



Imputation of Height and Weight in the Mexican Health and Aging Study Version 3¹

Prepared By
Matthew Miller
Alejandra Michaels-Obregón
Rebeca Wong

Product of the MHAS Working Group

July 10, 2026

“The MHAS (Mexican Health and Aging Study) is partly sponsored by the National Institutes of Health/National Institute on Aging (grant number NIH R01AG018016) and the *Instituto Nacional de Estadística y Geografía* (INEGI) in Mexico. Data files and documentation are public use and available at www.MHASweb.org”

¹Note: Version 3 adds imputations from MHAS 2024.

Suggested citation for this document:

MHAS (2024), “Imputation of Height and Weight in the Mexican Health and Aging Study. Version 3, July 2026.” Retrieved from www.MHASweb.org on (date).

Table of Contents

I.	Abstract.....	3
II.	Introduction	3
III.	Methods	4
	a) Preparing data for imputation	4
	i. Differences between self-reported and measured heights and weights.....	7
	ii. Differences in heights across waves	7
	iii. Total missingness	8
	b) Imputation Procedure	9
	c) Post-Imputation.....	11
	d) Final Datasets for End-Users	11
IV.	Results	12
V.	References	38
	Appendix A – STATA Program.....	39
	2001.....	39
	2003.....	49
	2012.....	61
	2015.....	75
	2018.....	100
	2021.....	112
	2024.....	123

I. Abstract

This document explains our rationale for imputing self-reports of height and weight in observations where such data are missing, and the procedure we followed to do so. To highlight the effect of imputation versus deletion of observations with missing data, we compared the distributions of height, weight, and BMI among three groups: cases where those data were observed (non-imputed cases), cases where those data were imputed (imputed cases), and all cases (non-imputed plus imputed cases). We also compared two groups—non-imputed cases and all cases—by fitting three linear regression models of log-transformed BMI—each with a separate independent variable—in each group. As the final product, we constructed a database containing the mean and standard deviation of height, weight, and BMI of each individual in each wave across 10 imputations, along with flag variables indicating whether height and weight were imputed. These variables are shared with users in an MHAS data file for each wave, from 2001 to 2024.

II. Introduction

Imputation seeks to replace missing data with plausible values of each incomplete variable. Plausible values are simulated by estimating relationships between imputed variables and those with no missing values. Imputation adds a layer of uncertainty to results derived from imputed data, as it is not definitively known that the missing values would equal the imputed values if they were observed. Therefore, it is recommended that multiple sets of imputed data are created using a process that involves a degree of randomness. Such a procedure is called “multiple imputation,” and it is used here to compute plausible body mass indices (BMI) for subjects whose heights and/or weights are missing in each wave of the Mexican Health and Aging Study (MHAS).

Multiple imputation typically assumes that missing data are “missing at random” (MAR), meaning that the likelihood that data are missing is independent of the missing values themselves, given the observed values. Although it is difficult to tell whether our data are MAR from the observed data alone, assuming as much is reasonable, considering how many variables contributed to our imputations (Rässler, Rubin, & Zell, 2012; van Buuren, Boshuizen, & Knook, 1999).

Imputation of BMI in our data requires imputation of multiple variables, including a mixture of continuous and discrete variables with a non-monotonic pattern of missingness. Therefore, we employed multivariate imputation using chained equations (MICE) or fully conditional specification (FCS) algorithm because it provides the flexibility that we need (van Buuren, 2007).

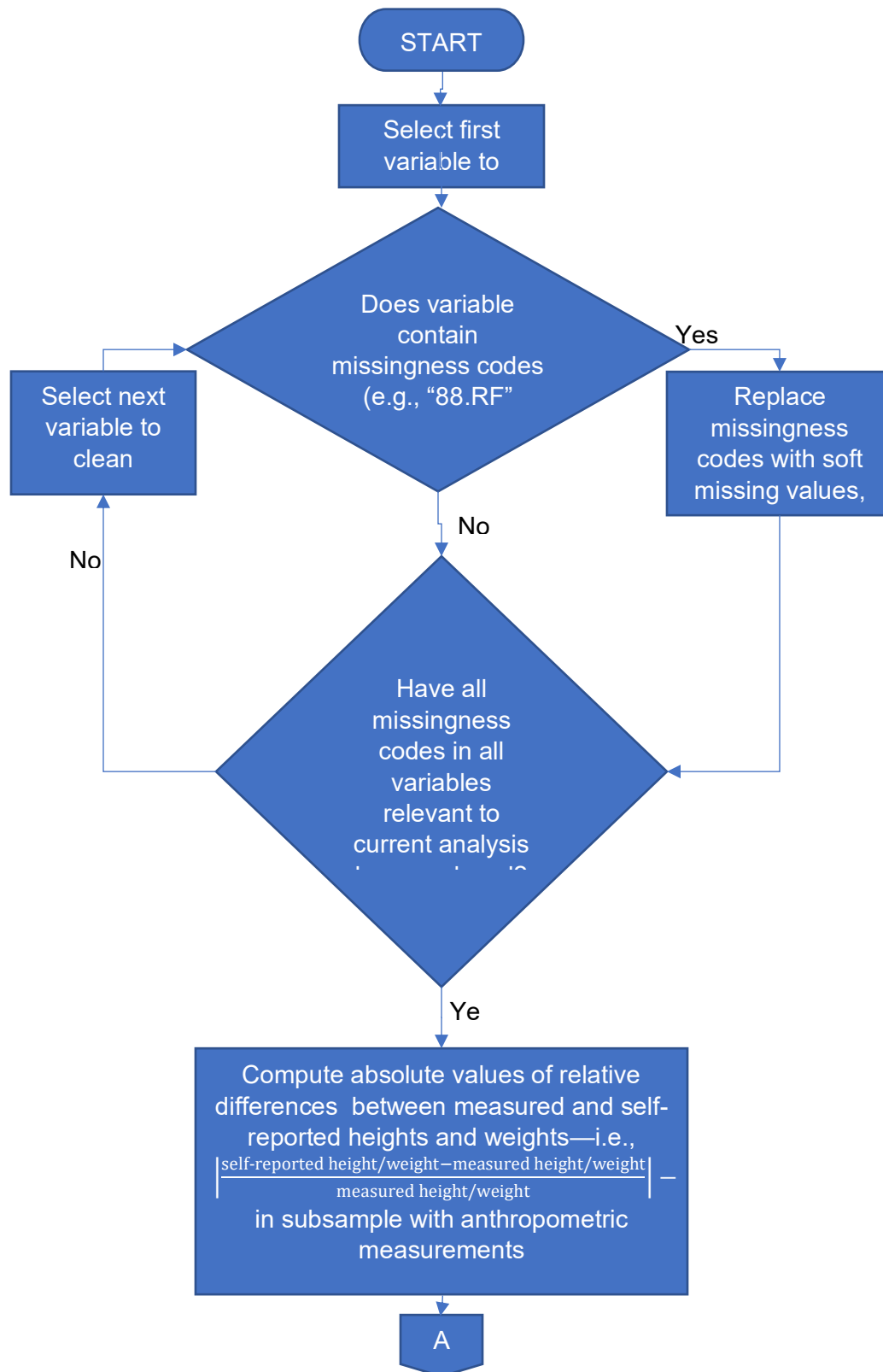
III. Methods

a) Preparing data for imputation

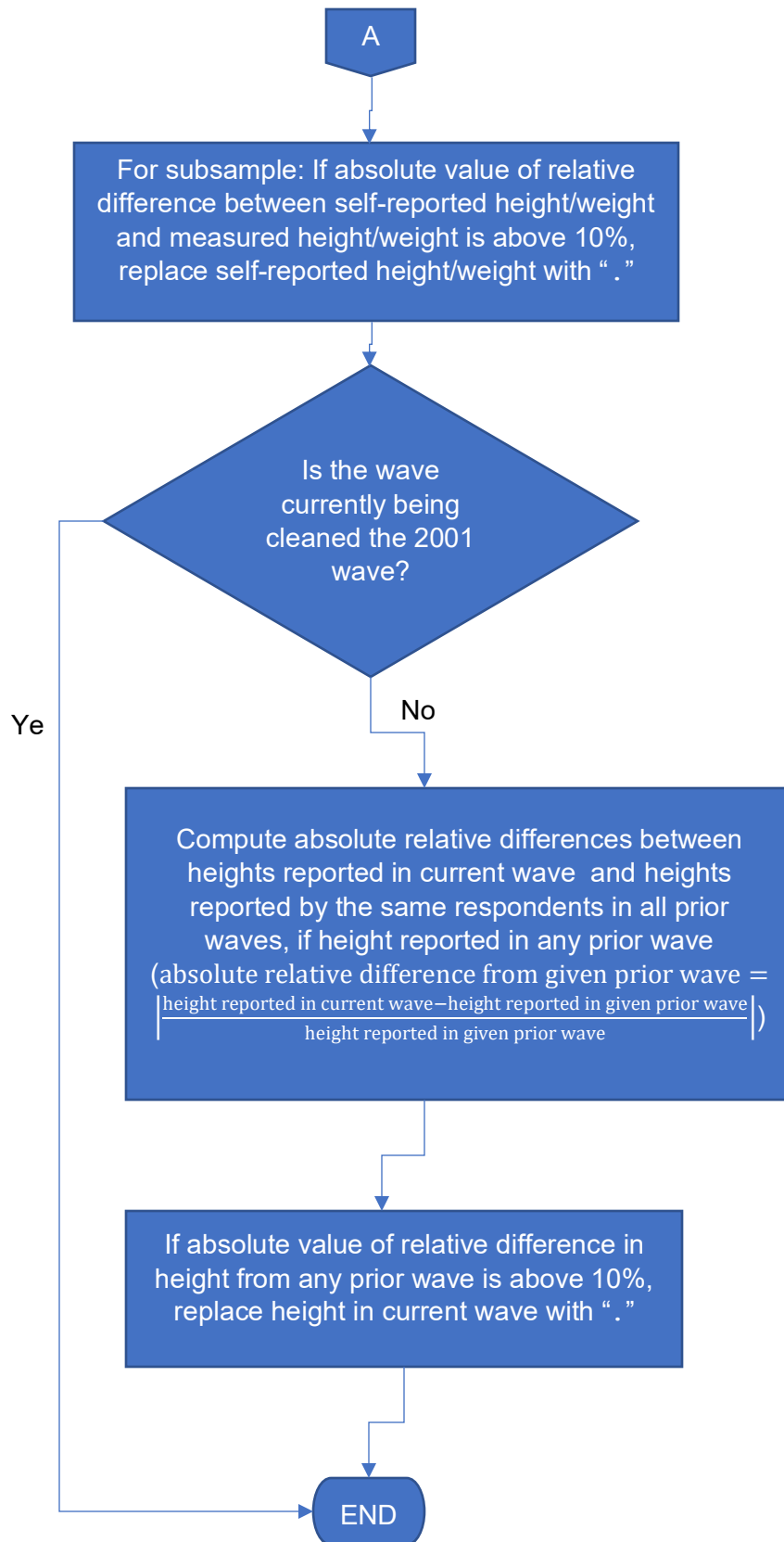
The variables that we seek to impute are self-reported height and weight. These are needed to construct imputed BMI.

