

Lecture 3: Data Cleaning Part 1

BIOSTAT 212



Amanda Irish
Amanda.Irish@ucsf.edu
Department of Epidemiology and Biostatistics
University of California San Francisco

A note on `clear`

- ◊ Use `clear` at the beginning of a do-file to clear Stata's working memory (previously loaded dataset & any value labels)
- ◊ Need to use `clear` when loading a new dataset IF you have made changes to your current dataset
 - If you have made changes to the current dataset and try to load a new one without using `clear`, Stata will give you an error message
 - If you have not made changes to the current dataset, Stata will load the new one without the need to use `clear`

Midterm Data Cleaning Project

- ◇ Due August 25th (no assignment due next week)
 - Worth the same # pts as the final
- ◇ Three parts:
 - Baseline data cleaning
 - Follow-up data cleaning
 - Merging and dealing with longitudinal data
- ◇ Will cover all the commands you need this week and next week.
- ◇ You may work together, but turn in your own assignments



Today

Data Cleaning Part 1:

- Modifying existing variables
- Labels
- Creating new variables
- Keeping and dropping observations
- Detecting and dealing with outliers
- String variables

Data Cleaning 101



- Your data **will** need cleaning (no matter how well-run the study is)
- Most of your time will be spent cleaning and processing data
- Keep a raw unedited version of the original dataset and do not ever touch it. Save revised data in a new file.

`/*Document everything and trust no one...especially yourself */`

Data Cleaning 101 – document, document, document!



- Divide your do-file into sections for each task and explain what each section is doing
- All your data cleaning decisions should be fully documented in your do-file
- You will not remember what you did in 6 months' time (or even 1 month!)
- Separate do-files for cleaning and analysis

`/*Document everything and trust no one...especially yourself */`

Load the dataset

◇ Download the “acupuncture_raw.csv” dataset from the course website

◇ Import the data set into Stata

```
import delimited acupuncture_raw.csv, clear
```

-or-

File > Import > Text data (*.csv)

Modifying Existing Variables

Renaming Variables

```
rename old_varname new_varname
```

- Changes the name of an existing variable
- Content of the variable is unchanged
- Any labels are unchanged

Recoding Variables

◇ recode *varname* (rule)

<i>rule</i>	Example	Meaning
# = #	3 = 1	3 recoded to 1
# # = #	2 . = 9	2 and . recoded to 9
#/# = #	1/5 = 4	1 through 5 recoded to 4
<u>nonmissing</u> = #	nonmiss = 8	all other nonmissing to 8
<u>missing</u> = #	miss = 9	all other missings to 9

Example: recode completed (-99=.)

Labels

3 levels of labeling in Stata

- ◇ Dataset label
- ◇ Variable labels
- ◇ Value labels

Dataset label

label data "label name"

```
. label data "this dataset contains raw acupuncture trial data"
```

```
. describe
```

Contains data

obs: 401

vars: 78

this dataset contains raw
acupuncture trial data

variable name	storage type	display format	value label	variable label
id	int	%8.0g		

*label name can have spaces, but must be in quotes

Variable labels: help you remember details

Data Editor (Browse)

id[1] 139

	chronicity	acupunctur-t	group	basehs	posths	oneyrhs	f1
21	45	2	0	59	18.75	43.75	28
22	34	2	1	22.25	4.5	6.25	20
23	6	2	1	17	8.75	17.75	15
24	20	3	1	17.75	11	14.25	21
25	16	3	0	18.5	16.75	9.75	20
26	36	9	0	20	30.25	19.75	10
27	15	7	1	17.75	17.25	16.75	16
28	19	9	0	63.75	59	71.5	28
29	24	9	0	36.25	41.75	46.75	26
30	35	9	1	77	.	.	28
31	45	9	1	11	14.75	9.5	6
32	15	3	0	9.75	.	.	14
33	2	9	1	28.5	14	17.25	28
34	37	7	0	10.25	.	.	10
35	49	7	1	20.75	11	12.5	17
36	2	3	1	18.75	20.25	11	25
37	10	3	0	11.5	9	7.5	10
38	10	6	0	11.25	18.75	.	10
39	10	6	1	34	.	.	23
40	41	6	0	21.25	.	.	12
41	20	8	0	23.25	18	16	18
42	23	8	1	11.75	4.75	3.75	11
43	4	8	0	22.5	.	.	26

Vars: 78 Order: Dataset Obs: 401 Filter: Off

Variables

Name	Label
age	
sex	sex
migraine	
chronicity	
acupuncturist	
group	
basehs	
posths	
oneyrhs	

