

APPENDIX TO IHIS DATA BRIEF NO. 1
SUPPLEMENTARY MATERIALS

Section 1. Variables Used to Define Multigenerational Families and Generate Analysis Results

Table 1. Variables Used to Identify Multigenerational Families in the Data Brief

IHIS Name	Description	Codes
<u>FAMREL</u>	Relationship to family reference person	Codes 1 to 17 (see website)
<u>HHX</u>	Household number (from NHIS)	
<u>FMX</u>	Family number (from NHIS)	
<u>AGE</u>	Age	0 to 85 (top coded)

Table 2. Variables Used for the Analyses

IHIS Name	Description	Codes
<u>YEAR</u>	Survey year	1998 to 2013
<u>STRATA</u>	Stratum for variance estimation	
<u>PSU</u>	Primary sampling unit (PSU) for variance estimation	
<u>PERWEIGHT</u>	Final annual person weight	
<u>FWEIGHT</u>	Final annual family weight	
<u>GOTSTAMPFAM</u>	Any family member received food stamp/SNAP benefits last calendar year	1: No 2: Yes 7 to 9: Unknown
<u>STAMPMO</u>	Number of months received food stamp last year	1 month to 12 months
<u>FSSTAT</u>	Family-level food security status in the last 30 days	1: Food security 2: Low food security 3: Very low food security

Section 2. Stata Code

```
local dir `replace with directory where data are stored'
```

```
local data ``dir'replace with name of IHIS extract"
```

```
capture log close
```

```
log using ``dir'data_brief.log", replace
```

```
use ``data"', clear
```

```
set more off
```

```
*IDENTIFY GENERATIONS & GENERATE INDICATOR FOR MULTIGENERATIONAL FAMILIES*
```

```
/*Base generation: family reference person, spouse, unmarried partner, or sibling of family reference person*/
```

```
gen base = (famrel == 1 | famrel == 2 | famrel == 3 | famrel == 8)
```

```
bysort year hhx fmx: egen basegen = max(base) /*indicates if a family has a base generation*/
```

```
/*Parent generation: parent, aunt or uncle, or legal guardian of family reference person*/
```

```
gen parent = (famrel == 7 | famrel == 10 | famrel == 16 | famrel == 17)
```

```
bysort year hhx fmx: egen parentgen = max(parent)
```

```
/*Grandparent generation: grandparent of family reference person*/
```

```
gen grandpa = (famrel == 9)
```

```
bysort year hhx fmx: egen grandpagen = max(grandpa)
```

```
/*ADULT children generation: child, child of unmarried partner, or niece/nephew aged 19+ of family reference person*/
```

```
gen adultchild = ((famrel == 4 | famrel == 5 | famrel == 11) & age >= 19)
```

```
bysort year hhx fmx: egen adultchildgen = max(adultchild)
```

```
/*YOUNG children generation: child, child of unmarried partner, or niece/nephew aged <19 of family reference person*/
```

```
gen youngchild = ((famrel == 4 | famrel == 5 | famrel == 11) & age < 19)
```

```
bysort year hhx fmx: egen youngchildgen = max(youngchild)
```

```
/*Children generation (regardless of age): child, child of unmarried partner, or niece/nephew of family reference person*/
```

```
gen anychild = (famrel == 4 | famrel == 5 | famrel == 11)
```

```
bysort year hhx fmx: egen anychildgen = max(anychild)
```

```
/*Grandchildren generation: grandchildren of family reference person*/
```

```
gen grandchild = (famrel == 6)
```

```
bysort year hhx fmx: egen grandchildgen = max(grandchild)
```

```
/*Count number of generations in a family*/
```

```
egen totalgen = rowtotal(basegen parentgen grandpagen anychildgen grandchildgen)
```

```
tab totalgen /*11 missing cases due to missing values of the family relationship variable FAMREL*/
```

```
/*Generate multigeneration variable*/
```

```
gen multigen = .
```

```
replace multigen = 1 if basegen == 1 & parentgen == 0 & grandpagen == 0 & adultchildgen == 1 & youngchildgen == 0 & grandchildgen == 0 /*head and adult children*/
```

```
replace multigen = 2 if basegen == 1 & parentgen == 0 & grandpagen == 0 & adultchildgen == 1 & youngchildgen == 1 & grandchildgen == 0 /*head, adult children, and young children*/
```

```

replace multigen = 3 if basegen == 1 & parentgen == 0 & grandpagen == 0 & anychildgen == 1 & grandchildgen == 1
/*head, any children, and grandchildren*/
replace multigen = 4 if basegen == 1 & parentgen == 0 & grandpagen == 0 & anychildgen == 0 & grandchildgen == 1
/*head and grandchildren*/
replace multigen = 5 if basegen == 1 & parentgen == 1 & grandpagen == 0 & anychildgen == 0 & grandchildgen == 0
/*parents and head*/
replace multigen = 6 if basegen == 1 & parentgen == 1 & grandpagen == 0 & anychildgen == 1 & grandchildgen == 0
/*parent, head, and any children*/
replace multigen = 7 if basegen == 1 & parentgen == 1 & grandpagen == 0 & anychildgen == 0 & grandchildgen == 1
/*parent, head, and grandchildren*/
replace multigen = 8 if basegen == 1 & parentgen == 1 & grandpagen == 1 & anychildgen == 0 & grandchildgen == 0
/*grandparents, parents, and head*/
replace multigen = 9 if basegen == 1 & parentgen == 0 & grandpagen == 1 & anychildgen == 0 & grandchildgen == 0
/*grandparents and head*/
replace multigen = 10 if basegen == 1 & parentgen == 0 & grandpagen == 1 & anychildgen == 1 & grandchildgen == 0
/*grandparents, head, and any children*/
replace multigen = 11 if basegen == 1 & parentgen == 0 & grandpagen == 1 & anychildgen == 0 & grandchildgen == 1
/*grandparents, head, and grandchildren*/
replace multigen = 12 if totalgen > 3 & multigen == . /*more-than-three generation families*/

