

**Sociology 706: Longitudinal Data Analysis**  
**Instructor: Natasha Sarkisian**

**Introduction to Longitudinal Data Analysis**

**Types of data:**

- Cross-sectional data = data collected at one point in a time
- Longitudinal data = data collected on the same variables from the same units are measured at 2 or more time periods
- Quasi-longitudinal data:
  - Repeated cross-sections: same variables are measured at 2 or more periods, but on different units
  - Time-ordered cross-sections: data are collected from the same units at two or more periods, but each variable is measured only once
  - Retrospective studies: data about different time points in the past are collected all at once

**Advantages of longitudinal data:**

- Can examine patterns of change over time
- Can study individual development
- Can analyze duration
- Can help locate the causes of social phenomena

Basic conditions for establishing causality:

1. Correlation (X and Y must change together)
2. Theory (logical explanation)
3. Non-spuriousness (other explanations must be ruled out)
4. Temporal order (X must precede Y temporally)

Longitudinal research can ensure temporal order, but by itself does not assure causality!

**Disadvantages of longitudinal data:**

- Costly and time-consuming
- Panel attrition problems: Refusals, changes of residence, death
- Discrete time measurement: Exactly what happens between the time points is unknown
- Time lag problems: Time intervals might not match the lag between cause and its consequence
- Panel conditioning: Responses in one wave can be influenced by those given in the previous waves, respondents themselves can change as a result of participating in the study

**Longitudinal analysis:**

Observations on the same unit over time → not independent → have to apply special techniques

Two types of questions:

- Descriptive: What kind of change takes place?
- Explanatory: What predicts this change (its nature and timing)?

## Analytical Models and Data Structures

<i>FEW CASES</i> ( <i>n &lt; 20</i> )	<i>MANY CASES</i> ( <i>n &gt; 100</i> )
<i>MANY PERIODS</i> ( <i>t &gt; 20</i> )	<i>MANY PERIODS</i> ( <i>t &gt; 10</i> )
ARIMA models: covariates, transfer function models, interrupted time series models	Continuous time event history analysis: Cox proportional hazards and parametric hazard models
Autoregressive (AR) time series models	
Lagged endogenous variable (LEV) models	Multilevel growth curve models
Multivariate dynamic analysis of categorical data with optimal scaling	
<i>FEW CASES</i> ( <i>n &lt; 20</i> )	<i>MANY CASES</i> ( <i>n &gt; 100</i> )
<i>FEW PERIODS</i> ( <i>t &lt; 10</i> )	<i>FEW PERIODS</i> ( <i>t &lt; 10</i> )
Pooled cross-sectional/time-series analysis	Linear panel analysis conditional change model (lagged endogenous variable)
	Linear panel analysis unconditional change model (change score)
	Latent growth curve analysis
	Discrete time event history analysis
	Multilevel growth curve models

From: "Longitudinal Research" by Scott W. Menard

### Introduction to Stata

#### Basic syntax of Stata commands:

1. Command – What do you want to do?
2. Names of variables, files, etc. – Which variables or files do you want to use?
3. Qualifier on observations -- Which observations do you want to use?
4. Options – Do you have any other preferences regarding this command?

#### Obtain help:

help *command*  
 search *keyword* or lookup *keyword*  
 net search *keyword*

#### Set preferences:

set memory #m [, perm] -- to increase the amount of memory for the data  
 set varlabelpos # – to change the number of characters allowed for variable names

#### Open and close files:

Data files:  
 use *filename.dta*, clear – opens data file  
 save *filename.dta*, replace

### Log files:

log using *filename.log* [, append replace] – open log file  
log close -- close log file (saves automatically)  
translate – convert log file types (.log and .smcl) and recover results  
cmdlog using *filename* – open command only log file

### Do-files:

doedit *filename.do* – to create or edit a do-file  
do *filename.do* – to execute a do-file

### **Add comments:**

\* comment

### **Examine the data:**

browse – explore the data  
describe – get information on variables and labels  
list *varnames* [in *exp*] – list the values of specified variables for specified observations  
codebook *varnames* – summarize variables in codebook format  
sum *varnames* [, detail] – get summary statistics  
tab *varname*, [nolabel missing] – get frequency distribution (options: without value labels, display the missing data)  
tab *varname varname* [, row col cell chi2] – generate a two-way table (Options: get percentages for rows, columns, cells; obtain chi-square test of independence)  
tab1 *varnames* – generate separate frequency distribution for each variable

