

Hye-Chung Kum
Population Informatics Research Group
http://research.tamhsc.edu/pinformatics/
http://pinformatics.web.unc.edu/

License:

Data Science in the Health Domain by Hye-Chung Kum is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License

Course URL:

http://pinformatics.tamhsc.edu/phpm672



Population Informatics Article

• Thoughts?



Lab 2 & Assignment 2: Objective

- To write conditional logic codes
- Subset columns (variables) from a table
- Subset rows (observations) from a table
- Recode, rename variables and calculate new variables
- Label variables and values



Recommended Reading

- Carefully read each of the modules below.
 Each has very good explanations of exactly how to do certain things.
 - http://www.ats.ucla.edu/stat/sas/modules/vars.htm
 - http://www.ats.ucla.edu/stat/sas/modules/subset.htm
 - http://www.ats.ucla.edu/stat/sas/modules/missing.htm
 - http://www.ats.ucla.edu/stat/sas/modules/labels.htm
- Little SAS book
 - Sections in Chapter 3



Subset columns (variables)

- SAS
 - Three places possible
 - Reading in, writing out, during datastep
 - keep, drop



Subset rows (observations)

SAS

```
• where cond;
```

• if cond;



Calculate new variable (assignment)

- SAS (in data step)
 - var1 = 1; * assignment;



Rename existing variable

- SAS (in data step)
 - Depending on where you do this, different behavior
 - rename oldvar=newvar



Swap x1 & x2

Write the code in SAS



Label variables

SAS

o label var1 = "LABLE";



Label values

SAS: define format, then use in data step

```
proc format;
value fname
    val1= "LAB1"
    val2= "LAB2";
* inside data step;
format var1 fname.
```



Label values

SAS: define format, then use in data step

```
proc format;
value fname
  val1= "LAB1"
  val2= "LAB2";
* inside data step;
format var1 fname.
```





Name	Туре	Size	V alue
bcigever	int8	I byte	I or 0

```
label bcigever= "Ever smoked";
```

- Labeling variable
 - Give a more human friendly name to the variable name.
 - Same as bcigever (the computer friendly name for the variable used in the programs)
 - Stored in the header information for the table



Label Var vs Value

```
NameTypeSizeValuebcigeverint8I byteI or 0
```

```
proc format;
   value bool
            "TRUE"
            "FALSE" :
* inside data step;
data outfile;
set infile;
format bcigever bool.;
* Removing a format;
data outfile;
set infile;
format bcigever;
```

- labeling value
 - Give a more human friendly name to the variable value.
 - Same as 1(=TRUE) or 0(=FALSE)
 - internally, the computer stores 0 or 1
 - But, when printing the values for humans, the computer uses the format you created and designated to use for this variable.
 - Can be used on multiple variables
 - It can be permanent (if done in the data step) or temporary (if done in proc steps)
 - The format must be created BEFORE use
 - Stored in the header information for the table



Variable type (for analysis)

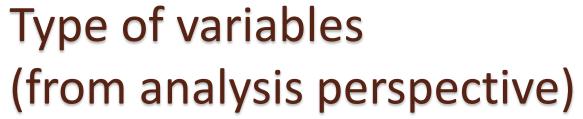
- Numerical
 - includes binary & numerical group coding
- Categorical
 - Numerical code groups
 - String code groups
- ID variables
 - Only used to identify obs, and not used for analysis



Basic descriptive analysis

- Numerical
 - N, mean, max, min, std dev, unique values (mode)
 - SAS: proc means
- Categorical
 - Frequencies, cross tabulation
 - SAS: proc freq;
 - tables var1list/nocol norow nopercent;
 - tables var1*var2/nocol norow nopercent;





- Var Types
 - Continuous (discrete is continuous in computers)
 - Categorical
 - Boolean
 - ID: no other information but to link tables together. i.e. random patient ID used in two tables.
- Helps you starting thinking about what you can do with the information
- Not all variables types exist in datasets.
- Just state NA.



Reminder

- Make sure to understand lab 2
 - You MUST submit programs, logs, and output along with assignment 2
 - This is how you will LEARN
 - Most IMPORTANT part of class
- Dataset(s) you want to use through out the class
 - Flu dataset
 - Texas Inpatient Public Use Data File (PUDF)
 - http://www.dshs.state.tx.us/thcic/hospitals/Inpatientpu df.shtm



Assignment 1

• How was it?



To write conditional logic codes

SAS

```
o if cond then [do;] ---prog---; [end;]
```

where cond;



Recode existing variables

- SAS (in data step)
 - No difference between existing/new
 - Use if/then/else to conditionally recode
 - var1 = 3; * assignment new value;

```
* One way;
if race= 'Asian' then race= 'Other';
else if race= 'Native' then race= 'Other';

* Another way;
if race in ( 'Asian', Native') then race= 'Other';
```



Thresholds

- Many used thresholds to recode continuous vars into categorical vars
- Food for thought: how should such thresholds be determined?