In the raw dataset, numeric variables contain values that, although they appear as real numbers, denote individuals for whom those variables were unobserved for some known reason—for instance, 888 to indicate refusal or 999 to indicate a response of “Don’t know.” Stata, the software used to perform all imputations and analyses described in this document, regards such values as observed and valid, and thus these values need to be replaced with explicit missing values to avoid biasing imputations or analytical results. The codebooks for each wave list such values for each variable (MHAS, 2001–2018). Stata has 27 different missing numeric values: ., .a, .b, ..., and .z. Because only “.” (so-called “soft” missing values) can be imputed, this was used to replace invalid values in every variable. The process for “cleaning” the data in this way is outlined in Figure 1.

Figure 1. Flowchart of Process of Cleaning Data in Each Wave



*Includes six .1's in self-reported weight in 2015 and seven .1's in self-reported height in 2015



i. Differences between self-reported and measured heights and weights

In each of the first four waves of MHAS, a subsample was selected to obtain objective anthropometric measurements, including height and weight. These height and weight measurements contributed to the imputation of self-reported heights and weights for those observations selected for the subsample in each wave. As part of the anthropometric measurement protocol, height and weight were each measured twice in 2012 and 2015; thus, we computed each subject's average height and weight measurements in those waves and treated the result as a single measured value.

Some recorded values of self-reported heights and weights differed so greatly from measured values that the accuracy of the recorded self-reported value was suspect. Therefore, self-reported heights and weights that differed from observed measured values in the same survey participant by more than 10% of the measured value were replaced with soft missing values for the purpose of imputation. Table 1 lists the number of cases for which heights and weights were replaced in each wave.

Table 1. Numbers of self-reported heights and weights that differed from measured values by more than 10%

WAVE	2001	2003	2012	2015
TOTAL SAMPLE SIZE	15,186	13,704	15,723	14,779
SUBSAMPLE SIZE	2,944	2,641	2,086	2,054
CASES WITH DIFFERING HEIGHTS	43	53	67	70
CASES WITH DIFFERING WEIGHTS	317	263	252	270

ii. Differences in heights across waves

Furthermore, starting in 2003, if a self-reported height differed from the self-reported height by the same individual in at least one prior wave by more than 10% of the height in the prior wave, then the height in the later wave was also assumed to be inaccurate and replaced with a soft missing value. The number of such observations in each wave is displayed in Table 2. This is because heights in the target population of the MHAS should not change significantly over time.

Note that we did not impute weights that differed significantly from those reported in prior waves because we expect that a person's weight can change notably over time. We have identified one subject whose weight in 2021 we suspect was erroneously recorded.

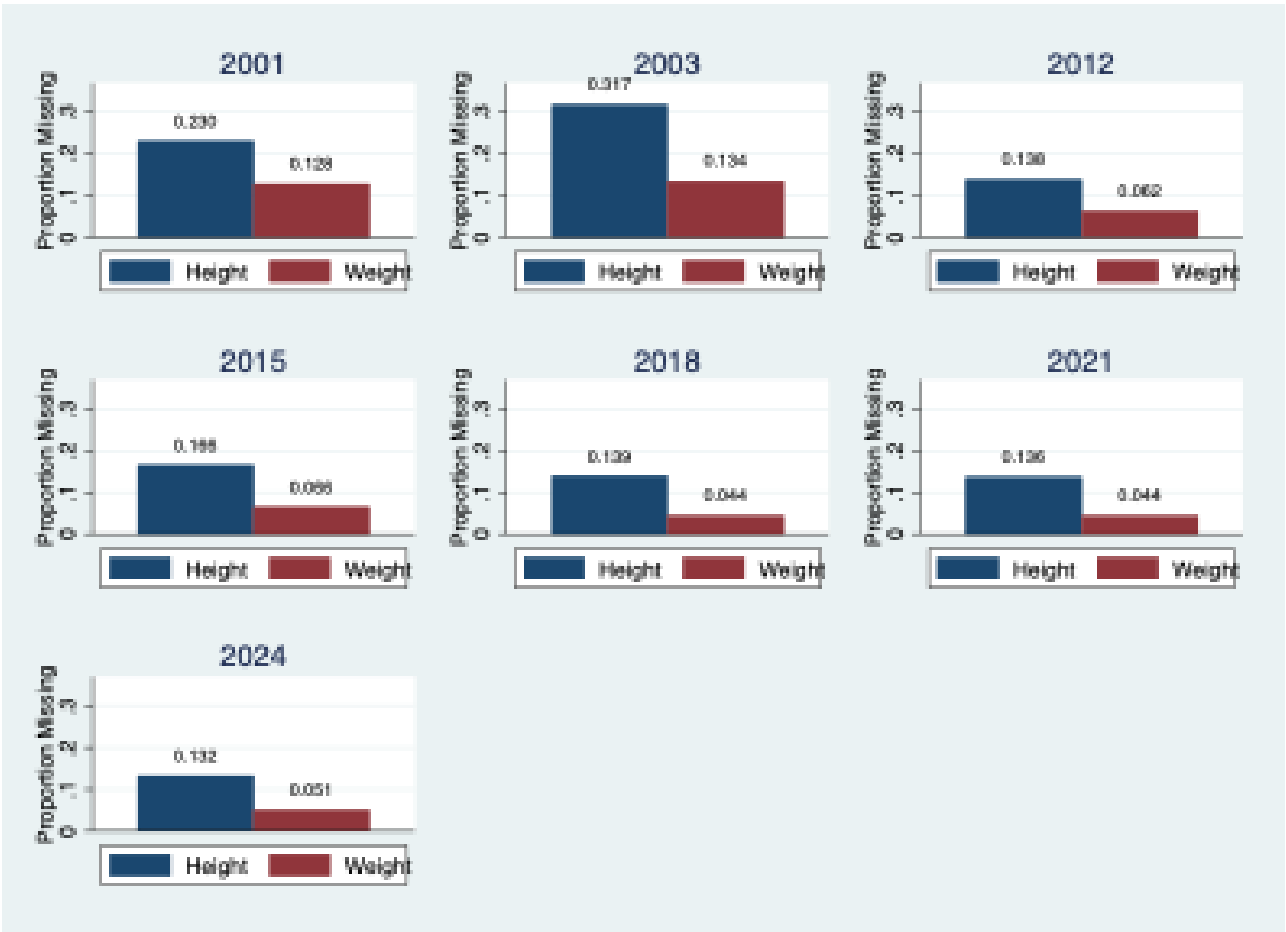
Table 2. Numbers of self-reported heights in each wave that differed from a height reported in at least one prior wave by more than 10%

WAVE	NUMBER OF DIFFERING HEIGHTS
2003	313
2012	477
2015	696
2018	831
2021	874
2024	1,203

iii. Total missingness

The proportion of observations in each wave with missing height and the proportion with missing weight after the data were prepared for imputation are shown in Figure 2.

Figure 2. Proportions of Individuals Missing Self-Reported Height and Self-Reported Weight in each Wave



b) Imputation Procedure

As previously mentioned, height, weight, and covariates thereof used in the imputation procedure were imputed with the MICE technique. MICE involves random draws from posterior predictive distributions, thus for the sake of reproducibility, we set the seed for pseudorandom-number generation to 101 each time that the command `mi impute chained` was called in Stata. The covariates for imputation of self-reported height and weight included sex, age, locality size, and years of education. MICE requires that missing values of one variable involved in the imputation of another variable also be imputed.

In addition to these covariates, measured heights and weights contributed to imputation of self-reported heights and weights for the subsamples selected for anthropometric measurements. Because measured heights and weights are expected to be missing outside of such subsamples, measurements were imputed conditionally upon selection for anthropometric measurements. When heights and weights reported in 2018 were imputed, anthropometric measurements from that wave were not available; thus, measured heights and weights did not contribute to imputation of height and weight in 2018.