Properties

▼ Variables

Name	id
Label	
Type	int
Format	%8.0g
Value label	
Notes	

▼ Data

► Filename	
Label	
► Notes	
Variables	78
Observations	401
Size	79.89K
Memory	64M

Create variable label

Variables Manager (78 variables)

Filter:

Name	Label	Type	Format	Value label	Notes
id		int	%8.0g		
age		int	%8.0g		
sex	sex	str6	%9s		
migraine		byte	%8.0g		
chronicity		byte	%8.0g		
acupuncturist		byte	%8.0g		
group		byte	%8.0g		
basehs		float	%9.0g		
posths		float	%9.0g		
oneyrhs		float	%9.0g		
f1		float	%9.0g		
f2		float	%9.0g		
f5		float	%9.0g		
pf1		float	%9.0g		
rlp1		byte	%8.0g		
rle1		float	%9.0g		
ef1		float	%9.0g		

Variable properties

Name:

Label:

Type:

Format:

Value label:

Notes:

Data > Variables manager

Create variable label

```
label variable varname "label name"
```

*label name can have spaces, but must be in quotes

Practice

Variable f1 = headache frequency at baseline

1. Rename f1 → hfreq_bl
2. Label hfreq_bl = headache frequency at baseline

Value labels

Data Editor (Browse)

id[1] 139

	chronicity	acupuncturist	group	basehs	posths	oneyrhs	f1
21	45	2	0	59	18.75	43.75	28
22	34	2	1	22.25	4.5	6.25	20
23	6	2	1	17	8.75	17.75	15
24	20	3	1	17.75	11	14.25	21
25	16	3	0	18.5	16.75	9.75	20
26	36	9	0	20	30.25	19.75	10
27	15	7	1	17.75	17.25	16.75	16
28	19	9	0	63.75	59	71.5	28
29	24	9	0	36.25	41.75	46.75	26
30	35	9	1	77	.	.	28
31	45	9	1	11	14.75	9.5	6
32	15	3	0	9.75	.	.	14
33	2	9	1	28.5	14	17.25	28
34	37	7	0	10.25	.	.	10
35	49	7	1	20.75	11	12.5	17
36	2	3	1	18.75	20.25	11	25
37	10	3	0	11.5	9	7.5	10
38	10	6	0	11.25	18.75	.	10
39	10	6	1	34	.	.	23
40	41	6	0	21.25	.	.	12
41	20	8	0	23.25	18	16	18
42	23	8	1	11.75	4.75	3.75	11
43	4	8	0	22.5	.	.	26

Vars: 78 Order: Dataset Obs: 401 Filter: Off

Variables

Name	Label
☒ age	
☒ sex	sex
☒ migraine	
☒ chronicity	
☒ acupuncturist	
☒ group	
☒ basehs	
☒ posths	
☒ oneyrhs	

