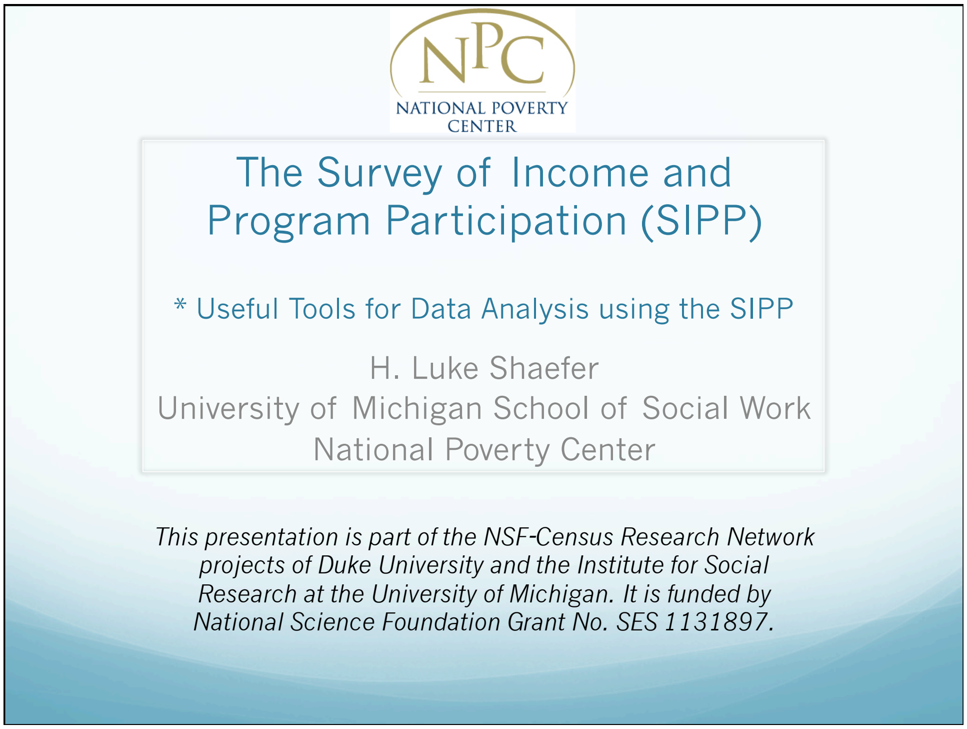


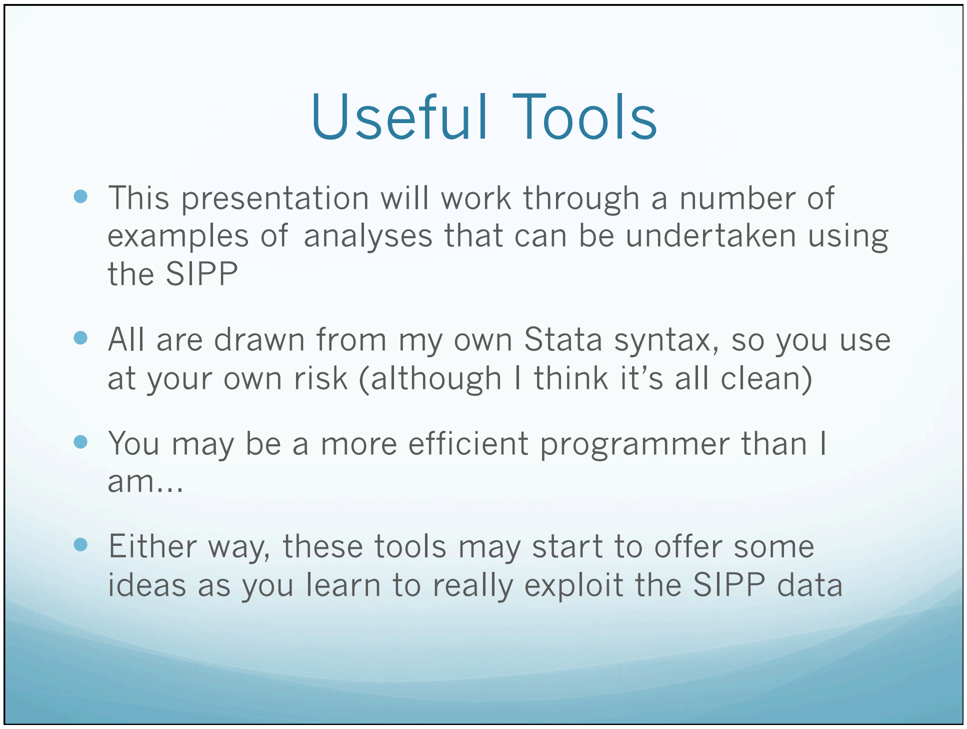
NPC
NATIONAL POVERTY
CENTER

# The Survey of Income and Program Participation (SIPP)

* Useful Tools for Data Analysis using the SIPP

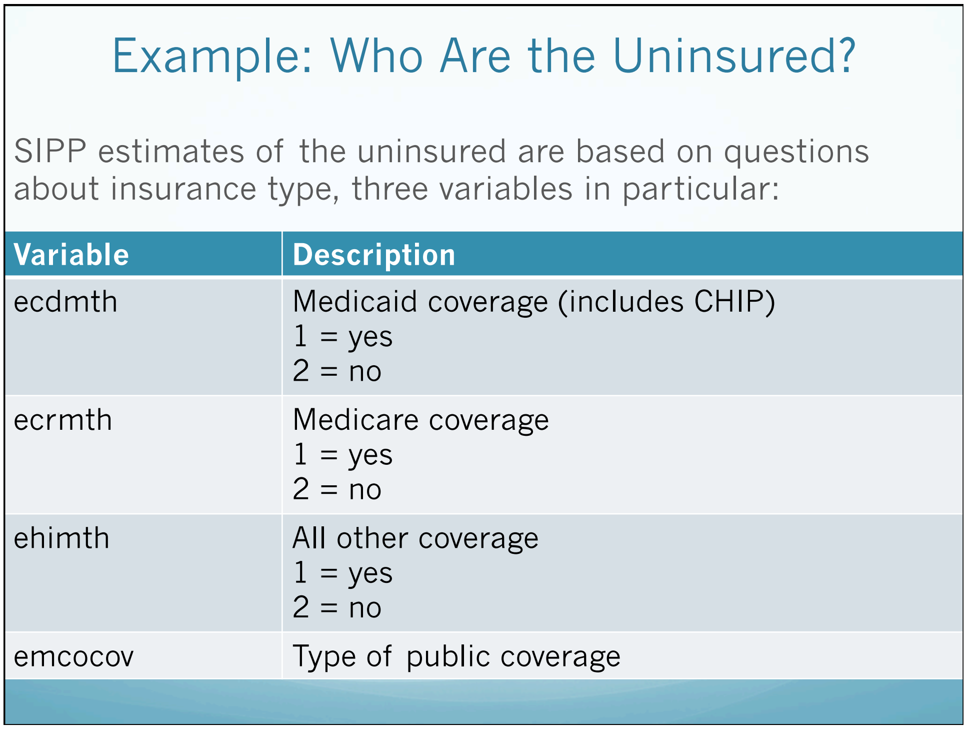
H. Luke Shaefer
University of Michigan School of Social Work
National Poverty Center