Stata requires each conditionally imputed variable X to have exactly one unique non-missing value among all observations that do not meet the condition(s) under which X was imputed; this unique non-missing value is imputed for all missing values in such observations. However, in each wave measured heights and weights had missing values in all observations not selected for the subsamples. Therefore, in each wave the measured height and weight variables were assigned values of 0 in observations not selected for the subsample. 0 was chosen because the process that we used to impute all of our missing data involves linear regression, so replacing missing values with 0 should not prevent the artificially non-missing measured heights and weights from interfering with the imputation of missing values in other variables.

MICE works by sequentially performing a univariate imputation on each variable, with the method of imputing each variable chosen by the user depending on the nature of the variable. The process by which a univariate imputation method was chosen for each variable is outlined in Figure 3. For a given observation with a missing value for a given variable, MICE selects a value to impute by performing the univariate imputation method assigned for that variable for a certain number of iterations, specified by the user, known as the “burn-in” period. The burn-in period for each wave was selected by constructing trace plots of means and standard deviations of imputed values of each imputed variable and examining which iteration was not followed by a notable long-term trend in any trace plot.

Table 3 shows which variables were imputed in which waves and the univariate imputation method used to impute them. (Rows where the univariate imputation method is listed as “Complete” indicate that the variable in that row had no missing values and, thus, was not imputed in that wave.)

Table 3. Variables Imputed and Univariate Imputation Methods Employed

Variable	Wave						
	2001	2003	2012	2015	2018	2021	2024
Self-reported height	PMM						
Self-reported weight	PMM						
Self-reported body mass index (BMI)	Calculated from self-reported height and weight after imputation						
Measured height (subsample only) ^{1/}	PMM				Data not available		
Measured weight (subsample only) ^{1/}	PMM						
Years of education	PMM						
Age	Complete	PMM					
Locality size ^{2/}	Complete						
Gender	Complete						

Notes: “Complete” indicates that the variable had no missing values and, thus, was not imputed in that wave.

^{1/} Average of two measurements each for waves 2012 and 2015.

^{2/} 2003 data from 2001.

All variables with missing values were imputed using the predictive mean matching (PMM) univariate method—called “regression switching” by van Buuren, Boshuizen, and Knook (1999). In predictive mean matching, predicted values from Bayesian linear regression of the imputed variable on all other variables are used as a measure of distance between observations. Then for each observation with a missing value of the imputed variable, the algorithm finds a predetermined number of observations that are “closest” to the observation with a missing value according to this measure of distance among all observations with non-missing values of the imputed variable. One of those observations is selected at random, and the observed value from the selected observation is imputed for the missing value. Each imputed value was selected from one of the five closest observations with non-missing values. Table 4 includes miscellaneous information about imputed versus complete variables in each wave.

Table 4. Miscellaneous information about imputations in each wave

Wave	Imputed variables	Complete variables	Measured height and weight (subsample only)?	Burn-in period
2001	Height & weight, education	Sex, age, locality size	Yes (single measurement)	375 iterations
2003	Height & weight, age, education	Sex, locality size	Yes (single measurement)	700 "
2012	Height & weight, age, education	Sex, locality size	Yes (average of two measurements)	300 "
2015	Height & weight, age, education	Sex, locality size	Yes (average of two measurements)	400 "
2018	Height & weight, age, education	Sex, locality size	No	300 "
2021	Height & weight, age, education	Sex, locality size	No	125 "
2024	Height & weight, age, education	Sex, locality size	No	75 "

c) Post-Imputation

After imputation, BMI was generated as a passive variable, a function of one or more imputed variables, in each wave. In 2024 BMI was generated before imputation and recalculated using imputed values for observations where height and/or weight was missing. In all waves the final value of BMI was the median among 10 imputations.

To compare the results of imputation of missing values with those of listwise deletion, three linear regression models of the natural logarithm of BMI were constructed in each wave using both imputed and non-imputed data. Each model had only one independent variable: diabetic status, years of education, or locality size; and models were constructed on the same sets of variables using only non-imputed data. The models that included imputed data were pooled across 10 imputations, and the standard errors of estimated coefficients were adjusted to account for the added variability introduced by such pooling.

d) Final Datasets for End-Users

We collapsed the imputed dataset from each wave to the means and standard deviations of BMI, self-reported height, and self-reported weight for each individual across 10 imputations and uploaded the

collapsed dataset to the MHAS website. In cases where such values are observed, the means are the same as the observed values. The dataset includes the following variables:

- unhhid - Unique Household ID (=cunicah)
- cunicah - Clave Unica del Hogar (=unhhid)
- np - Person Number/ Numero de Persona
- codent01 (2001 only)
- acthog & codent03 (2003 only)
- subhog - Sub-household identifier
- tipent - Type of interview
- tam_loc (tamloc_01 for 2001 and 2003) - Locality size
- factori - Individual weight
- edad/age - Age
- sexo/sex - Sex
- ht_imp - Imputed/self-reported height without shoes (cm)
- wt_imp - Imputed/self-reported weight (kg)
- bmi_imp - Imputed/self-reported BMI (kg/m²)
- ht_imp_sd - Imputed Height: Between-imputation standard deviation
- wt_imp_sd - Imputed Weight: Between-imputation standard
- bmi_imp_sd - Imputed BMI: Between-imputation standard deviation
- ht_flag - Imputed Height: Flag if imputed
- wt_flag - Imputed Weight: Flag if imputed

IV. Results

Height is captured in meters plus centimeters. Three observations had values of 9 in `c67_1_12`, the meter portion of self-reported height in 2012, but a value of 9 in `c67_2_12`, the centimeter portion. The 2012 Section C codebook lists 9 as an invalid value of `c67_1_12` but not of `c67_2_12`. The variable `c67_2_12` was changed to a missing value in these cases anyway because `c67_1_12` and `c67_2_12` should be either both observed or both missing in every case. The same was done for one observation in which a value of 9 was recorded for both `c67_1_15` and `c67_2_15`, which represent the same values as `c67_1_12` and `c67_2_12` but for the 2015 wave. Six observations in 2015 had values of `.i`, indicating incomplete interviews, in self-reported weight, and seven observations in 2015 had such values in self-

reported height. Because such data can be considered MAR, these values were also replaced with soft missing values.

Tables 5–7 show great similarity between summaries of the distributions of observed (non-missing) values for self-reported height and weight in each wave compared to the distributions of both observed and imputed values.

Table 5. Self-Reported Heights (cm) by Percentile in Non-Imputed Cases and by Imputation Status, 2001–2021

2001	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(<i>N</i> = 11,677)	(<i>N</i> = 3,509)	(<i>N</i> = 15,186)
1 st	135	144.7	139
5 th	146	148.5	147
10 th	150	150.1	150
25 th	155	153.2	154
50 th	160	156.4	160
75 th	168	161.7	166
90 th	173	166.1	172
95 th	177	168.2	175
99 th	185	171.2	183

2003	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(<i>N</i> = 9,278)	(<i>N</i> = 4,426)	(<i>N</i> = 13,704)
1 st	140	145.1	140
5 th	146	148.1	147
10 th	150	150.2	150
25 th	155	153	153.7
50 th	160	156.4	160
75 th	168	161.5	165
90 th	173	166.1	170
95 th	176	168.1	175
99 th	183	172	182

2012	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(<i>N</i> = 13,622)	(<i>N</i> = 2,101)	(<i>N</i> = 15,723)
1 st	140	144.7	140
5 th	145	147.9	145.8
10 th	149	149.7	149.4
25 th	153	152.4	153
50 th	160	155.5	160
75 th	167	160.5	165
90 th	172	165.4	171
95 th	175	167.6	175
99 th	182	171.3	180

2015	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(N = 12,386)	(N = 2,393)	(N = 14,779)
1 st	140	145	140
5 th	145	147.7	146
10 th	149	149.2	149
25 th	153	151.8	152.4
50 th	160	155	159
75 th	166	160.4	165
90 th	172	165.6	170
95 th	175	168	175
99 th	182	172.5	180
2018	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(N = 14,730)	(N = 2,384)	(N = 17,114)
1 st	140	144.9	140
5 th	145	147.6	146
10 th	150	149	149.4
25 th	153	151.7	153
50 th	160	155.2	160
75 th	167	160.1	165
90 th	172	165.6	172
95 th	176	168.1	175
99 th	183	172.8	182
2021	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(N = 13,597)	(N = 2,142)	(N = 15,739)
1 st	140	146	140
5 th	146	148.3	146
10 th	150	149.7	149.9
25 th	153	151.6	153
50 th	160	154.9	160
75 th	167	160.2	165
90 th	172	165.7	171
95 th	175	168.1	175
99 th	183	172.5	182
2024	Non-Imputed Cases	Imputed Cases*	All Cases*
Percentile	(N = 18,318)	(N = 2,772)	(N = 21,090)
1 st	140	146	140
5 th	146	148.3	147
10 th	150	149.5	150
25 th	153	152	153
50 th	160	155.4	160
75 th	167	160.6	166
90 th	173	165.7	172
95 th	177	168.5	176
99 th	185	173.5	185

*Imputed values for each individual averaged across 10 imputations

Table 6. Self-Reported Weights (kg) by Percentile in Non-Imputed Cases and by Imputation Status, 2001–2024