Properties

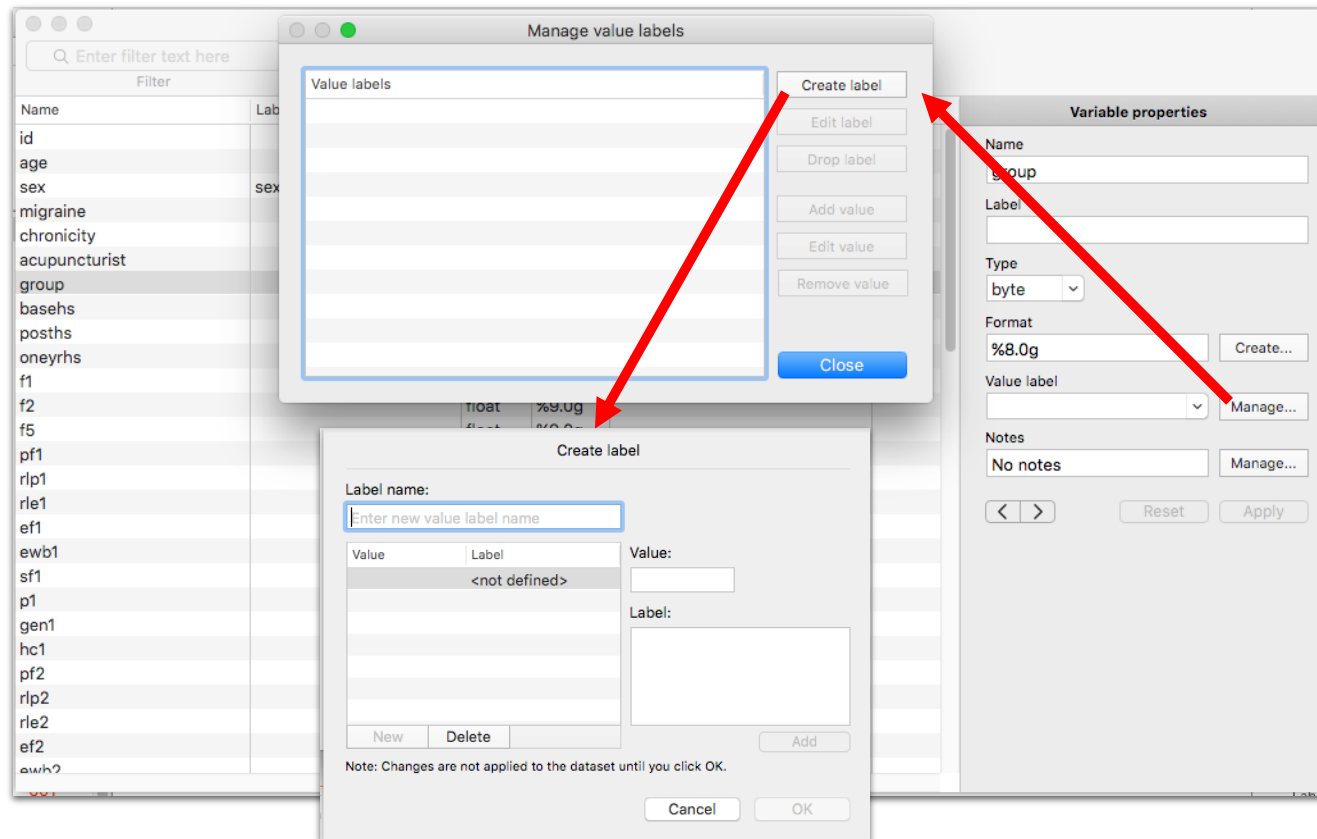
▼ Variables

Name	id
Label	
Type	int
Format	%8.0g
Value label	
Notes	

▼ Data

► Filename	
Label	
► Notes	
Variables	78
Observations	401
Size	79.89K
Memory	64M

Add value labels



Data > Variables manager

Add value labels for categorical variables

2-step process:

```
label define labelname # "label" # "label" ...  
label values varname labelname
```

Example:

```
label define yesno 0 "no" 1 "yes"  
label values completed yesno
```

Practice

- ◊ Label group (1=intervention) (0=control)

Displaying numeric values for labels

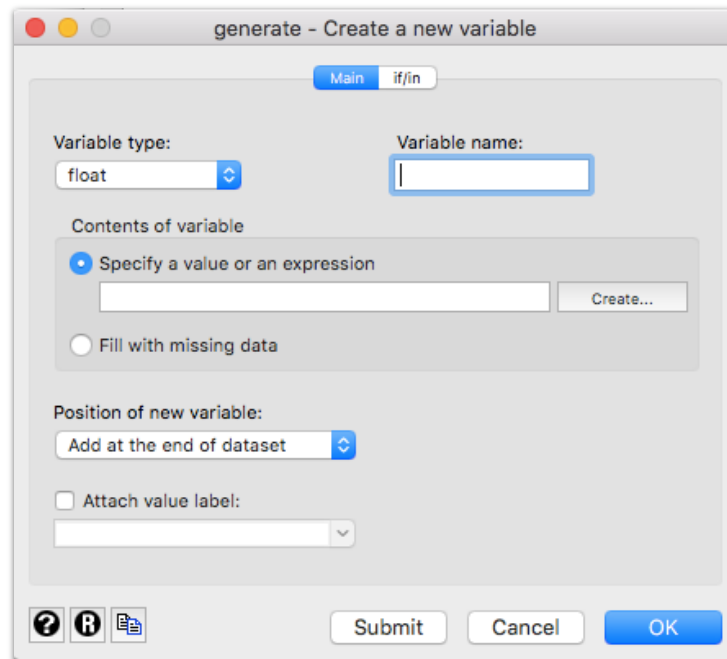
```
numlabel [lblname], add
```

. tab sex			
1=male, 2=female	Freq.	Percent	Cum.
1. Male	4,915	47.48	47.48
2. Female	5,436	52.52	100.00
Total	10,351	100.00	