### **Basic graphical examination of the data:**

dotplot *varname* – obtain a univariate frequency distribution graph  
graph box *varname* – obtain a univariate boxplot  
scatter *varname varname* – obtain a scatterplot for two variables  
graph matrix *varnames* – obtains all possible scatterplots for a set of variables  
graph save *filename* [,replace] – saves a graph into a .gph file  
graph use *filename* – displays a previously saved graph

### **Manage the data:**

Edit – edit the data  
drop [in *range*] [if *exp*] – drop observations  
keep [in *range*] [if *exp*] – keep observations  
drop *varnames* – drop variables  
keep *varnames* – keep variables

### **Recode variables:**

generate *newvarname* = *exp* [in *exp*] [if *exp*] – make a new variable  
replace *varname* = *exp* [in *exp*] [if *exp*] – replace values of existing variable  
recode *varname* (*rule*) (*rule*) ... , generate(*newvarname*) – make a new variable  
label variable *varname* “*label*” – create variable label  
Create value labels:  
label define *labelname label value label value*... -- defines a set of value labels  
label values *varname labelname* – applies a set of value labels to a variable

**Good resource for learning Stata:** <http://www.ats.ucla.edu/stat/stata/>



```

occ80          int      %8.0g      occ80          rs census occupation code (1980)
prestg80       byte     %8.0g      prestg80       rs occupational prestige score
                                                (1980)
indus80       int      %8.0g      indus80       rs industry code (1980)
marital       byte     %8.0g      marital       marital status
divorce       byte     %8.0g      divorce       ever been divorced or separated
widowed       byte     %8.0g      widowed       ever been widowed

```

```
--Break--
```

```
r(1);
```

```
*used Break button to stop Stata from producing more output
```

```
*using data browser to look at the data
```

```
. browse
```

```
*using data editor to change data
```

```
. edit
```

```
- preserve
```

```
- replace hrs2 = 1 in 7
```

```
- restore
```

```
*preserve in the beginning saved a copy of the dataset in Stata memory; restore
in the end returned to that saved version, so we didn't make any changes
```

```
*Get summary statistics
```

```
. sum hrs1 hrs2
```

Variable	Obs	Mean	Std. Dev.	Min	Max
hrs1	1729	41.77675	14.62304	1	89
hrs2	50	34.88	15.55719	1	60

```
. sum hrs1 hrs2, detail
```

```
number of hours worked last week
```

Percentiles		Smallest		
1%	6	1		
5%	16	2		
10%	21	2	Obs	1729
25%	36	2	Sum of Wgt.	1729
50%		40	Mean	41.77675
			Std. Dev.	14.62304
75%	50	89		
90%	60	89	Variance	213.8332
95%	68	89	Skewness	.2834814
99%	88	89	Kurtosis	4.310339

```
number of hours usually work a week
```

Percentiles		Smallest		
1%	1	1		
5%	6	3		
10%	9	6	Obs	50
25%	24	7	Sum of Wgt.	50
50%		40	Mean	34.88
			Std. Dev.	15.55719
75%	43	57		
90%	53	60	Variance	242.0261
95%	60	60	Skewness	-.5207683
99%	60	60	Kurtosis	2.545694

```
*List values of selected variables for each observation
```

```
. list wrkstat hrs1 wrkslf
```

	wrkstat	hrs1	wrkslf
1.	working	40	someone
2.	working	72	someone
3.	working	40	someone
4.	working	60	someone
5.	working	40	someone
6.	working	42	someone
7.	retired	.	someone
8.	keeping	.	someone

--Break--

r(1);

\*same but for observations 100-200

```
. list wrkstat hrs1 wrkslf in 100/200
```

	wrkstat	hrs1	wrkslf
100.	working	40	someone
101.	school	.	someone
102.	working	40	someone
103.	working	51	someone
104.	working	40	someone
105.	unempl,	.	someone
106.	school	.	someone
107.	retired	.	someone

--Break--

r(1);

\*Get codebook info

```
. codebook wrkstat
```

```
wrkstat
labor frce status
```

```
-----
                type: numeric (byte)
                label: wrkstat

                range: [1,8]                units: 1
unique values: 8                            missing .: 0/2765
```

```
tabulation:  Freq.  Numeric  Label
              1432      1  working fulltime
              312       2  working parttime
               52       3  temp not working
              121       4  unempl, laid off
              414       5  retired
               78       6  school
              268       7  keeping house
               88       8  other
```