2001	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 13,225)	(N = 1,961)	(N = 15,186)
1 st	40	46.7	41.1
5 th	50	51.9	50
10 th	53	55	53
25 th	60	60.5	60
50 th	70	66.3	68.8
75 th	78	72.1	78
90 th	87	78	86
95 th	94	82.4	93
99 th	109	94.1	108
2003	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 11,765)	(N = 1,939)	(N = 13,704)
1 st	40	51	40
5 th	49	55.6	50
10 th	53	57.7	54
25 th	60	62.1	60
50 th	69	67.1	68.2
75 th	78	73.3	78
90 th	87	80.5	86
95 th	94	85.7	93
99 th	108	92.9	106
2012	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 14,746)	(N = 977)	(N = 15,723)
1 st	42	44.6	42
5 th	50	49.7	50
10 th	53	53.9	53
25 th	60	60.3	60
50 th	69	66.3	69
75 th	78	72.5	78
90 th	88	79	87
95 th	95	84.7	95
99 th	108	103.5	108
2015	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 13,807)	(N = 972)	(N = 14,779)
1 st	41	43.7	41
5 th	49	49.1	49
10 th	53	51.4	53
25 th	60	58.8	60
50 th	69	65.05	68
75 th	78	71.3	78
90 th	88	78.5	87
95 th	95	83.3	95
99 th	109	95.6	109

2018	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 16,361)	(N = 753)	(N = 17,114)
1 st	42	51.5	42
5 th	50	56.1	50
10 th	53	58.2	54
25 th	60	62.3	60.8
50 th	70	66.9	70
75 th	80	72	79
90 th	89	77	88
95 th	96	81	96
99 th	110	88.9	110
2021	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 15,050)	(N = 689)	(N = 15,739)
1 st	43	51.1	43
5 th	50	55.4	50
10 th	54	57.4	54
25 th	60	61.5	60
50 th	69	65.8	69
75 th	79	71.2	78
90 th	88	76.7	88
95 th	95	79.8	95
99 th	110	85.3	110
2024	Non-Imputed	Imputed Cases*	All Cases*
Percentile	Cases (N = 20,012)	(N = 1,078)	(N = 21,090)
1 st	43	52.6	43
5 th	50	56.7	50
10 th	54	59.1	55
25 th	62	63.2	62
50 th	70	68.3	70
75 th	80	73.6	80
90 th	90	79.1	90
95 th	98	82	98
99 th	114	90.2	112

*Imputed values for each individual averaged across 10 imputations.

Table 7. Summary Statistics of Self-Reported Heights and Weights by Imputation Status, 2001–2024

	Number of Missing Values (% of Sample)	Non-Imputed Cases		Imputed Cases*		All Cases*	
		Mean	S.D.	Mean	S.D.	Mean	S.D.
2001 (N = 15,186)							
Self-Reported Height (cm)	3,493 (23.0)	160.94	9.99	157.20	6.07	160.08	9.37
Self-Reported Weight (kg)	1,937 (12.8)	69.94	13.98	66.53	9.50	69.50	13.53
2003 (N = 13,704)							
Self-Reported Height (cm)	4,426 (32.3)	161.04	9.62	157.26	6.14	159.82	8.83
Self-Reported Weight (kg)	1,939 (14.1)	69.79	14.09	68.20	8.88	69.57	13.49

2012 (N = 15,723)							
Self-Reported Height (cm)	2,101 (13.4)	159.99	9.44	156.54	6.06	159.52	9.14
Self-Reported Weight (kg)	977 (6.2)	70.00	13.99	66.88	10.93	69.80	13.84
2015 (N = 14,779)							
Self-Reported Height (cm)	2,393 (16.2)	159.84	9.34	156.29	6.33	159.27	9.02
Self-Reported Weight (kg)	972 (6.6)	69.77	14.19	65.40	10.65	69.48	14.03
2018 (N = 17,114)							
Self-Reported Height (cm)	2,384 (13.9)	160.16	9.51	156.24	6.31	159.61	9.23
Self-Reported Weight (kg)	753 (4.4)	70.60	14.30	67.47	7.67	70.46	14.08
2021 (N = 15,739)							
Self-Reported Height (cm)	2,142 (13.6)	160.10	9.33	156.33	6.21	159.59	9.07
Self-Reported Weight (kg)	689 (4.4)	70.38	14.10	66.59	7.36	70.21	13.90
2024 (N = 21,090)							
Self-Reported Height (cm)	2,772 (13.1)	160.64	9.78	156.65	6.26	160.11	9.49
Self-Reported Weight (kg)	1,078 (5.1)	71.60	14.72	68.75	7.90	71.45	14.46

*Imputed values for each individual averaged across 10 imputations

Figures 4a–5f show similar results even after controlling for locality size or diabetic status. These are expected, as imputation of missing values should not distort the distribution of data used to perform imputation.

Figure 4a. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2001)

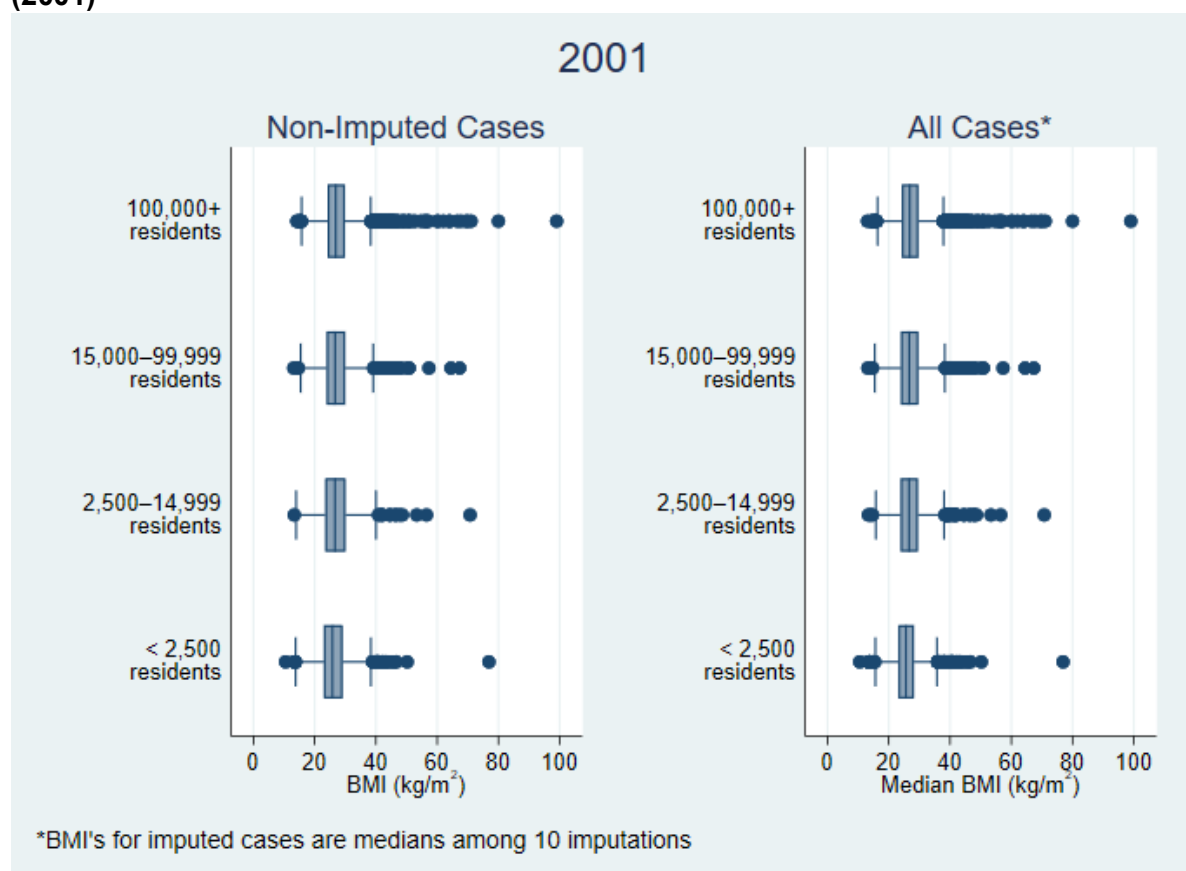


Figure 4b. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2003)



Figure 4c. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2012)



Figure 4d. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2015)