*This presentation is part of the NSF-Census Research Network projects of Duke University and the Institute for Social Research at the University of Michigan. It is funded by National Science Foundation Grant No. SES 1131897.*



## Useful Tools

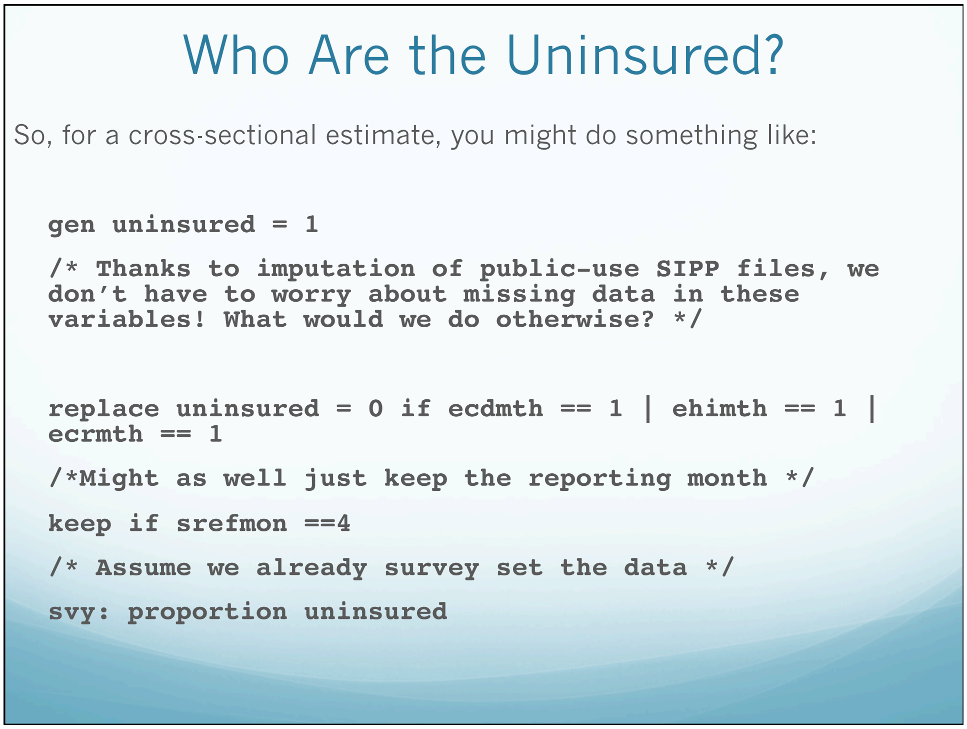
- This presentation will work through a number of examples of analyses that can be undertaken using the SIPP
- All are drawn from my own Stata syntax, so you use at your own risk (although I think it's all clean)
- You may be a more efficient programmer than I am...
- Either way, these tools may start to offer some ideas as you learn to really exploit the SIPP data



# Example: Who Are the Uninsured?

SIPP estimates of the uninsured are based on questions about insurance type, three variables in particular:

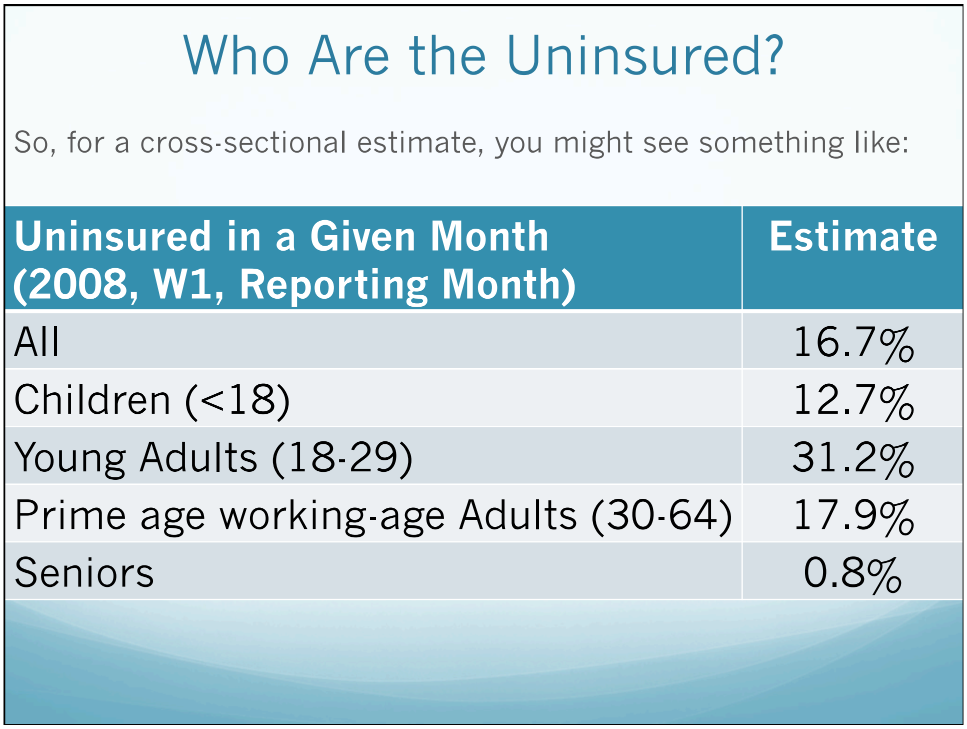
| Variable | Description |
|---|---|
| ecdmth | Medicaid coverage (includes CHIP)<br>1 = yes<br>2 = no |
| ecrmth | Medicare coverage<br>1 = yes<br>2 = no |
| ehimth | All other coverage<br>1 = yes<br>2 = no |
| emcocov | Type of public coverage |