\*Frequency tables -- tabulate command

```
. tab wrkstat
      labor frce |
      status    |      Freq.      Percent      Cum.
-----+-----
```

working fulltime	1,432	51.79	51.79
working parttime	312	11.28	63.07
temp not working	52	1.88	64.95
unempl, laid off	121	4.38	69.33
retired	414	14.97	84.30
school	78	2.82	87.12
keeping house	268	9.69	96.82
other	88	3.18	100.00

-----+-----  
 Total | 2,765 100.00

\*Including missing values

. tab wrkslf, miss

r self-emp or works for somebody	Freq.	Percent	Cum.
self-employed	307	11.10	11.10
someone else	2,362	85.42	96.53
.	96	3.47	100.00

-----+-----  
 Total | 2,765 100.00

\*Note that missing values are in fact stored as very large numbers -- should be careful when doing data management

\*To suppress labels:

. tab wrkslf, miss nolabel

r self-emp or works for somebody	Freq.	Percent	Cum.
1	307	11.10	11.10
2	2,362	85.42	96.53
.	96	3.47	100.00

-----+-----  
 Total | 2,765 100.00

\*Cross-tabulation

. tab wrkslf wrkgovt

r self-emp or works for somebody	govt or private employee		Total
	governmen	private	
self-employed	13	271	284
someone else	441	1,914	2,355
Total	454	2,185	2,639

\*With row percentages

. tab wrkslf wrkgovt, row

+-----+			
Key			
frequency			
row percentage			
+-----+			
r self-emp or works for somebody	govt or private employee		Total
	governmen	private	

self-employed	13	271	284
	4.58	95.42	100.00
someone else	441	1,914	2,355
	18.73	81.27	100.00
Total	454	2,185	2,639
	17.20	82.80	100.00

\*with all three types of percentages

. tab wrkslf wrkgovt, row col cell

Key			
frequency			
row percentage			
column percentage			
cell percentage			
r self-emp or works for somebody	govt or private employee		Total
	governmen	private	
self-employed	13	271	284
	4.58	95.42	100.00
	2.86	12.40	10.76
	0.49	10.27	10.76
someone else	441	1,914	2,355
	18.73	81.27	100.00
	97.14	87.60	89.24
	16.71	72.53	89.24
Total	454	2,185	2,639
	17.20	82.80	100.00
	100.00	100.00	100.00
	17.20	82.80	100.00

\*with a chi-square test

. tab wrkslf wrkgovt, row col cell chi2

Key			
frequency			
row percentage			
column percentage			
cell percentage			
r self-emp or works for somebody	govt or private employee		Total
	governmen	private	
self-employed	13	271	284
	4.58	95.42	100.00
	2.86	12.40	10.76
	0.49	10.27	10.76
someone else	441	1,914	2,355
	18.73	81.27	100.00

	97.14	87.60	89.24
	16.71	72.53	89.24
-----			
Total	454	2,185	2,639
	17.20	82.80	100.00
	100.00	100.00	100.00
	17.20	82.80	100.00

Pearson chi2(1) = 35.6181 Pr = 0.000

\*Multiple univariate tables of frequencies are obtained using tab1 command  
. tab1 wrkslf wrkgovt

-> tabulation of wrkslf

r self-emp or works for somebody	Freq.	Percent	Cum.
self-employed	307	11.50	11.50
someone else	2,362	88.50	100.00
-----			
Total	2,669	100.00	

-> tabulation of wrkgovt

govt or private employee	Freq.	Percent	Cum.
government	454	17.19	17.19
private	2,187	82.81	100.00
-----			
Total	2,641	100.00	

\*Using conditions

\*Can use:

< less

> more

== equal

<= less or equal

>= more or equal

~= not equal

Can connect them with & (and) and | (or). Can also use parentheses to combine conditions.