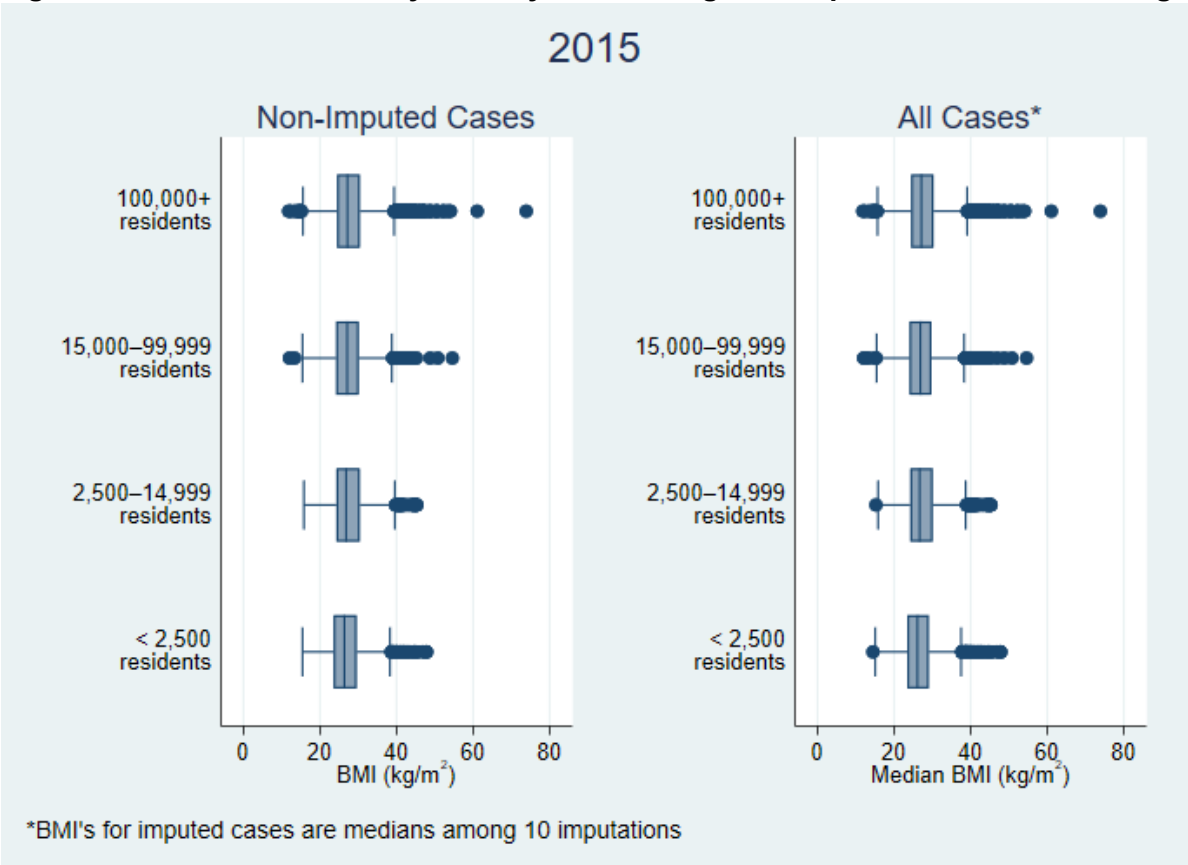


Figure 4e. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2018)

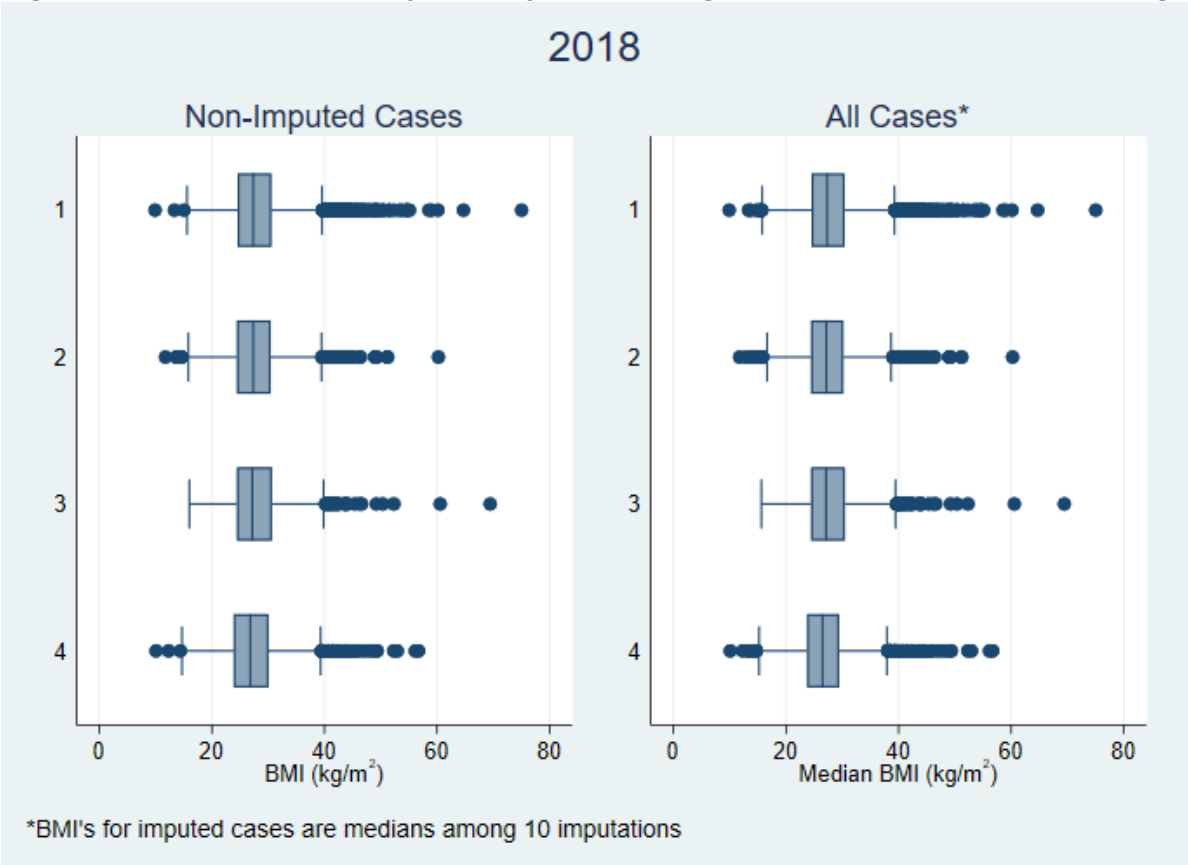


Figure 4f. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2021)

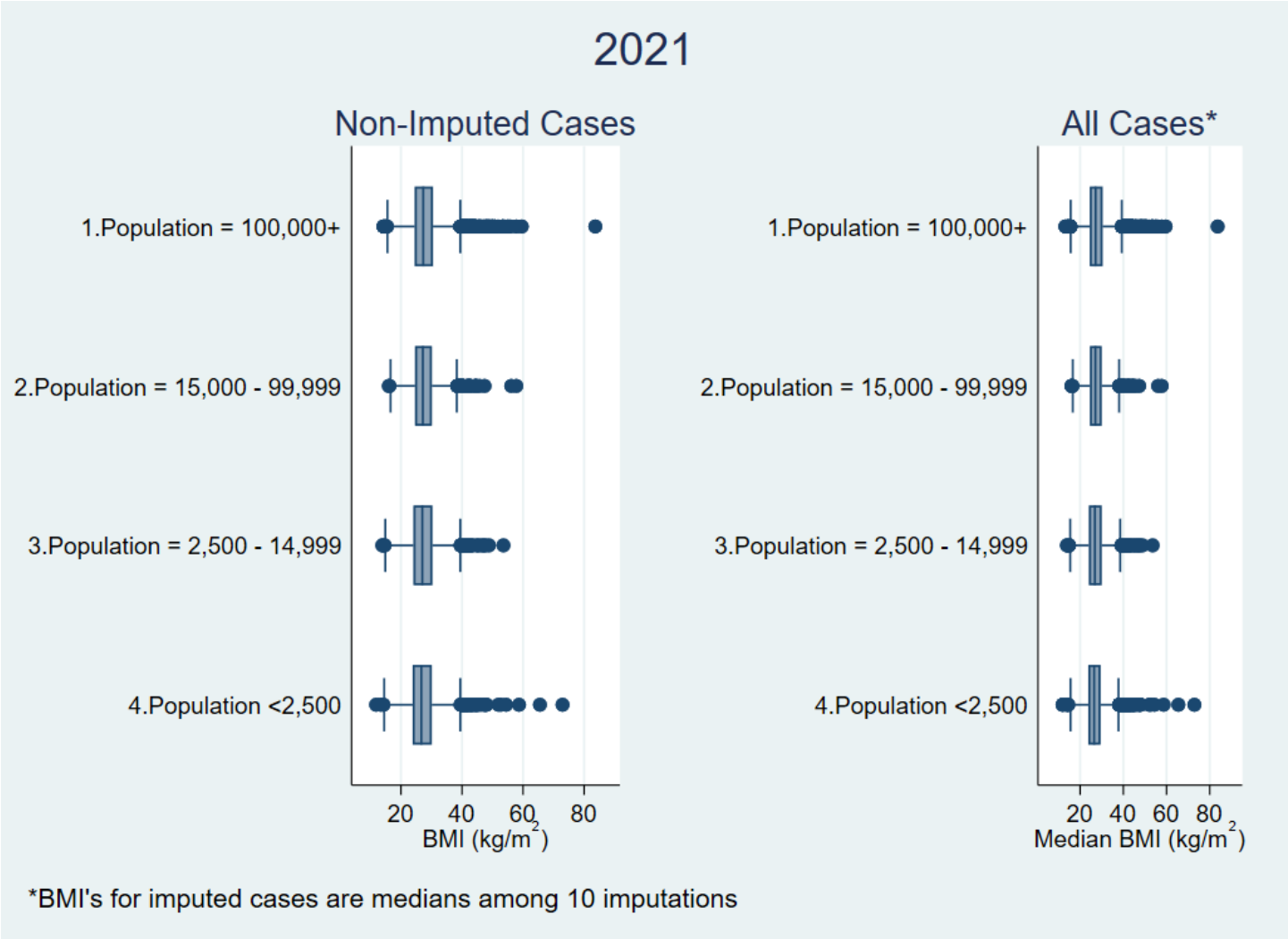


Figure 4g. Box Plots of BMI by Locality Size Among Non-Imputed Cases and Among All Cases (2024)