labelbook – gets you a list of all value labels

Generating New Variables

Create a new variable



Data > Create or change data > Create a new variable

Create a new variable

generate `new_varname = exp`

generate a new variable

`exp` = #, ".", _n, addition(+), subtraction(-)

example: `gen calsum = cal1+cal2`

generate `new_varname=(condition)`

generate a new dichotomous variable
where 1= condition_True &
0=condition_False

Options

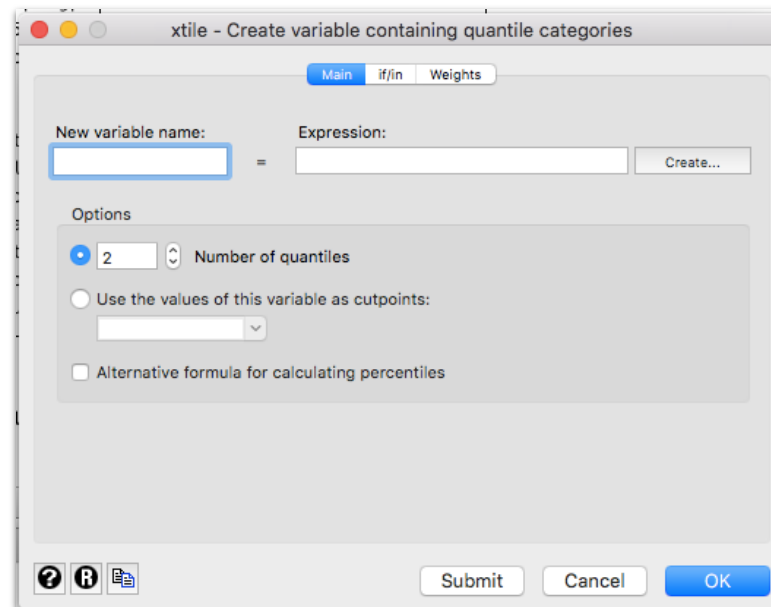
`before (varname) /after (varname)`

place new variable before/after
specified variable

Practice

- ◇ Generate a new running id variable and a blank variable
- ◇ Generate a new variable for 'severe migraine' for a migraine + a headache score >50

Create variable containing quantile categories



Statistics > Summaries, tables, and tests > Summary and descriptive statistics > Create variable of quantiles

Create variable containing quantile categories

`xtile new_varname = varname` create a new variable containing percentiles

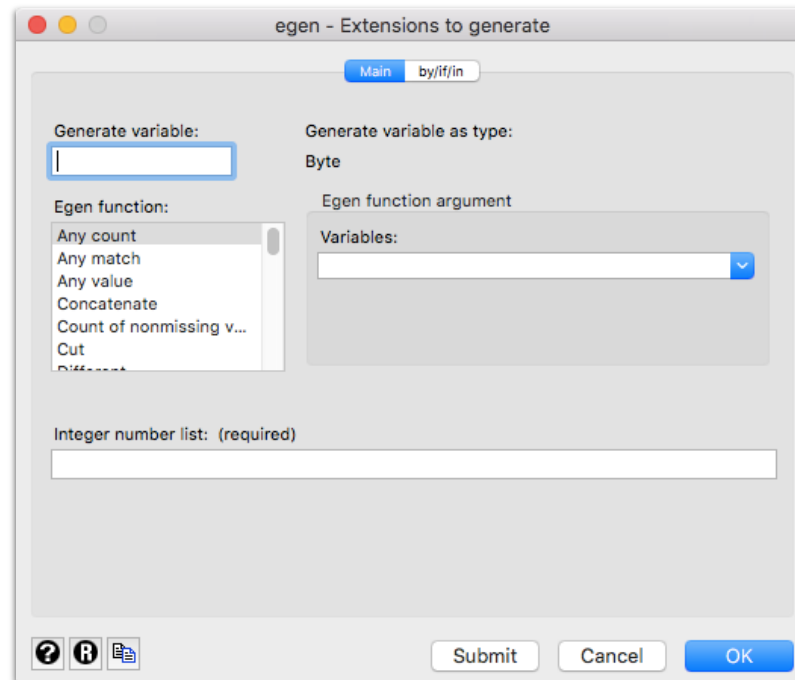
Options

`if` –use logic operators
`, nq(#)` –set number of percentiles (default is 2)

Practice

- ◇ Create a new categorical variable that is a quartile of baseline headache score.
- ◇ What is the mean baseline headache score in the first quartile?