. codebook marital

-----  
marital  
marital status  
-----

```

      type: numeric (byte)
      label: marital

      range: [1,5]
unique values: 5
units: 1
missing .: 0/2765

      tabulation: Freq.  Numeric  Label
                  1269      1      married

```

```

                247          2 widowed
                445          3 divorced
                 96          4 separated
                708          5 never married

```

```
. sum hrs1 if wrkslf==1 & marital==5
```

Variable	Obs	Mean	Std. Dev.	Min	Max
hrs1	35	38.48571	20.74406	8	89

```
. sum hrs1 if wrkslf==1 & marital>1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
hrs1	96	39.48958	20.22609	5	89

```
. sum hrs1 if wrkslf==1 & marital>1 & marital<=5
```

Variable	Obs	Mean	Std. Dev.	Min	Max
hrs1	96	39.48958	20.22609	5	89

```
. sum hrs1 if wrkslf==1 & marital>1 & marital~=.
```

Variable	Obs	Mean	Std. Dev.	Min	Max
hrs1	96	39.48958	20.22609	5	89

```
. sum hrs1 if wrkslf==1 & (marital==1 | marital==2)
```

Variable	Obs	Mean	Std. Dev.	Min	Max
hrs1	137	41.46715	18.42515	3	89

```
*help in Stata -- help, search, lookup commands
```

```
. help tabulate
```

```
. search logistic
```

```
Keyword search
```

```
Keywords: logistic
```

```
Search: (1) Official help files, FAQs, Examples, SJs, and STBs
```

```
Search of official help files, FAQs, Examples, SJs, and STBs
```

```
[U] Chapter 26 . . . . . Overview of Stata estimation commands
(help estcom)
```

```
[R] clogit . . . . . Conditional (fixed-effects) logistic regression
(help clogit)
```

```
[R] cloglog . . . . . Complementary log-log regression
(help cloglog)
```

```
[R] constraint . . . . . Define and list constraints
(help constraint)
```

```
[R] fracpoly . . . . . Fractional polynomial regression
(help fracpoly)
```

```
[R] glogit . . . . . Logit and probit for grouped data
(help glogit)
```

[R]     logistic . . . . . Logistic regression, reporting odds ratios  
(help logistic)

[R]     logistic postestimation . . . . . Postestimation tools for logistic  
(help logistic postestimation)

[R]     logit . . . . . logistic regression, reporting coefficients  
(help logit)

[R]     logit postestimation . . . . . Postestimation tools for logit  
(help logit postestimation)

[R]     mfp . . . . . Multivariable fractional polynomial models  
(help mfp)

[R]     mlogit . . . . . Multinomial (polytomous) logistic regression  
(help mlogit)

[R]     nlogit . . . . . Nested logit regression  
(help nlogit)

[R]     ologit . . . . . Ordered logistic regression  
(help ologit)

--Break--

r(1);

\*Using do-files

\*Open do-file editor, create and save your file (.do)

\*You can execute that file from the do-file editor or using the command line

do mydofile.do

\*But be careful to specify the location of your file.

\*It is often convenient to create and edit do-files in another text editor - I prefer TextPad: <http://www.textpad.com>

\*You can also keep the log of just the commands:

cmdlog using filename

\*Then you can use that log as a do-file

\*And if you want to save all commands you've done so far, just right click on the command window and select "Save Review Contents"

\*Graphics in Stata

. scatter hrs1 prestg80

. graph matrix hrs1 hrs2 prestg80 sphrs1 sppres80

. histogram hrs1  
(bin=32, start=1, width=2.75)

\*We can save graphs for future use:

graph save mygraph.gph

\*To then display that graph, we type:

graph use mygraph.gph

\*Or you can just copy them and paste them into your word processor

\*To further explore the options available for graphics, use:

. help graph

## Basics of Data Management in Stata

\*To sort all variables in the dataset, use order command to specify a certain order and aorder command to sort alphabetically.

```
. order wrkstat marital sibs childs  
. aorder
```

\*To keep only a subselection of variables in the dataset, use drop and keep

```
. drop spwrksta- spind80  
. keep wrkstat marital sibs childs
```

\*Can also use if and in qualifiers with drop and keep commands:

```
. drop if wrkstat==2  
. keep in 1/100
```

\*to return to the original dataset without saving the modified one:

```
. use "M:\gss2002.dta", clear
```

\*Creating new variables

```
. gen hrs40=  
(2765 missing values generated)  
. replace hrs40 = 0 if hrs1<40  
(490 real changes made)  
. replace hrs40 = 1 if hrs1>=40 & hrs1~=  
(1239 real changes made)
```