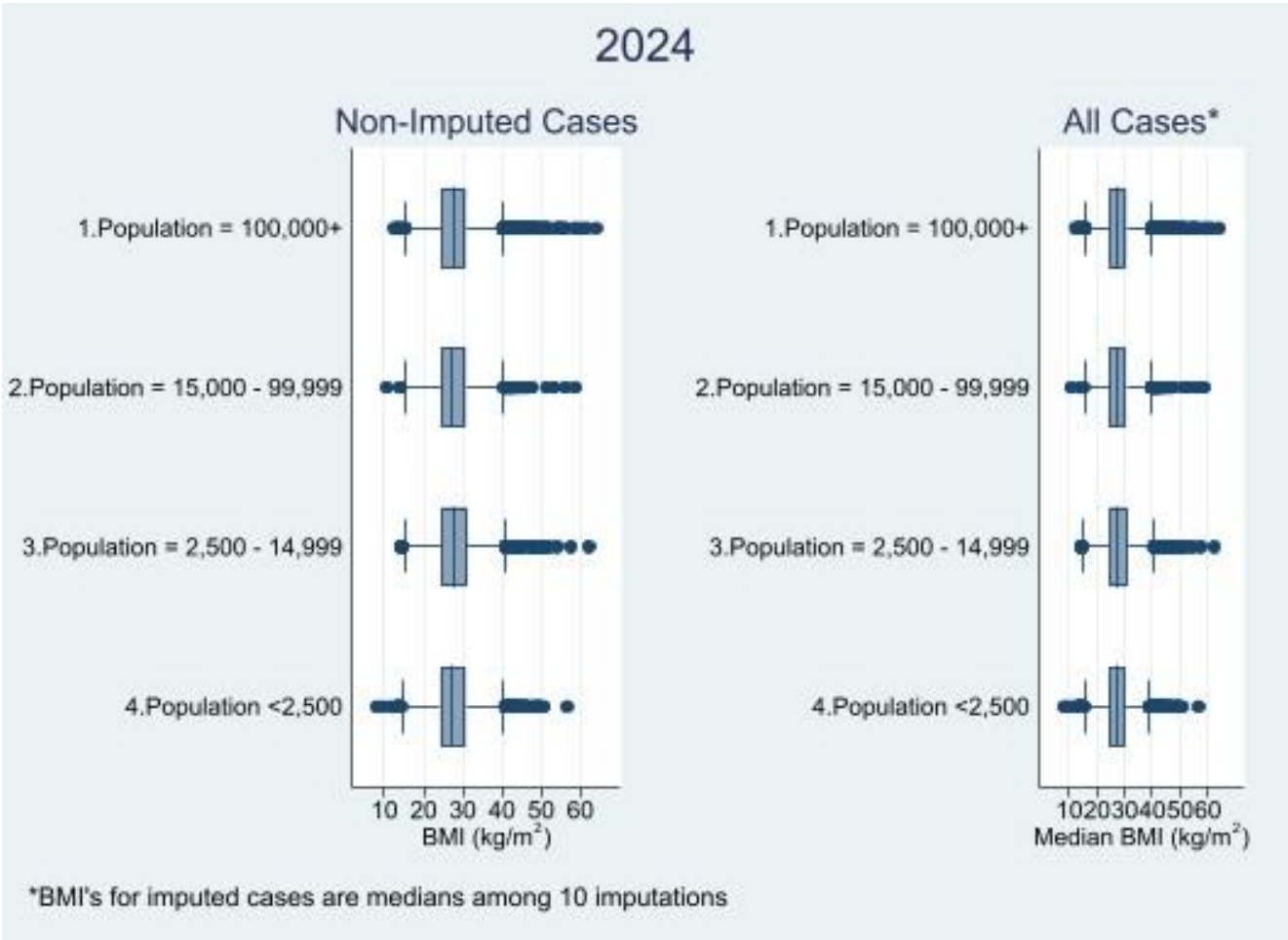


Figure 5a. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2001)

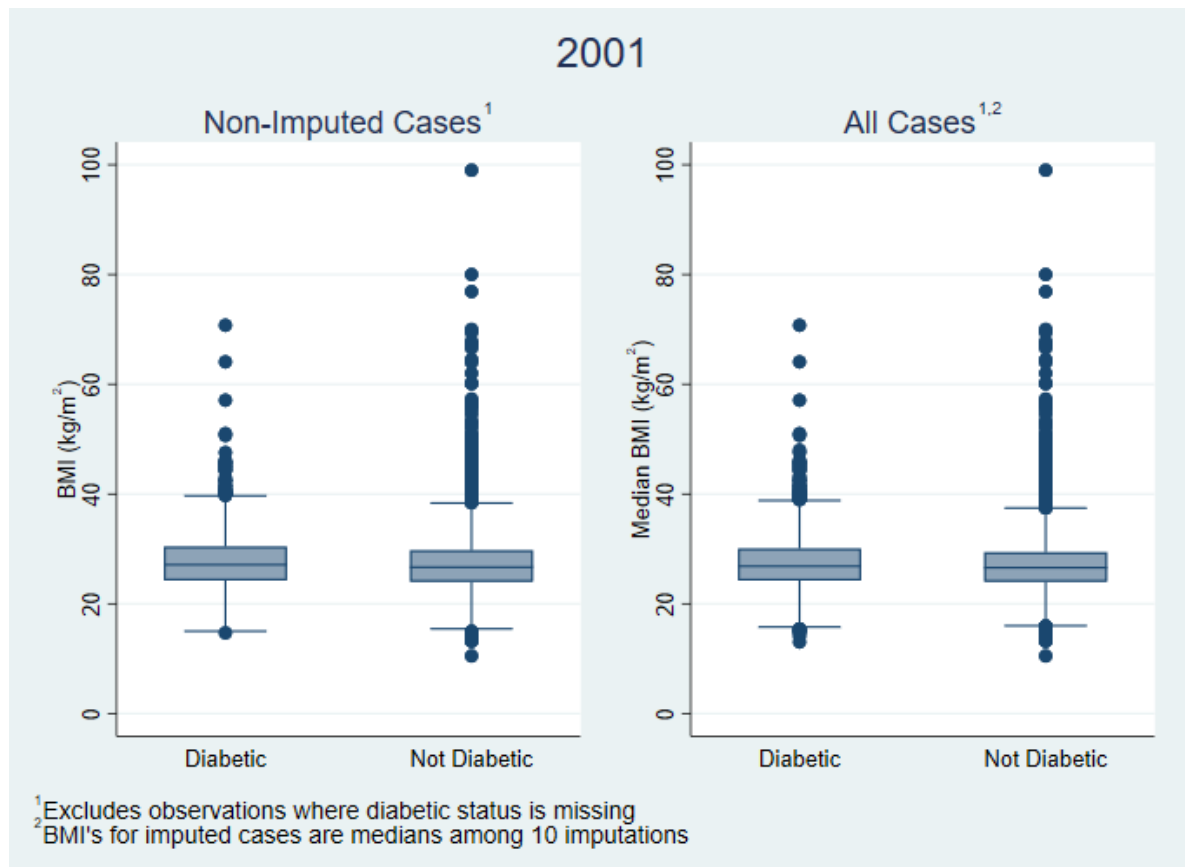


Figure 5b. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2003)



Figure 5c. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2012)

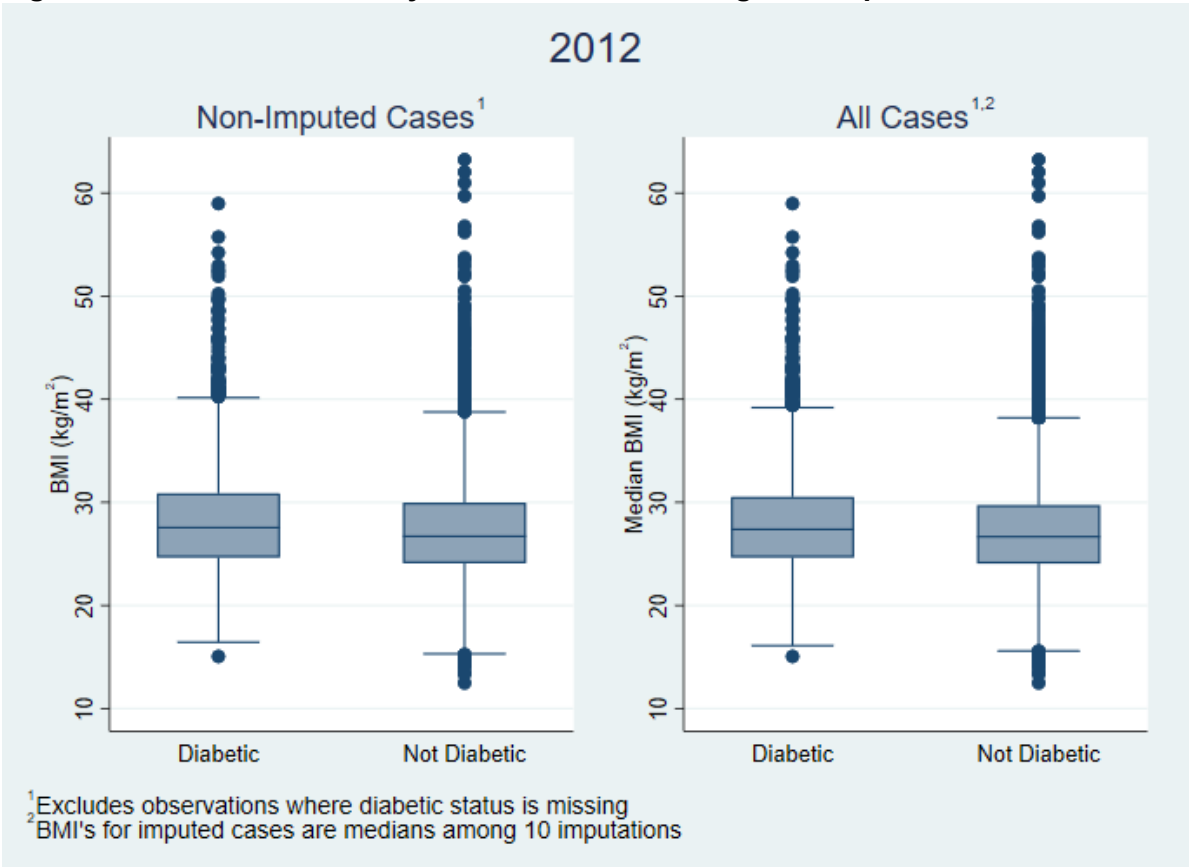


Figure 5d. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2015)