# Who Are the Uninsured?

So, for a cross-sectional estimate, you might do something like:

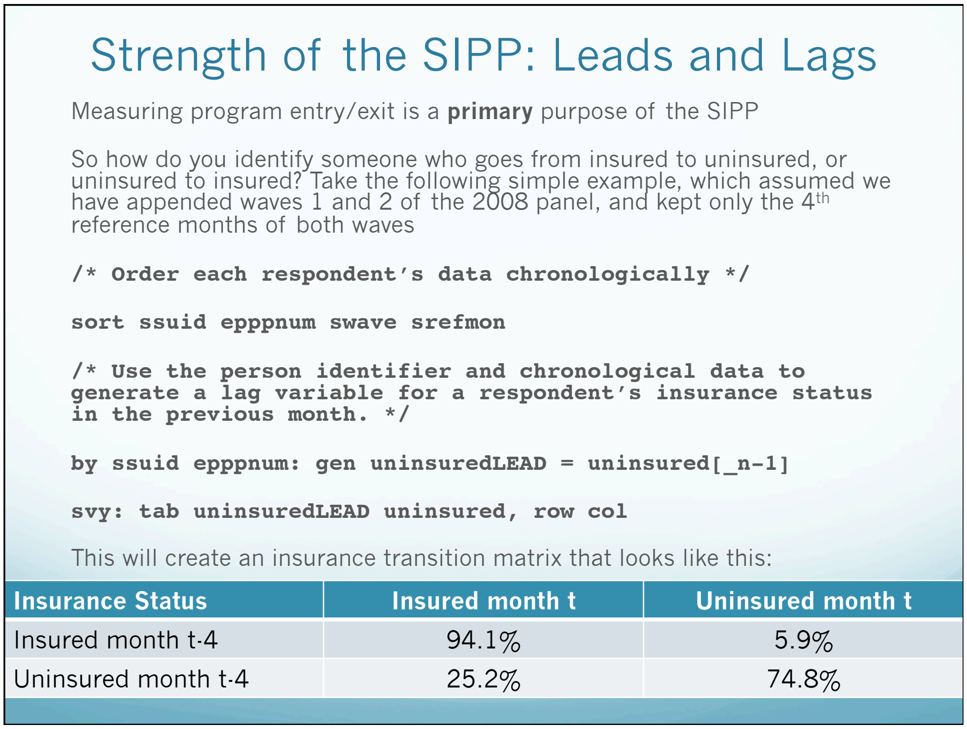
```
gen uninsured = 1

/* Thanks to imputation of public-use SIPP files, we
don’t have to worry about missing data in these
variables! What would we do otherwise? */


replace uninsured = 0 if ecdmth == 1 | ehimth == 1 |
ecrmth == 1

/*Might as well just keep the reporting month */

keep if srefmon ==4

/* Assume we already survey set the data */

svy: proportion uninsured
```



## Who Are the Uninsured?

So, for a cross-sectional estimate, you might see something like:

| Uninsured in a Given Month (2008, W1, Reporting Month) | Estimate |
|---|---|
| All | 16.7% |
| Children (<18) | 12.7% |
| Young Adults (18-29) | 31.2% |
| Prime age working-age Adults (30-64) | 17.9% |
| Seniors | 0.8% |



## Strength of the SIPP: Leads and Lags

Measuring program entry/exit is a **primary** purpose of the SIPP

So how do you identify someone who goes from insured to uninsured, or uninsured to insured? Take the following simple example, which assumed we have appended waves 1 and 2 of the 2008 panel, and kept only the 4th reference months of both waves

```
/* Order each respondent's data chronologically */

sort ssuid epppnum swave srefmon

/* Use the person identifier and chronological data to
generate a lag variable for a respondent's insurance status
in the previous month. */

by ssuid epppnum: gen uninsuredLEAD = uninsured[_n-1]

svy: tab uninsuredLEAD uninsured, row col
```

This will create an insurance transition matrix that looks like this:

| Insurance Status | Insured month t | Uninsured month t |
|---|---|---|
| Insured month t-4 | 94.1% | 5.9% |
| Uninsured month t-4 | 25.2% | 74.8% |