```

```

label define multi 1 "head and adult children" 2 "head, adult children, and young children" 3 "head, any children, and
grandchildren" 4 "head and grandchildren" 5 "parents and head" 6 "parent, head, and any children" 7 "parent, head,
and grandchildren" 8 "grandparents, parents, and head" 9 "grandparents and head" 10 "grandparents, head, and any
children" 11 "grandparents, head, and grandchildren" 12 "more than three generations"
label values multigen multi

```

```

/*Indicator for any type of multigenerational families*/

```

```

gen anymultigen = 0

```

```

replace anymultigen = 1 if multigen >= 1 & multigen <= 12

```

```

replace anymultigen = . if totalgen == 0 /* 11 persons (or 6 families) have missing values in FAMREL*/

```

```

/*Indicator for single-generation families*/

```

```

gen singlegen = 0

```

```

replace singlegen = 1 if basegen == 1 & parentgen == 0 & grandpagen == 0 & anychildgen == 0 & grandchildgen == 0

```

```

/*families with only the head generation*/

```

```

replace singlegen = 1 if basegen == 1 & parentgen == 0 & grandpagen == 0 & adultchildgen == 0 & youngchildgen == 1 &
grandchildgen == 0 /*head and young children under 19*/

```

```

replace singlegen = . if totalgen == 0 /* 11 persons (or 6 families) have missing values in FAMREL*/

```

```

/*Less-detailed family structure: multigenerational or single-generation family*/

```

```

gen famstruc1 = .

```

```

replace famstruc1 = 0 if singlegen == 1

```

```

replace famstruc1 = 1 if anymultigen == 1

```

```

label define fam1 0 "single-generation family" 1 "multigenerational family"

```

```

label values famstruc1 fam1

```

```

/*More-detailed family structure*/

```

```

gen famstruc2 = .

```

```

replace famstruc2 = 1 if singlegen == 1 /*single generation*/

```

```

replace famstruc2 = 2 if multigen == 1 | multigen == 2 | multigen == 5 /*2 generations*/

```

```

replace famstruc2 = 3 if multigen == 3 | multigen == 6 | multigen == 8 /*3 generations*/

```

```

replace famstruc2 = 4 if multigen == 4 | multigen == 7 | multigen == 9 | multigen == 10 | multigen == 11 /*skipped
generations*/