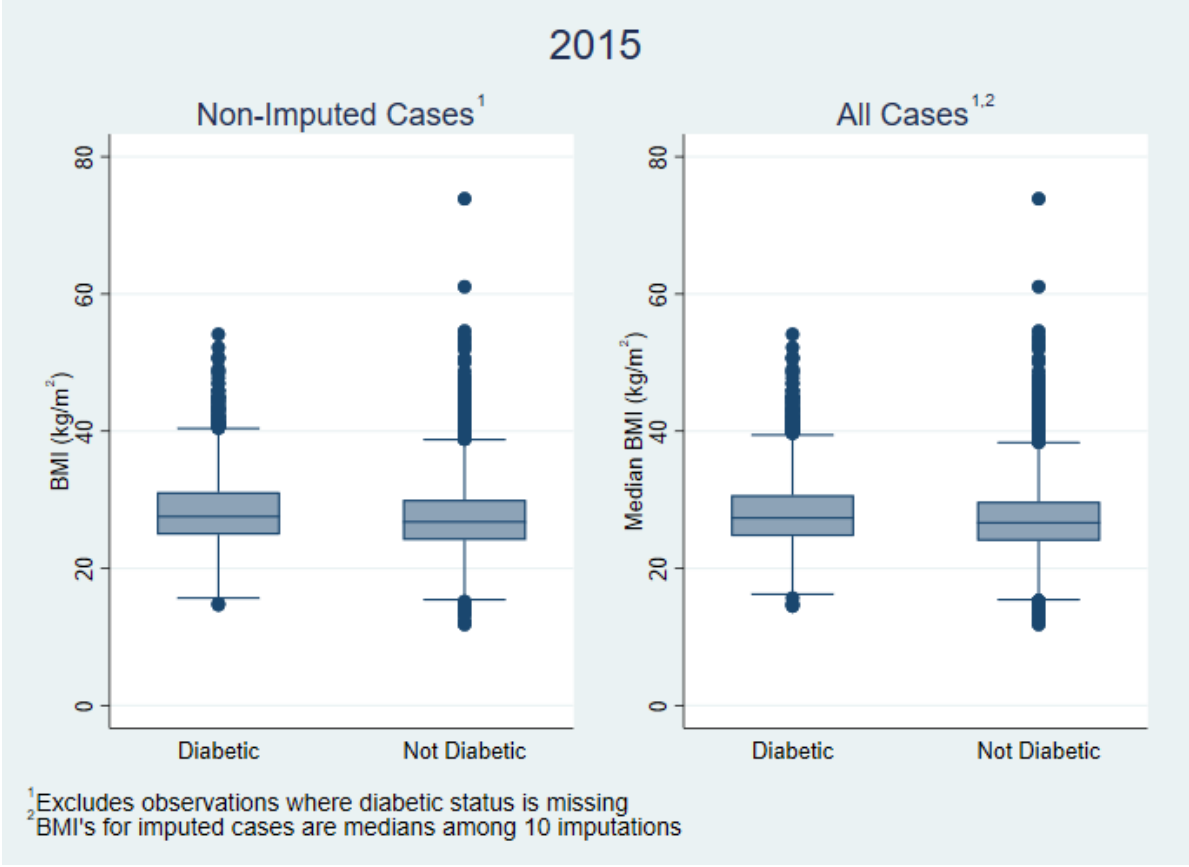


Figure 5e. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2018)

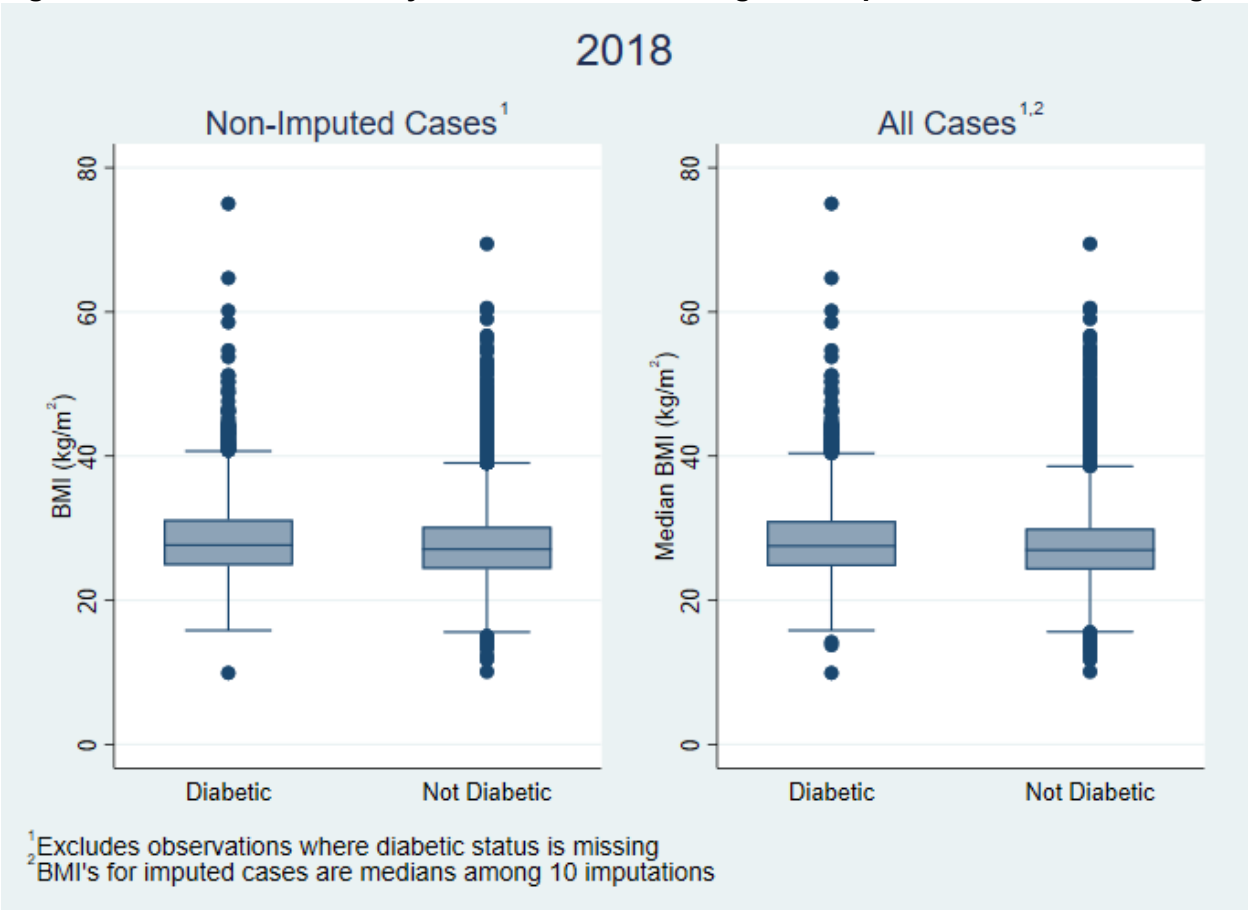


Figure 5f. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2021)