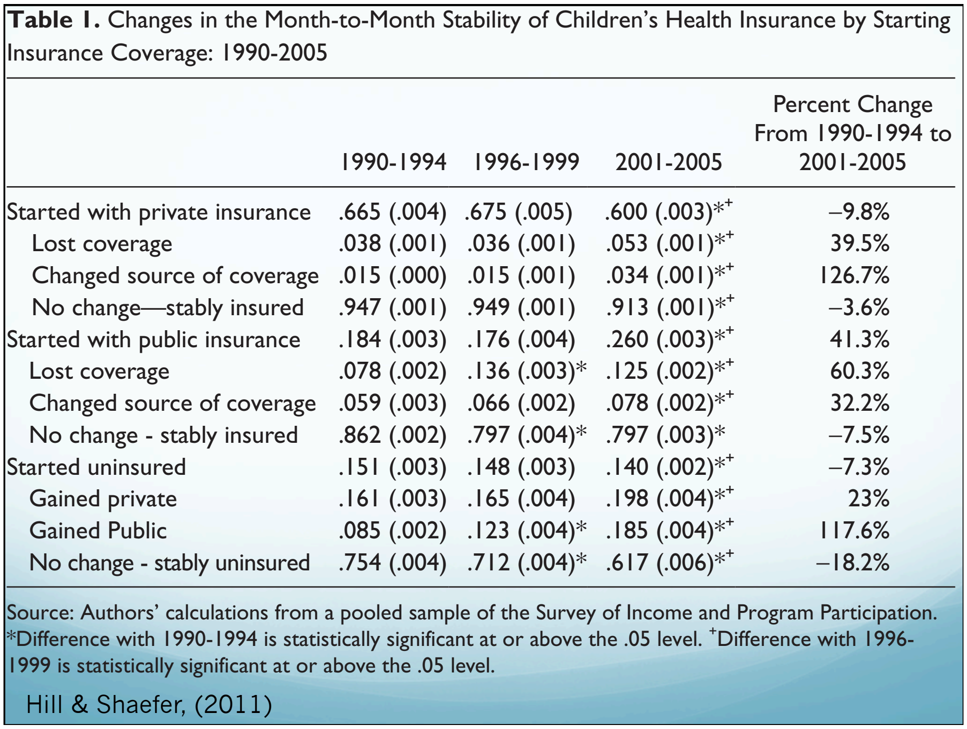
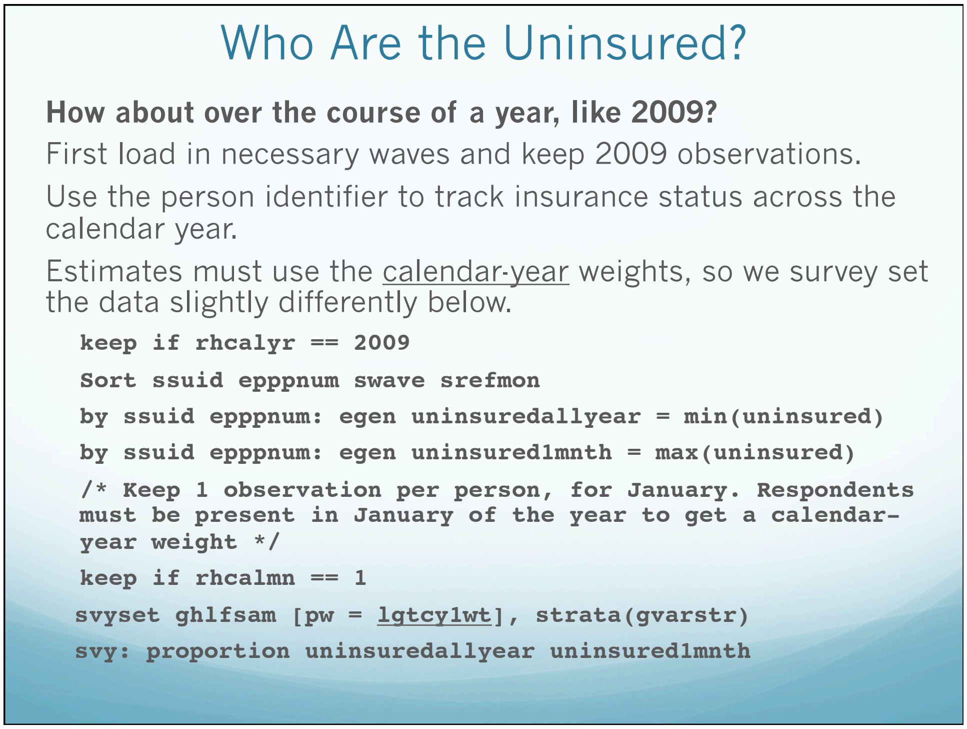


**Table 1.** Changes in the Month-to-Month Stability of Children's Health Insurance by Starting Insurance Coverage: 1990-2005

| | 1990-1994 | 1996-1999 | 2001-2005 | Percent Change From 1990-1994 to 2001-2005 |
|---|---|---|---|---|
| Started with private insurance | .665 (.004) | .675 (.005) | .600 (.003)*+ | –9.8% |
| Lost coverage | .038 (.001) | .036 (.001) | .053 (.001)*+ | 39.5% |
| Changed source of coverage | .015 (.000) | .015 (.001) | .034 (.001)*+ | 126.7% |
| No change—stably insured | .947 (.001) | .949 (.001) | .913 (.001)*+ | –3.6% |
| Started with public insurance | .184 (.003) | .176 (.004) | .260 (.003)*+ | 41.3% |
| Lost coverage | .078 (.002) | .136 (.003)* | .125 (.002)*+ | 60.3% |
| Changed source of coverage | .059 (.003) | .066 (.002) | .078 (.002)*+ | 32.2% |
| No change - stably insured | .862 (.002) | .797 (.004)* | .797 (.003)* | –7.5% |
| Started uninsured | .151 (.003) | .148 (.003) | .140 (.002)*+ | –7.3% |
| Gained private | .161 (.003) | .165 (.004) | .198 (.004)*+ | 23% |
| Gained Public | .085 (.002) | .123 (.004)* | .185 (.004)*+ | 117.6% |
| No change - stably uninsured | .754 (.004) | .712 (.004)* | .617 (.006)*+ | –18.2% |

Source: Authors' calculations from a pooled sample of the Survey of Income and Program Participation. *Difference with 1990-1994 is statistically significant at or above the .05 level. +Difference with 1996-1999 is statistically significant at or above the .05 level.

Hill & Shaefer, (2011)



## Who Are the Uninsured?

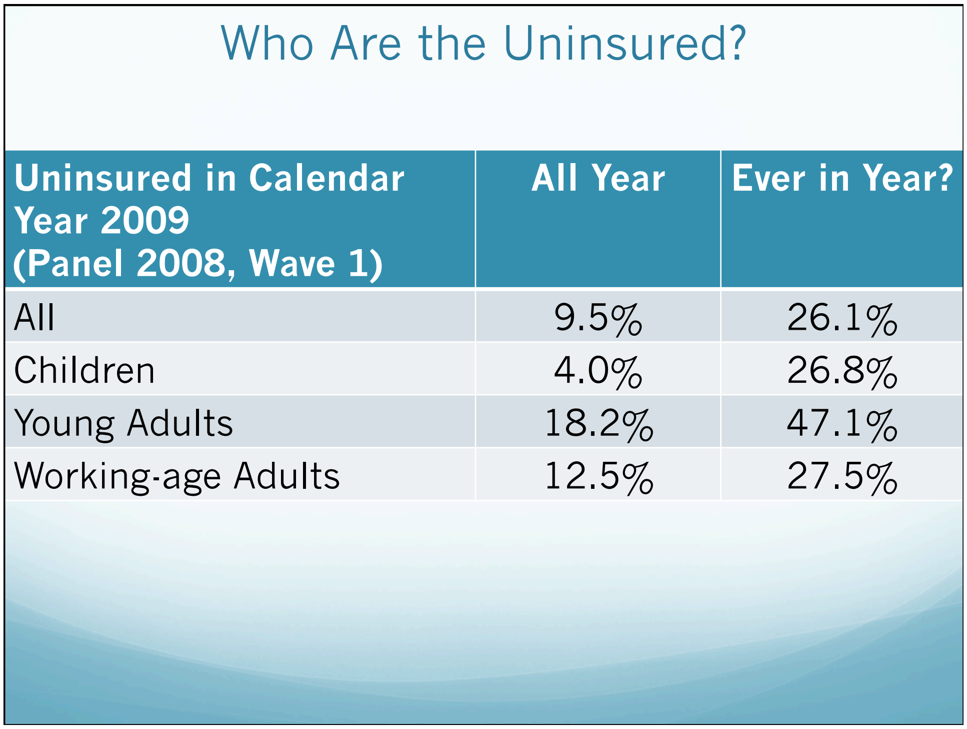
**How about over the course of a year, like 2009?**