```

```
replace famstruc2 = 5 if multigen == 12 /*more than 3 generations*/
label define fam2 1 "single-generation family" 2 "two generations" 3 "three generations" 4 "skipped generation" 5
"more than three generations"
label values famstruc2 fam2
```

```
*GENERATE VARIABLES FOR THE ANALYSES*
```

```
/*Young children aged <19 who live in multigenerational families*/
gen children = .
replace children = 0 if age < 19
replace children = 1 if famstruc1 == 1 & children == 0
```

```
/*Young adults aged 19-35 who live in multigenerational families*/
gen youngadult35 = .
replace youngadult35 = 0 if age >= 19 & age <= 35
replace youngadult35 = 1 if famstruc1 == 1 & youngadult35 == 0
```

```
/*Older adults aged 65+ living in multi-generational families*/
gen oldadult = .
replace oldadult = 0 if age >= 65
replace oldadult = 1 if famstruc1 == 1 & oldadult == 0
```

```
/*Generation binary variable for food security status*/
gen foodinsecure = .
replace foodinsecure = 0 if fsstat == 1 /*food secure*/
replace foodinsecure = 1 if fsstat == 2 | fsstat == 3 /*food insecure*/
label define fs 1 "food insecure" 0 "food secure"
label values foodinsecure fs
```

```
/*Recode IHIS variables: number of months received food stamp*/
replace stampmo = . if stampmo == 0 | (stampmo >= 97 & stampmo <= 99) /*missing (unknown) or not in universe*/
bysort year hhx fmx: egen stampmo_fam = max(stampmo) /*assign stampmo value to every member in a family*/
```

```
/*Recode IHIS variables: indicator for whether any family member received food stamp*/
replace gotstampfam = . if (gotstampfam >= 7 & gotstampfam <= 9) | gotstampfam == 0 /*missing (unknown) or not in universe*/
replace gotstampfam = 0 if gotstampfam == 1 /*Did not get food stamp*/
replace gotstampfam = 1 if gotstampfam == 2 /*Got food stamp*/
label define stampfam 0 "no" 1 "yes"
label values gotstampfam stampfam
```

```
/*Label variables*/
label var multigen "Detailed multigenerational family types"
label var singlegen "Single-generation family"
label var anymultigen "Multigenerational family (any types)"
label var famstruc1 "Less detailed family structure"
label var famstruc2 "More detailed family structure"
label var children "Children aged < 19 living in multigenerational or single-generation family"
label var youngadult35 "Young adults aged 19-35 living in multigenerational or single-generation family"
label var oldadult "Older adults aged 65+ living in multigenerational or single generational family"
label var foodinsecure "Family food security status"
label var stampmo_fam "Months that family received food stamp, last calendar year"
```

```

/*drop unnecessary variables*/
drop age fsstat base basegen parent parentgen grandpa grandpagen adultchild stampmo /*
*/ adultchildgen youngchild youngchildgen anychild anychildgen grandchild grandchildgen totalgen

save "finaldata.dta", replace

*****
*****
*ANALYSIS*

*TABLE 1 (family level, 1998-2013): Share of Families that are Multigenerational*
use "finaldata.dta", clear
set more off
svyset psu [pweight=fweight], strata(strata)

/*Trend of multigenerational (any types) and single-generation families 1998-2013*/
svy, subpop(if famrel == 1): mean famstruc1, over(year)

/*Trend of specific multigenerational family types*/
svy, subpop(if famrel == 1): proportion famstruc2, over(year)

*TABLE 2 (family level, pooled 2011-2013): Food Insecurity Prevalence by Family Structures, 2011-2013 IHIS Pooled
Sample*
use "finaldata.dta", clear
set more off
replace fweight = fweight / 3
svyset psu [pweight=fweight], strata(strata)

/*Food security status of detailed types of multigenerational families*/
svy, subpop(if famrel == 1 & year >= 2011 & year <= 2013): mean foodinsecure, over(famstruc2)
svy, subpop(if famrel == 1 & year >= 2011 & year <= 2013): reg foodinsecure i.famstruc2

/*Food security status of multigenerational families (any types)*/
svy, subpop(if famrel == 1 & year >= 2011 & year <= 2013): mean foodinsecure, over(famstruc1)
svy, subpop(if famrel == 1 & year >= 2011 & year <= 2013): reg foodinsecure i.famstruc1

*TABLE 3 (person level, pooled 2011-2013): Food Insecurity Prevalence Among Individuals Living in Single Generational
and Multigenerational Families, 2011-2013 IHIS Pooled Sample*
use "finaldata.dta", clear
set more off
replace perweight = perweight / 3
svyset psu [pweight=perweight], strata(strata)

/*Children aged < 19*/
svy, subpop(if year >= 2011 & year <= 2013): mean foodinsecure, over(children)
svy, subpop(if year >= 2011 & year <= 2013): reg foodinsecure i.children

/*Young adults 19-35*/
svy, subpop(if year >= 2011 & year <= 2013): mean foodinsecure, over(youngadult35)
svy, subpop(if year >= 2011 & year <= 2013): reg foodinsecure i.youngadult35

```

```
/*Older adults aged 65+*/  
svy, subpop(if year >= 2011 & year <= 2013): mean foodinsecure, over(oldadult)  
svy, subpop(if year >= 2011 & year <= 2013): reg foodinsecure i.oldadult
```

```
*TABLE 4 (family level, 1998-2013): Percentage of Families Receiving SNAP in Last Calendar Year, 1998-2013 IHIS*  
use "finaldata.dta", clear  
set more off  
svyset psu [pweight=fweight], strata(strata)
```

```
/*Trend of SNAP receipt for multigenerational families (all types) and single-generation families*/  
svy, subpop(if famrel == 1): mean gotstampfam, over(year) /*all types*/  
svy, subpop(if famrel == 1 & famstruc1 == 1): mean gotstampfam, over(year) /*multi-generation*/  
svy, subpop(if famrel == 1 & famstruc1 == 0): mean gotstampfam, over(year) /*single generation*/
```

```
/*t-test in each year*/  
forvalues i = 1998(1)2013 {  
svy, subpop(if famrel == 1 & year == `i'): reg gotstampfam i.famstruc1  
}
```

```
*TABLE 5 (family level, 1998-2013): Average Number of Months that Families Received SNAP Benefits in the Last  
Calendar Year*  
use "finaldata.dta", clear  
set more off  
svyset psu [pweight=fweight], strata(strata)
```

```
/*Trend in the number of months receiving SNAP benefits*/  
svy, subpop(if famrel == 1 & famstruc1 == 1): mean stampmo_fam, over(year) /*multi-generation*/  
svy, subpop(if famrel == 1 & famstruc1 == 0): mean stampmo_fam, over(year) /*single generation*/
```

```
/*t-test in each year*/  
forvalues i = 1998(1)2013 {  
svy, subpop(if famrel == 1 & year == `i'): reg stampmo_fam famstruc1  
}
```