

```

cap cd "Z:\\"
log using "Gallup_income.log", text replace
clear
set mem 2000m
set more off
use Gallup_work.dta /*Don't use the original "file.dta" provided by Gallup */

* Deal with problems in income data
for X in any "Kenya" "Laos" "Uzbek": replace Inc_X=.
label define inc_pol 1 "Less than 700" 2 "701-925" 3 "926-1150" 4 "1151-1500" 5 "1501-2000" 6 "2001-2500" 7 "2501 or more"
label values Inc_Poland inc_pol

egen t=tag(cty)
gen inc_lower=.
gen inc_upper=.
gen inc=.
gen ln_inc2=.

gen k_pareto=.
gen top_coded=.
gen bottom_coded=.
foreach v of varlist Inc_* {
    summ `v'
    if r(N)==0 {
        rename `v' empty_`v'
    }
    else {
        local vl: value label `v'
        display("value label`vl'!")
        if "`vl'"==" " {
            display("Unlabelled")
            replace inc=`v' if `v'~=.
            replace ln_inc2=ln(`v') if `v'~=.
        }
        else {
            levelsof(`v'), local(values)
            foreach l of local values {
                local lb: label `vl' `l'
                display("The label is `lb'")
                if "`lb'"=="0" | "`lb'"=="None" {
                    local lb="0 0"
                }
                local lb=subinstr("`lb'", "None", "0 0", 2)
                local lb=subinstr("`lb'", "-", " ", 2)
                local lb=subinstr("`lb'", ",", "", 6)
                local lb=subinstr("`lb'", "Less than", "0", 2)
                local lb=subinstr("`lb'", "less than", "0", 2)
                local lb=subinstr("`lb'", "Under", "0", 2)
                local lb=subinstr("`lb'", "Up", "0", 2)
                local lb=subinstr("`lb'", "or more", " -9", 2)
                local lb=subinstr("`lb'", "or More", " -9", 2)
                display("Restate: `lb'")
                local lower: word 1 of `lb'
                local upper: word 2 of `lb'
                replace inc_lower=`lower' if `v'==`l'
                if "`upper'"~="" {
                    replace inc_upper=`upper' if `v'==`l'
                    if `lower'>`upper' & `upper'>0 {
                        replace inc_lower=`upper' if `v'==`l'
                        replace inc_upper=`lower' if `v'==`l'
                    }
                }
            }
        }
    }
    replace inc_lower=300001 if Inc_Benin==13
    * Deal with zero incomes: Put them back into the grouping above them
    summ inc_lower if `v'~=. & inc_lower~-9
    local ll=r(min)

```

```

summ inc_upper if `v'~= . & inc_upper== -9
local lu=r(min)
if `ll'==0 & `lu'==0 {
    summ inc_lower if `v'~= . & inc_lower== -9 & inc_lower==0
    local l2=r(min)
    summ inc_upper if `v'~= . & inc_upper== -9 & inc_upper==0
    local u2=r(min)
    replace inc_lower=`l2' if `v'~= . & inc_lower==0 & inc_upper==0
    replace inc_upper=`u2' if `v'~= . & inc_lower==`l2' & inc_upper==0
    replace inc_lower=0 if `v'~= . & inc_lower==`l2' & inc_upper==`u2'
}
summ inc_lower if `v'~= . & inc_lower== -9
replace bottom_coded=1 if `v'~= . & inc_lower==r(min)
* Deal with top-codes
qui summ inc_upper if `v'~= . & inc_upper== -9
local u=r(max)
qui summ inc_lower if `v'~= . & inc_upper== -9
local b=r(max)
qui summ `v' [w=wt] if inc_upper==`u' & inc_lower==`b'
local ftop2=r(sum_w)
qui summ `v' [w=wt] if inc_upper== -9
local ftop=r(sum_w)
local k=(ln(`ftop2'+`ftop')-ln(`ftop'))/(ln(`u')-ln(`b'))
replace k_pareto=`k' if `v'~= .
local k=max(2,`k')
* Replicate what we did previously
replace inc=(inc_lower+inc_upper)/2 if `v'~= .
replace top_coded=1 if inc_upper== -9 & `v'~= .
replace inc=`u'*`k'/(`k'-1) if inc_upper== -9 & `v'~= .
* Do the interval regression instead
gen u=ln(inc_upper)
replace u=. if inc_upper== -9
gen l=ln(inc_lower)
replace l=. if inc_lower==0
intreg l u if `v'~= . [pw=wt]
egen grp=group(inc_lower inc_upper) if `v'~= .
levelsof grp, local(groups)
foreach g of local(groups) {
    qui summ l if `v'~= . & grp==`g'
    local lower=r(mean)
    qui summ u if `v'~= . & grp==`g'
    local upper=r(mean)
    predict hat, e(`lower', `upper')
    replace ln_inc2=hat if `v'~= . & grp==`g'
    drop hat
}
drop l u grp
}

}

rename inc inc_old
gen inc=exp(ln_inc2)
la var inc_old "Income variable used in conference draft"
la var inc "Income: interval estimation on log-normality"
drop ln_inc2

gen ln_inc=ln(inc)

/*
gen inc2=.
for X in num 0/28 \ Y in num 0 0.01 1 730 1100 1500 2000 2750 3500 4250 5000 5750 6500
7250 9000 11000 13000 14000 18000 20000 24000 28000 32000 36000 40000 45000 55000 75000
125000 \ Z in num 0 .99 2 1099 1499 1999 2749 3499 4249 4999 5749 6499 7249 8999 10999
12999 13999 17999 19999 23999 27999 31999 35999 39999 44999 54999 74999 124999 250000:
replace inc2=(Y+Z)/2 if Income_ID_29==X
la var inc2 "Midpoint of Income_ID_29"
*/

```

```
compress
save gallup_work_inc, replace

for var Inc_*: table X, c(m inc m inc_old n inc n inc_old)
log close
```