Create new variable (extended)



Data > Create or change data > Create new variable (extended)

Create new variable (extended)

```
egen new_varname = function(varname)
```

-create a new variable that is a function of a different variable(s)

Functions

mean	egen new_varname = mean(varname)
row mean	egen new_varname = rowmean(varlist)
row total	egen new_varname = rowtotal(varlist)
concatenate	egen new_varname = concat(varlist)

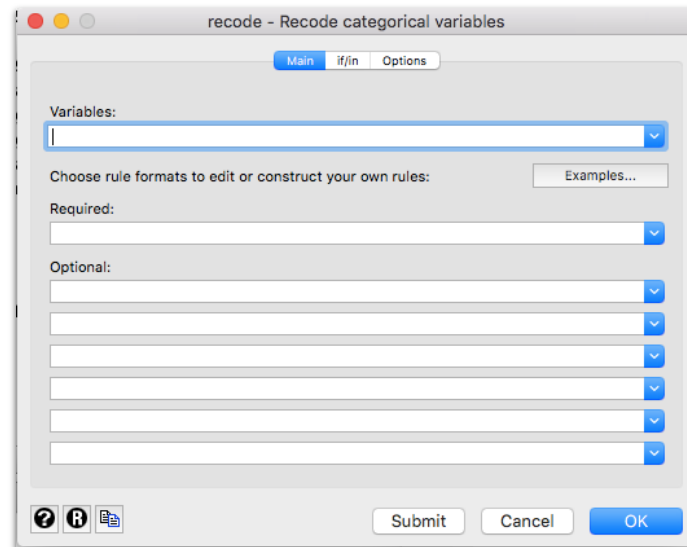
Options: by, if

*many more functions → look in help file

Practice

- ◇ Create a variable for average headache score (basehs, posths, oneyrhs)

Create categorical variable from continuous variable



Data > Create or change data > Other variable-transformation commands > Recode categorical variable

Create categorical variable from continuous variable

```
recode old_varname (min/# = 0 "value label") (#/# = 1 "value  
label") (#/max = 2 "value label"), gen(new_varname)
```

Practice

- ◇ Create a dichotomous variable for age (>50)
- ◇ How many study participants are 50 years old or older?

Keeping and Dropping

Subset data: drop

`drop if` drops subset of data given condition
example `drop if age>100`

`drop in` drops subset of observations
example `drop in 1/4`

`drop varlist` drop one or more variables

****careful! you can't undo**

Subset data: keep

keep if	keep observations meeting condition, all others are dropped! example keep if age<100
keep in	keep subset of observations example keep in 1/30
keep varlist	keep one or more variables, all others dropped!

****careful! you can't undo**

Detecting and Dealing with Outliers

Range checks

◇ Are there any outliers? (unrealistic values?)

- `codebook varname`
- `inspect varname`
- `histogram varname`
- `summarize varname`
- `spikeplot varname`

Removing outliers: replace

```
replace varname = # if condition
```

replace the value of a variable meeting a certain condition

Example

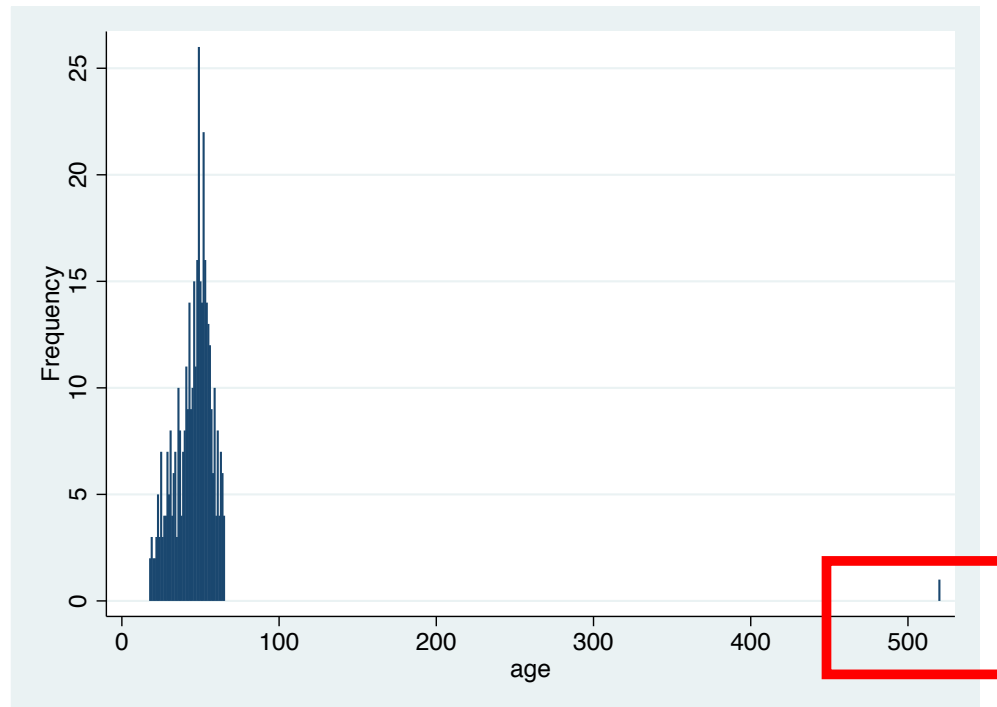
```
replace weight = 230 if id == "13des4"  
replace age = . if age>110
```