```
. tab hrs40, missing  
      hrs40 |          Freq.      Percent      Cum.  
-----+-----  
          0 |             490       17.72      17.72  
          1 |            1,239       44.81      62.53  
          . |            1,036       37.47     100.00  
-----+-----  
      Total |            2,765     100.00
```

\*label the variable

```
. label variable hrs40 "R works 40 hours a week or more"  
*label its values: two steps, first define a set of labels  
. label define hrs40label 0 "less than 40" 1 "40 or more"  
*next, apply this set to the new variable
```

```
. label values hrs40 hrs40label  
. tab hrs40, missing  
      R works 40 |  
hours a week |  
or more |          Freq.      Percent      Cum.  
-----+-----  
less than 40 |             490       17.72      17.72  
  40 or more |            1,239       44.81      62.53  
          . |            1,036       37.47     100.00  
-----+-----  
      Total |            2,765     100.00
```

```
. codebook hrs40
```

```
-----  
hrs40                                     R works 40 hours a week or more  
-----  
      type:  numeric (float)  
      label:  hrs40label  
  
      range:  [0,1]                               units:  1
```

unique values: 2 missing .: 1036/2765

```
tabulation: Freq. Numeric Label
             490      0 less than 40
             1239     1 40 or more
             1036     .
```

\*To rename a variable, use the rename command:

```
.rename hrs40 hours40
```

\*generate a dummy variable indicating married respondents

```
. codebook marital
```

```
-----
marital                                                    marital status
-----
```

```
type: numeric (byte)
label: marital
```

```
range: [1,5] units: 1
unique values: 5 missing .: 0/2765
```

```
tabulation: Freq. Numeric Label
             1269     1 married
             247     2 widowed
             445     3 divorced
             96     4 separated
             708     5 never married
```

```
. gen married=(marital==1)
```

```
. tab married
```

married	Freq.	Percent	Cum.
0	1,496	54.10	54.10
1	1,269	45.90	100.00
-----			
Total	2,765	100.00	

```
. replace married=. if marital==.
```

```
(0 real changes made)
```

\*another way to generate such a dummy variable

```
. gen married2=0
```

```
. replace married2=1 if marital==1
```

```
(1269 real changes made)
```

```
. tab married2
```

married2	Freq.	Percent	Cum.
0	1,496	54.10	54.10
1	1,269	45.90	100.00
-----			
Total	2,765	100.00	

\*generate age squared variable

```
. gen age2=age^2
```

```
(14 missing values generated)
```

\*generate square root of age variable

```
. gen age2sqrt=sqrt(age2)
```

```
(14 missing values generated)
```

\*generate log of age variable

```
. gen agelg=log(age)
```

```
(14 missing values generated)
```

```
*generate marital status with 3 categories
. recode marital (1=1) (2=2) (3=2) (4=2) (5=3), gen(married3)
(1249 differences between marital and married3)
```

```
*or, we can do the same but a bit shorter:
. recode marital (1=1) (2/4=2) (5=3), gen(marital3)
(1249 differences between marital and marital3)
```

```
. tab marital3
RECODE of |
marital |
(marital |
status) |          Freq.      Percent      Cum.
-----+-----
      1 |          1,269         45.90         45.90
      2 |           788         28.50         74.39
      3 |           708         25.61        100.00
-----+-----
    Total |          2,765        100.00
```

```
*label the new variable
. label variable marital3 "marital status 3 categories"
*label values of the new variable
. label define marital3label 1"married" 2 "previously married" 3 "never married"
. label values marital3 marital3label
```

```
*check the results
. codebook marital3
```

```
-----
marital3                                marital status 3 categories
-----
              type:  numeric (byte)
              label:  marital3label

              range:  [1,3]
unique values:  3
                                units:  1
                                missing .:  0/2765

tabulation:  Freq.  Numeric  Label
              1269      1  married
              788       2  previously married
              708       3  never married
```

```
*Saving the dataset with newly created variable
. save "M:\ gss2002changed.dta"
file M:\gss2002changed.dta saved
```

\*You should keep a do-file with all your data management steps, and in most cases it's a good idea to have one with your analysis steps as well - that way, if you make a mistake, you can easily rerun things. To have that, we can save all the commands that we did interactively into a do-file, or we can right away write a do-file and then execute it.

\*Note that if you are opening a Stata log file in a Word processor, you should change the font to a fixed width font, such as Courier New (otherwise the output looks misaligned). Courier New 10 point usually works the best.

```
*exiting Stata
. exit, clear
```