First load in necessary waves and keep 2009 observations.

Use the person identifier to track insurance status across the calendar year.

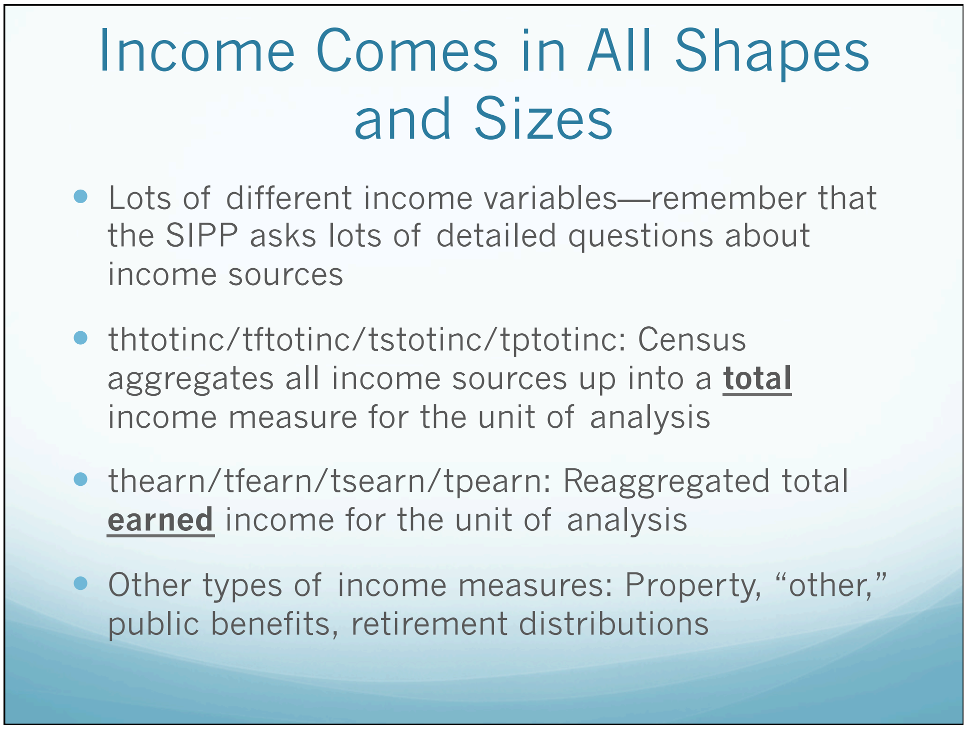
Estimates must use the calendar-year weights, so we survey set the data slightly differently below.

```
 keep if rhcalyr == 2009
 Sort ssuid epppnum swave srefmon
 by ssuid epppnum: egen uninsuredallyear = min(uninsured)
 by ssuid epppnum: egen uninsured1mnth = max(uninsured)
 /* Keep 1 observation per person, for January. Respondents
 must be present in January of the year to get a calendar-
 year weight */
 keep if rhcalmn == 1
svyset ghlfsam [pw = lgtcy1wt], strata(gvarstr)
svy: proportion uninsuredallyear uninsured1mnth
```



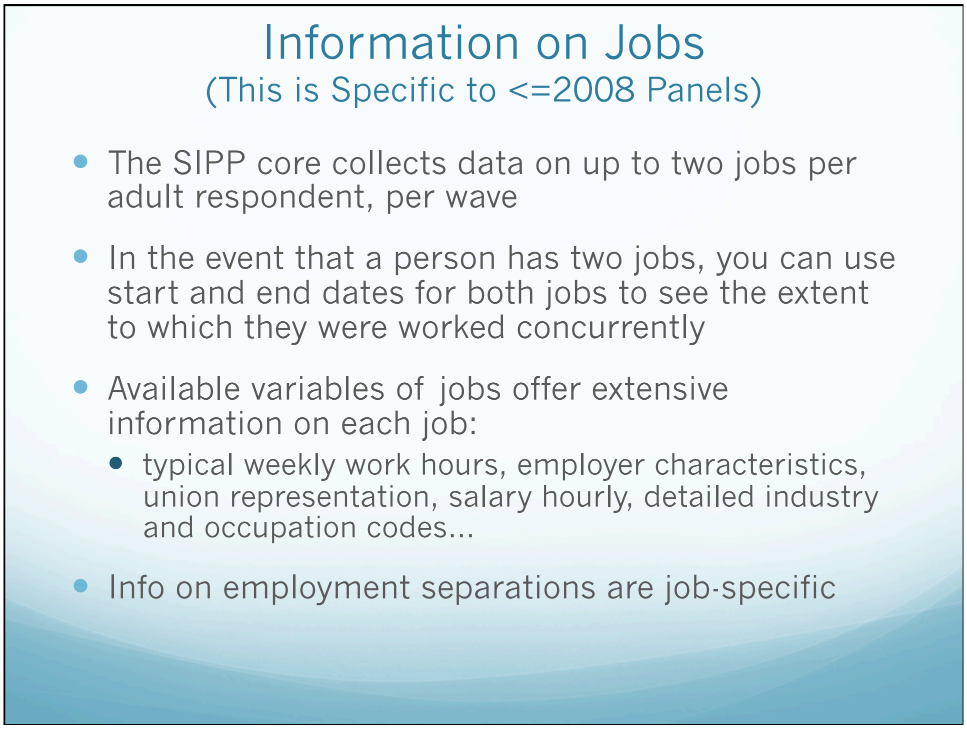
## Who Are the Uninsured?