Practice

- Are there any outliers for age?
- Replace outlier

Range checks

`spikeplot age`



String Variables

What is a string variable?

- ◇ Short for “string of characters”
- ◇ Example: data coded as “yes” and “no” instead of 1/0
- ◇ Why we care: Stata can’t perform analytic operations on string variables
 - Commands like `summarize`, `regress` and any other analytical commands won’t work
 - `tabulate` will work but the strings will be in alphabetical order – may not be what you want

Someone has a string variable containing

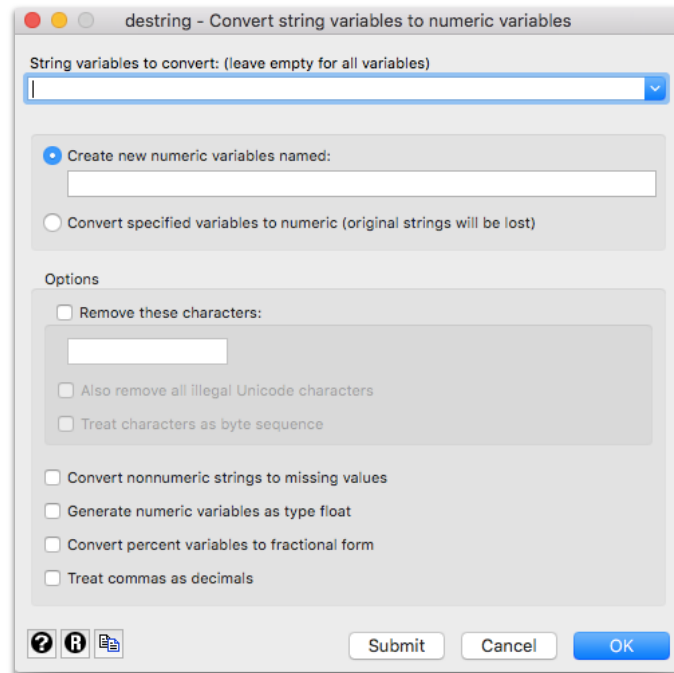
MC = most competitive
HC = highly competitive
VC = Very competitive
C = Competitive
LC = Less Competitive
NC = Non-Competitive

If you **tabulate** the string variable, the results are presented in alphabetical order:

```
. tabulate comp
```

comp	Freq.	Percent	Cum.
C	41	20.50	20.50
HC	35	17.50	38.00
LC	27	13.50	51.50
MC	34	17.00	68.50
NC	34	17.00	85.50
VC	29	14.50	100.00
Total	200	100.00	

Convert string variables to numeric



Data > Create or change data > Other variable-transformation commands > Convert variables from string to numeric

Convert string variables to numeric

destring `varname`, generate(`new_varname`)
convert string into numeric variable

encode `varname`, generate(`new_varname`)
encode string into numeric variable with labels

destring or encode?

encode and `destring` both convert string variables to numeric variables. What's the difference?

◇ Use `destring` to:

- convert intrinsically numeric variables such as **age** that do not have labels but happen to be stored as strings.

◇ Use `encode` to:

- convert categorical variables that will be included in statistical models but have been stored as strings. **encode** can convert the categories to value labels for the new numeric variable.

Practice

- ◇ Change withdrawal_reason from string to numeric/categorical:
- ◇ What percent of individuals who dropped out withdrew consent?

Next Week

Data Cleaning Part 2:

- Working with dates
- Missing data
- Duplicates
- Merging/Appending
- Reshape

Lecture 3: Data Cleaning Part 1

BIOSTAT 212



Amanda Irish

Amanda.Irish@ucsf.edu

Department of Epidemiology and Biostatistics

University of California San Francisco