Introduction to Social Data Analytics

Week 5: Class 10

Arushi Kaushik

arkaushi@ucsd.edu

Today: Data Wrangling in Stata

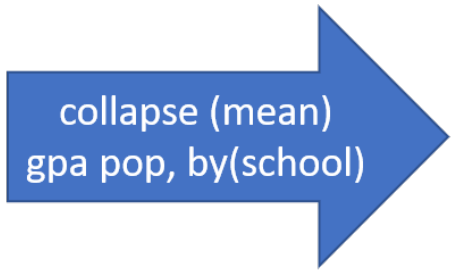
By the end of today's lecture, you should be able to:

- Some Applications of identifiers that we generated in previous class to differentiate between observations within a group
- Collapse a dataset to a coarser unit of analysis
- Identify whether a dataset is long or wide and reshape it from one to the other

Open class10.do if you haven't already.

Collapse to coarsen the unit of analysis

student	school	gpa	pop
1	A	3.3	1411
2	A	3.2	1411
3	B	2.9	2692
4	B	3.0	2692



school	mean_gpa	mean_pop
A	3.25	1411
B	2.95	2692

We can collapse our data to easily calculate summary statistics.

What is the unit of analysis of class10.dta?

We can collapse our data to easily calculate summary statistics.

What is the unit of analysis of class10.dta?

1a. Collapse the data from person-year to year:

```
collapse (count) id (mean) income hours age white
female , by(year))
```

We can collapse our data to easily calculate summary statistics.

What is the unit of analysis of class10.dta?

1a. Collapse the data from person-year to year:

```
collapse (count) id (mean) income hours age white
female , by(year))
```

1b. Run the code listed to rename your variables appropriately.

We can collapse our data to easily calculate summary statistics.

What is the unit of analysis of class10.dta?

1a. Collapse the data from person-year to year:

```
collapse (count) id (mean) income hours age white
female , by(year))
```

1b. Run the code listed to rename your variables appropriately.

1c. Save this new data frame as class10collapsed1.dta:

We can collapse our data to easily calculate summary statistics.

What is the unit of analysis of class10.dta?

1a. Collapse the data from person-year to year:

```
collapse (count) id (mean) income hours age white
female , by(year))
```

1b. Run the code listed to rename your variables appropriately.

1c. Save this new data frame as class10collapsed1.dta:

```
save class10collapsed1, replace
```

We can collapse our data to easily calculate summary statistics.

2a. Now use the `collapse` command to calculate the mean annual income and hours worked for each year and race separately :

```
collapse (count) id (mean) income hours, by(year white)
```

We can collapse our data to easily calculate summary statistics.

2a. Now use the `collapse` command to calculate the mean annual income and hours worked for each year and race separately :

```
collapse (count) id (mean) income hours, by(year white)
```

2b. Run the code listed to rename your variables appropriately.

2c. Save this new data frame as class10collapsed2.dta:

```
save class10collapsed2, replace
```

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta?

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

3a. Sort your data:

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

3a. Sort your data:

```
sort year id
```

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

3a. Sort your data:

```
sort year id
```

3b. Generate no. of observations (i.e, count of `id` variable) by year:

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

3a. Sort your data:

```
sort year id
```

3b. Generate no. of observations (i.e, count of `id` variable) by year:

```
gen k=1
bysort year:  egen no_obs = sum(k)
```

Let's use `bysort` command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

3a. Sort your data:

```
sort year id
```

3b. Generate no. of observations (i.e, count of `id` variable) by year:

```
gen k=1
bysort year:  egen no_obs = sum(k)
```

3c. Generate means of `income, hours, age, female, white` by year:

Let's use bysort command to generate summary statistics

What is the unit of analysis of class10.dta? Let's change it to year level:

3a. Sort your data:

```
sort year id
```

3b. Generate no. of observations (i.e, count of id variable) by year:

```
gen k=1
bysort year:  egen no_obs = sum(k)
```

3c. Generate means of income, hours, age, female, white by year:

```
bysort year:  egen incomemean = mean(income)
bysort year:  egen hoursmean = mean(hours)
bysort year:  egen agemean = mean(age)
bysort year:  egen whitemean = mean(white)
bysort year:  egen femalemean = mean(female)
```

Compare your answers for 3c. with1b.

Let's generate year-race summary statistics using `bysort` command

4a. Sort your data:

```
sort year id
```

Let's generate year-race summary statistics using `bysort` command

4a. Sort your data:

```
sort year id
```

4b. Generate no. of observations (i.e, count of `id` variable) by year-white variables:

```
gen k=1
```

Let's generate year-race summary statistics using `bysort` command

4a. Sort your data:

```
sort year id
```

4b. Generate no. of observations (i.e, count of `id` variable) by year-white variables:

```
gen k=1
bysort year white:  egen no_obs = sum(k)
```

Let's generate year-race summary statistics using `bysort` command

4a. Sort your data:

```
sort year id
```

4b. Generate no. of observations (i.e, count of `id` variable) by year-white variables:

```
gen k=1
bysort year white:  egen no_obs = sum(k)
```

4c. Generate means of `income, hours` by year-white variables:

Let's generate year-race summary statistics using `bysort` command

4a. Sort your data:

```
sort year id
```

4b. Generate no. of observations (i.e, count of `id` variable) by year-white variables:

```
gen k=1
bysort year white:  egen no_obs = sum(k)
```

4c. Generate means of `income, hours` by year-white variables:

```
bysort year white:  egen incomemean = mean(income)
bysort year white:  egen hoursmean = mean(hours)
```

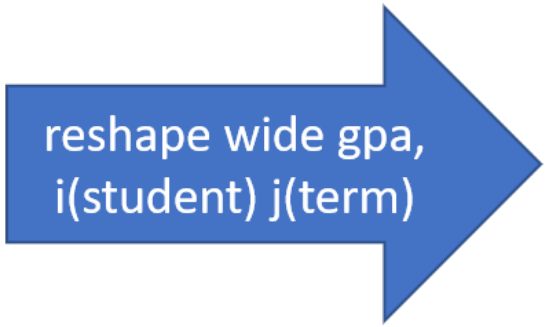
Compare your answers for 4c. with 2b.

Changing data form: "long" vs "wide"

Often we have multiple entries of a given variable for the same unit (e.g. multiple GPAs for the same student observed once per quarter).

We can present these data as long or wide and convert between the two using reshape.

student	term	gpa
1	fall	3.3
1	spring	3.2
2	fall	2.9
2	spring	3.0

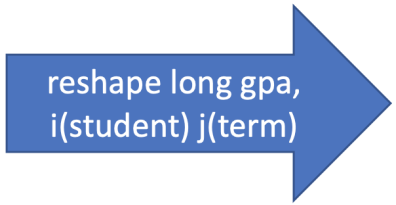


student	fall_ gpa	spring_ gpa
1	3.3	3.2
2	2.9	3.0

Changing data form: "long" vs "wide"

Similarly, we can reshape back from wide form to long form

student	fall_gpa	spring_gpa
1	3.3	3.2
2	2.9	3



student	term	gpa
1	fall	3.3
1	spring	3.2
2	fall	2.9
2	spring	3

Class exercise: *reshape*

- Open dataset `class10.dta`

  ```
  use "class10.dta", replace
  ```

- Verify that the data is in *long* format
- reshape the data into **wide** format, ie.

  person-year $\Rightarrow$ person
- reshape back in **long** form, i.e.:

  person $\Rightarrow$ person-year

Here are the commands/operators we covered today:

- `collapse`
- `reshape wide`
- `reshape long`