| Uninsured in Calendar Year 2009<br>(Panel 2008, Wave 1) | All Year | Ever in Year? |
|---|---|---|
| All | 9.5% | 26.1% |
| Children | 4.0% | 26.8% |
| Young Adults | 18.2% | 47.1% |
| Working-age Adults | 12.5% | 27.5% |



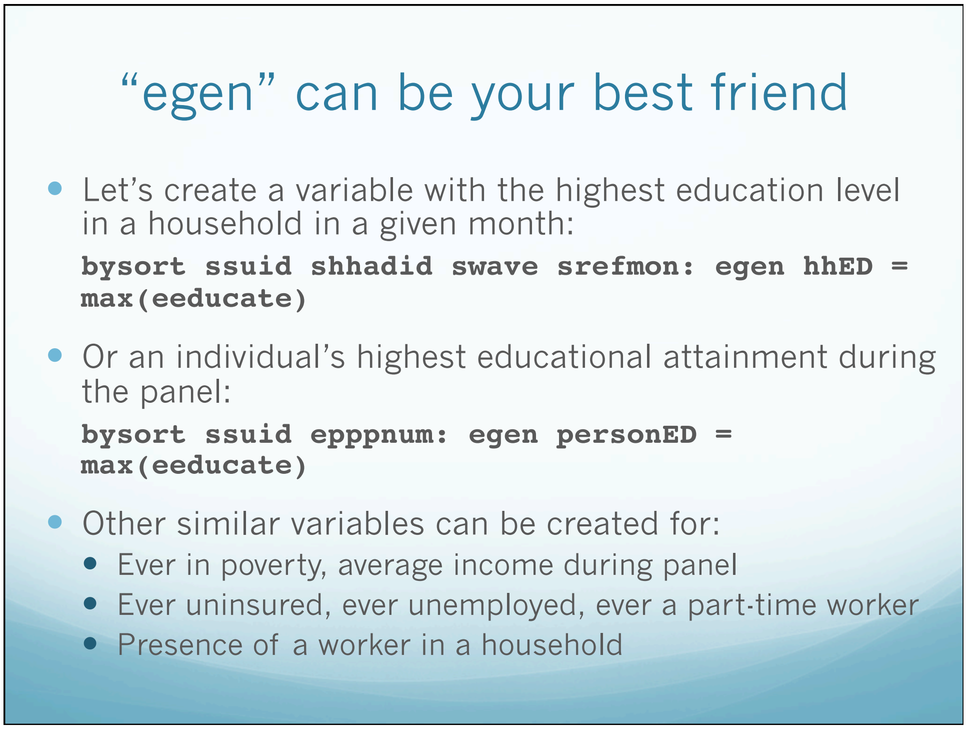
## Income Comes in All Shapes and Sizes

- Lots of different income variables—remember that the SIPP asks lots of detailed questions about income sources
- thtotinc/tftotinc/tstotinc/tptotinc: Census aggregates all income sources up into a **total** income measure for the unit of analysis
- thearn/tfearn/tsearn/tpearn: Reaggregated total **earned** income for the unit of analysis
- Other types of income measures: Property, “other,” public benefits, retirement distributions



## Information on Jobs
(This is Specific to <=2008 Panels)

- The SIPP core collects data on up to two jobs per adult respondent, per wave
- In the event that a person has two jobs, you can use start and end dates for both jobs to see the extent to which they were worked concurrently
- Available variables of jobs offer extensive information on each job:
  - typical weekly work hours, employer characteristics, union representation, salary hourly, detailed industry and occupation codes…
- Info on employment separations are job-specific



## “egen” can be your best friend

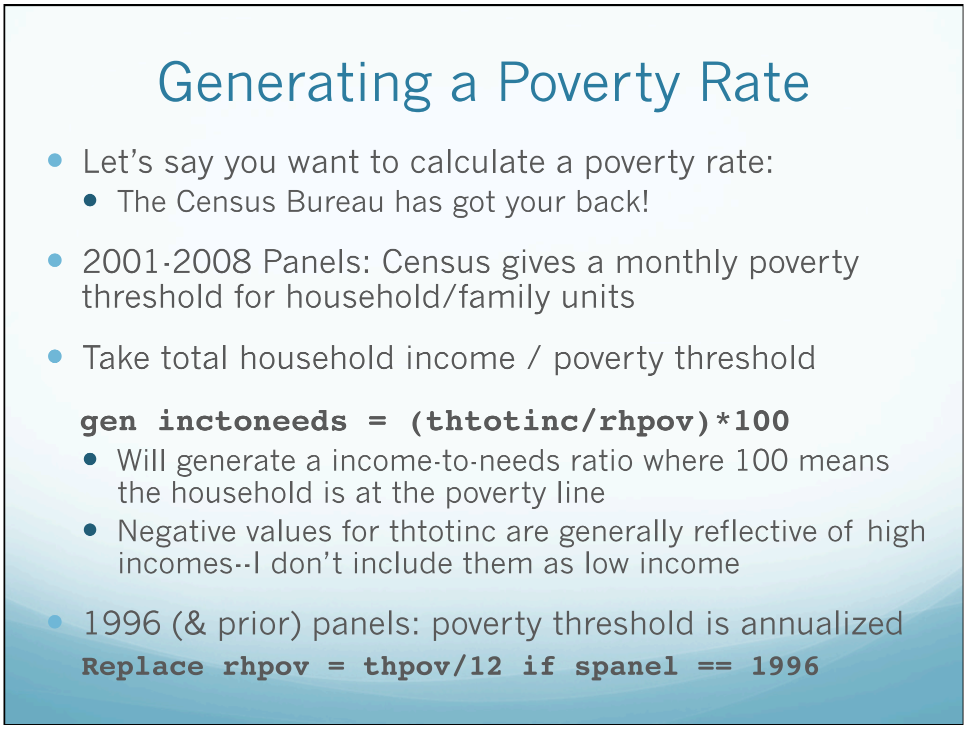
- Let’s create a variable with the highest education level in a household in a given month:

```
bysort ssuid shhadid swave srefmon: egen hhED =
max(eeducate)
```

