name	GPA	dob	day
1	Bill	3.4	13069	6
2	Susan	2.7	12228	5

STAT 330: Lecture 6

14 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

SAS formats (variable *value* display)

SAS Code

```

PROC PRINT DATA = work.class ;
  FORMAT dob DATE9.
  day WEEKDATE9. ;
RUN ;

```

SAS Code

SAS Code

```

DATA work.class2 ;
  SET class;
  FORMAT dob DATE9.
  day WEEKDATE9. ;
RUN ;
PROC PRINT DATA = work.class2 ;
RUN ;

```

SAS Code

- ▶ Formats applied in PROCs are temporary
- ▶ Only applies for the duration of the procedure

- ▶ Formats applied in DATA are permanent
- ▶ Such formats will be applied to all procedures

STAT 330: Lecture 6

15 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

SAS labels (variable *name* display)

SAS Code

```

PROC PRINT DATA = work.class LABEL ;
  LABEL dob = "Date of Birth"
  gpa = "Grade Point Average" ;
RUN ;

```

SAS Code

- ▶ Labels applied in PROCs are temporary
- ▶ Only applies for the duration of the procedure

SAS Code

```

DATA work.class2 ;
  SET class ;
  LABEL dob = "Date of Birth"
  gpa = "Grade Point Average" ;
RUN ;
PROC PRINT DATA = work.class2 LABEL ;
RUN ;

```

SAS Code

- ▶ Labels applied in DATA are permanent
- ▶ Such formats will be applied to all procedures

STAT 330: Lecture 6

16 / 24

Overview

0000

Informats

00000

Formats and Labels

00000●

PROC FORMAT

0000000

Before and after

SAS Code

```
PROC PRINT DATA = work.class ;
RUN ;
```

Obs	name	GPA	dob	day
1	Bill	3.4	13069	6
2	Susan	2.7	12228	5

SAS Code

```
PROC PRINT DATA = work.class LABEL ;
FORMAT dob DATE9.
      day WEEKDATE9. ;
LABEL dob = "Date of Birth"
      gpa = "Grade Point Average" ;
RUN ;
```

Obs	name	Grade Point Average	Date of Birth	day
1	Bill	3.4	13OCT1995	Thursday
2	Susan	2.7	24JUN1993	Wednesday

STAT 330: Lecture 6

17 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

●000000

Overview

Informats

Formats and Labels

PROC FORMAT

STAT 330: Lecture 6

18 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

●●00000

PROC FORMAT

DATE9., COMMA8., etc., are examples of formats built in to SAS. You can create your own custom format for either character or numeric variables with PROC FORMAT.

SAS Code

```
PROC FORMAT ;
VALUE nameA range1 = "formatted value 1"
      range2 = "formatted value 2" ;

VALUE $nameB "range1" = "formatted value 1"
      "range2" = "formatted value 2" ;
RUN ;

PROC PRINT DATA = example ;
FORMAT var1 nameA. var2 $nameB. var3 nameA. ;
RUN;
```

► *nameA* formats a numeric value

► *\$nameB* formats a character value

STAT 330: Lecture 6

19 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

00●0000

PROC FORMAT - range key words

Keyword	Description
hyphen (–)	continuous range
LOW/HIGH	used in ranges to indicate lowest/highest non-missing value
less than (<)	used in ranges to exclude end point
OTHER	assigns format to any values not yet listed

STAT 330: Lecture 6

20 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

PROC FORMAT - ranges

Value in data set	Value display	Explanation
"A" =	"Asia"	A is a character value, goes in quotes
1,3,5 =	"Odd"	looking for numeric values 1 3 or 5
500-high =	"Upper end"	numeric values from 500 to infinity
3-<13 =	"Child"	numeric values between 3 and 13, excluding 13 exactly
OTHER =	"anything else"	any other value

STAT 330: Lecture 6

21 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

Babies.csv

bwt	baby's weight at birth in ounces
parity	0=first born, 1=otherwise
smoke	smoking status of mother: 0=not now, 1=yes now

SAS Code

```

PROC FORMAT ;
  VALUE birthorder 0 = "first born"
                  1 = "otherwise" ;

  VALUE smokestatus 0 = "not now"
                   1 = "yes now" ;

  VALUE birthweight low-88 = "under"
                   88<-high = "normal" ;
RUN ;

```

SAS Code

STAT 330: Lecture 6

22 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

Two categorical variables, with and without formats

SAS Code

```

PROC FREQ DATA = work.babies ;
  TABLES parity*smoke /
  NOROW NOCOL NOPERCENT ;
RUN;

```

SAS Code

parity	smoke		Total
	0	1	
0	548	363	911
1	194	121	315
Total	742	484	1226

Frequency Missing = 10

SAS Code

```

PROC FREQ DATA = work.babies ;
  TABLES parity*smoke /
  NOROW NOCOL NOPERCENT ;
  FORMAT parity birthorder.
         smoke smokestatus. ;
RUN;

```

SAS Code

parity	smoke		Total
	not now	yes now	
first born	548	363	911
otherwise	194	121	315
Total	742	484	1226

Frequency Missing = 10

STAT 330: Lecture 6

23 / 24

Overview

0000

Informats

00000

Formats and Labels

000000

PROC FORMAT

0000000

One quantitative variable, with and without format

SAS Code

```

PROC FREQ DATA = work.babies ;
  TABLES bwt ;
RUN;

```

SAS Code

bwt	Frequency	Percent	Cumulative Frequency	Cumulative Percent
55	1	0.08	1	0.08
58	1	0.08	2	0.16
62	1	0.08	3	0.24
63	1	0.08	4	0.32
65	2	0.16	6	0.49
68	1	0.08	7	0.57
69	1	0.08	8	0.65
71	5	0.40	13	1.05
72	2	0.16	15	1.21
73	1	0.08	16	1.29
75	5	0.40	21	1.70
77	2	0.16	23	1.86
78	3	0.24	26	2.10
79	1	0.08	27	2.18
80	2	0.16	29	2.35
81	3	0.24	32	2.59

SAS Code

```

PROC FREQ DATA = work.babies ;
  TABLES bwt ;
  FORMAT bwt birthweight. ;
RUN;

```

SAS Code

bwt	Frequency	Percent	Cumulative Frequency	Cumulative Percent
under	63	5.10	63	5.10
normal	1173	94.90	1236	100.00

STAT 330: Lecture 6

24 / 24