- Or an individual’s highest educational attainment during the panel:

```
bysort ssuid epppnum: egen personED =
max(eeducate)
```

- Other similar variables can be created for:
  - Ever in poverty, average income during panel
  - Ever uninsured, ever unemployed, ever a part-time worker
  - Presence of a worker in a household



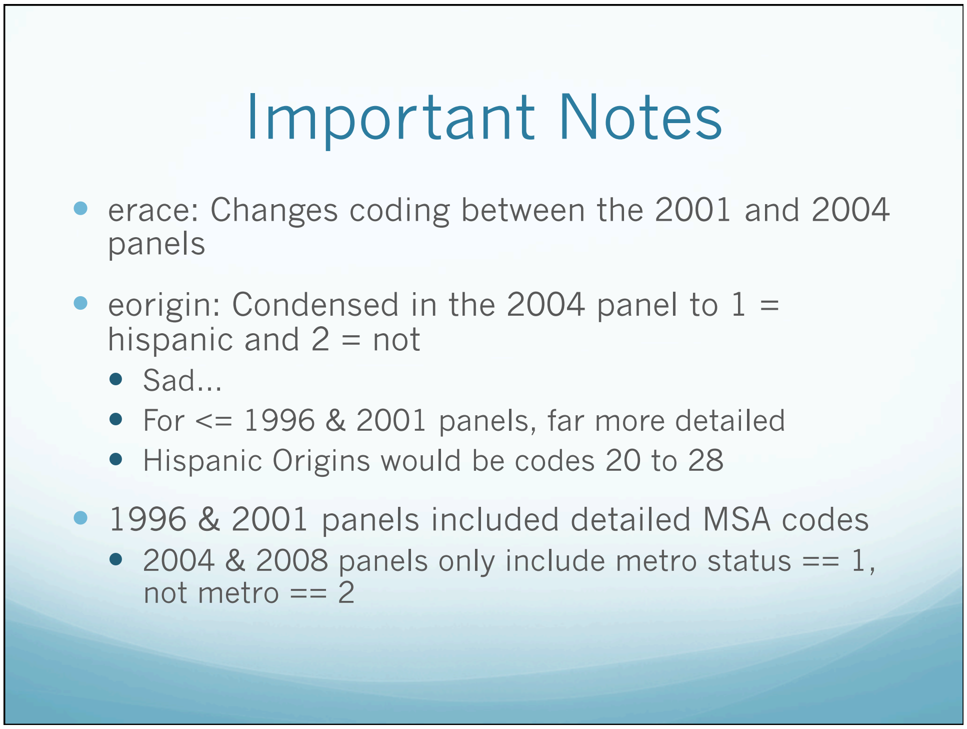
# Generating a Poverty Rate

- Let’s say you want to calculate a poverty rate:
  - The Census Bureau has got your back!
- 2001-2008 Panels: Census gives a monthly poverty threshold for household/family units
- Take total household income / poverty threshold

```
gen inctoneeds = (thtotinc/rhpov)*100
```

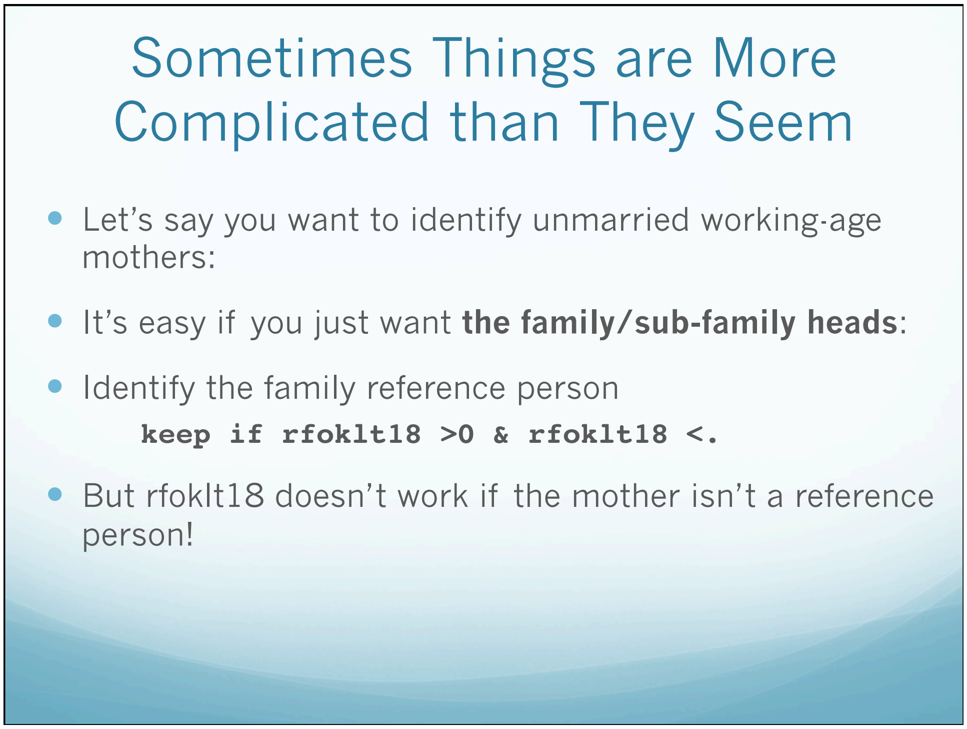
  - Will generate a income-to-needs ratio where 100 means the household is at the poverty line
  - Negative values for thtotinc are generally reflective of high incomes--I don’t include them as low income
- 1996 (& prior) panels: poverty threshold is annualized

```
Replace rhpov = thpov/12 if spanel == 1996
```



# Important Notes

- erace: Changes coding between the 2001 and 2004 panels
- eorigin: Condensed in the 2004 panel to 1 = hispanic and 2 = not
  - Sad…
  - For <= 1996 & 2001 panels, far more detailed
  - Hispanic Origins would be codes 20 to 28
- 1996 & 2001 panels included detailed MSA codes
  - 2004 & 2008 panels only include metro status == 1, not metro == 2

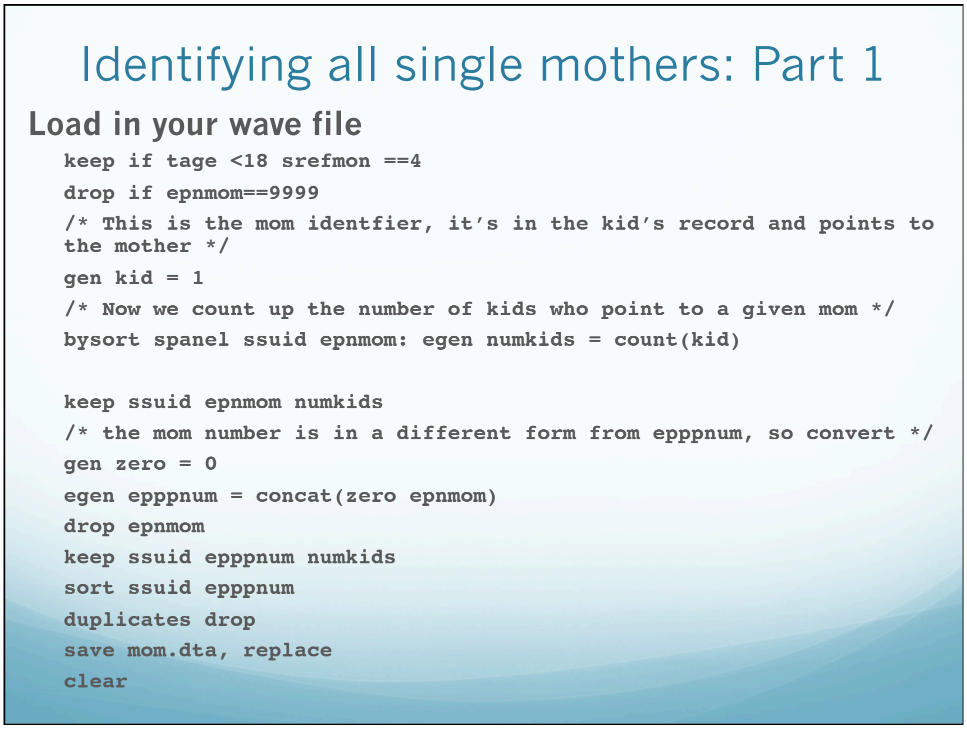


# Sometimes Things are More Complicated than They Seem

- Let’s say you want to identify unmarried working-age mothers:
- It’s easy if you just want **the family/sub-family heads**:
- Identify the family reference person

```
keep if rfoklt18 >0 & rfoklt18 <.
```

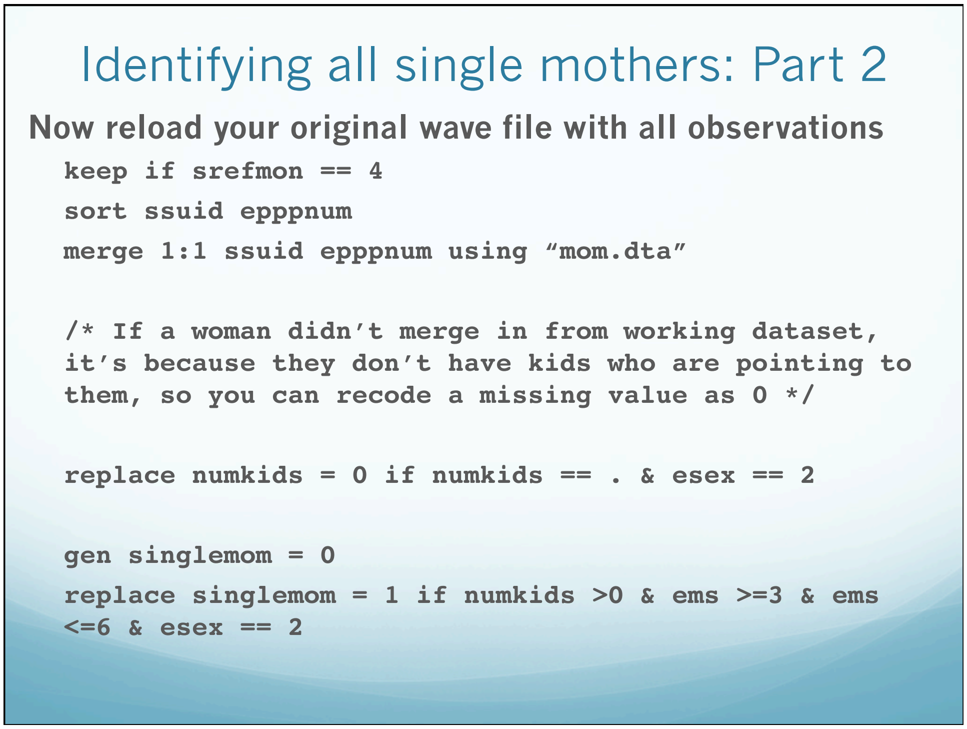
- But rfoklt18 doesn’t work if the mother isn’t a reference person!



# Identifying all single mothers: Part 1

**Load in your wave file**

```
keep if tage <18 srefmon ==4
drop if epnmom==9999
/* This is the mom identfier, it’s in the kid’s record and points to
the mother */
gen kid = 1
/* Now we count up the number of kids who point to a given mom */
bysort spanel ssuid epnmom: egen numkids = count(kid)

keep ssuid epnmom numkids
/* the mom number is in a different form from epppnum, so convert */
gen zero = 0
egen epppnum = concat(zero epnmom)
drop epnmom
keep ssuid epppnum numkids
sort ssuid epppnum
duplicates drop
save mom.dta, replace
clear
```



# Identifying all single mothers: Part 2

Now reload your original wave file with all observations

```
keep if srefmon == 4
sort ssuid epppnum
merge 1:1 ssuid epppnum using “mom.dta”

/* If a woman didn’t merge in from working dataset,
it’s because they don’t have kids who are pointing to
them, so you can recode a missing value as 0 */

replace numkids = 0 if numkids == . & esex == 2

gen singlemom = 0
replace singlemom = 1 if numkids >0 & ems >=3 & ems
<=6 & esex == 2
```