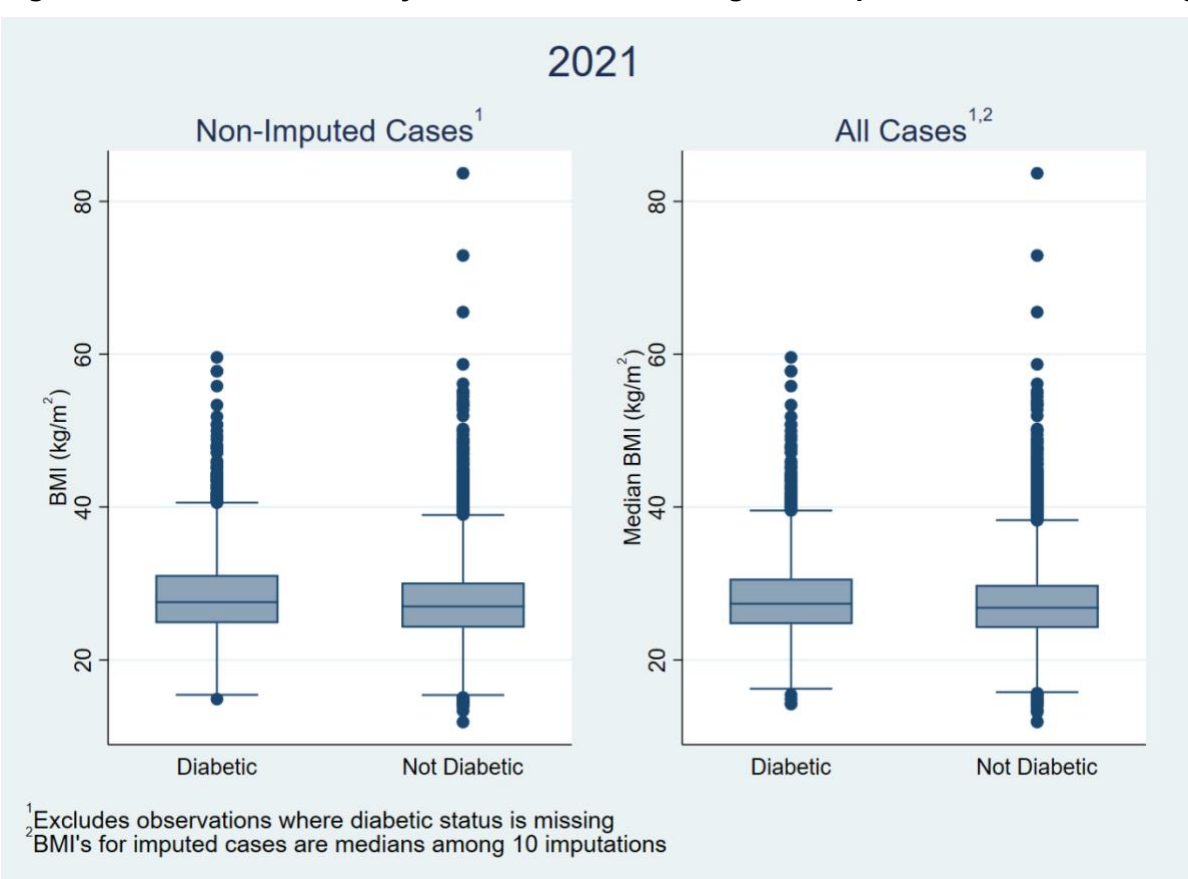
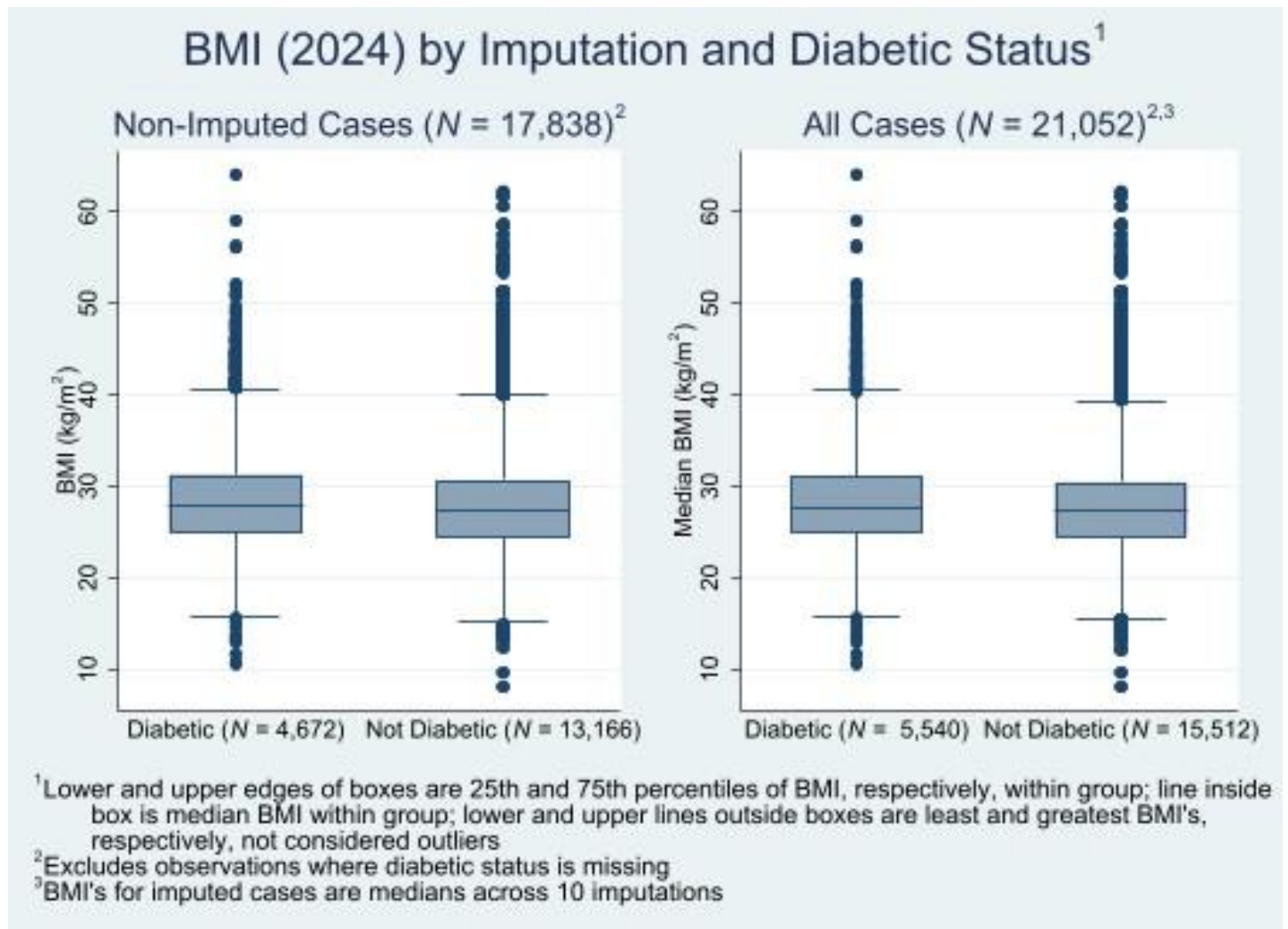


Figure 5g. Box Plots of BMI by Diabetic Status Among Non-Imputed Cases and Among All Cases (2024)



However, histograms of the two distributions (Figures 6a–6g) have visible differences. Said differences could be caused by averaging imputed values across imputations. Means are less variable than the data from which they are computed, thus it makes sense that histograms that include values averaged across 10 imputations show greater densities of values closer to the center of the distribution than the histograms that exclude averaged values.

Figure 6a. Histograms of Self-Reported Heights and Weights (2001)

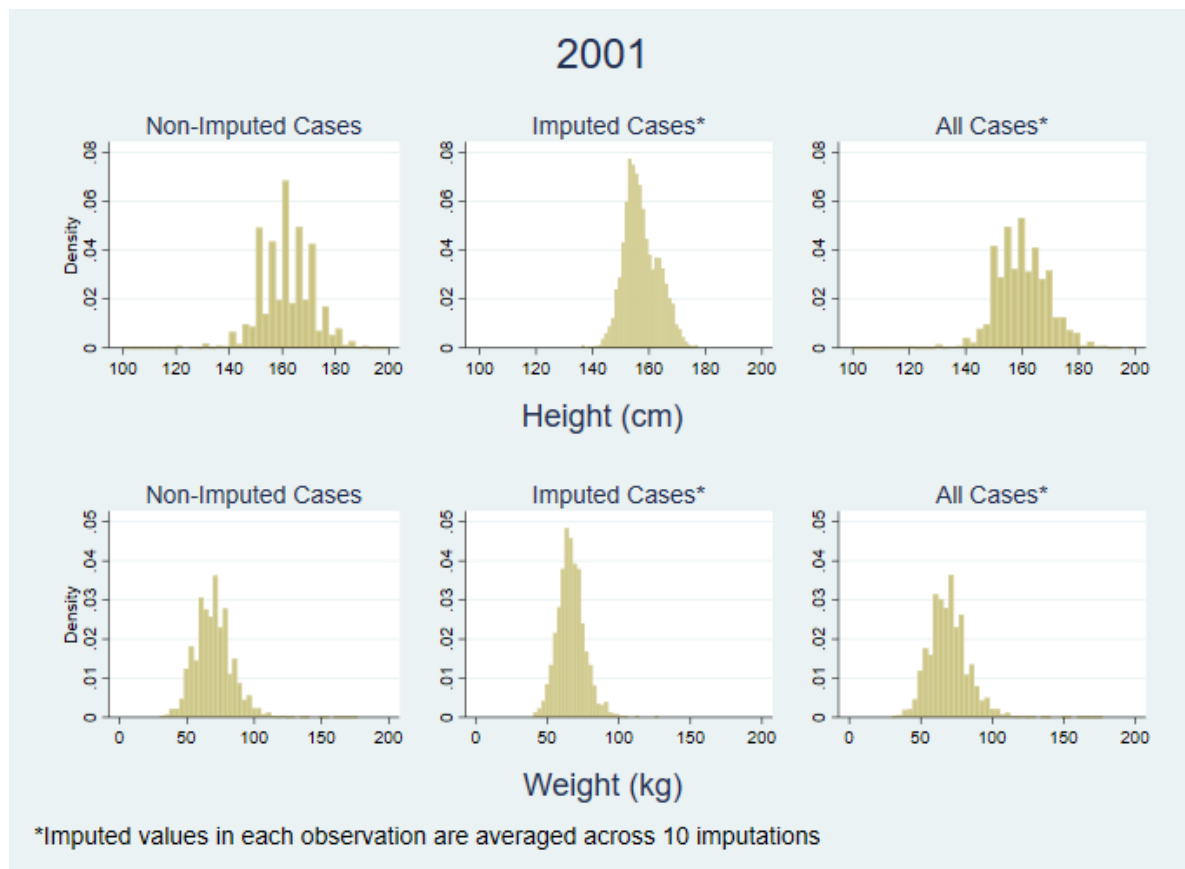


Figure 6b. Histograms of Self-Reported Heights and Weights (2003)

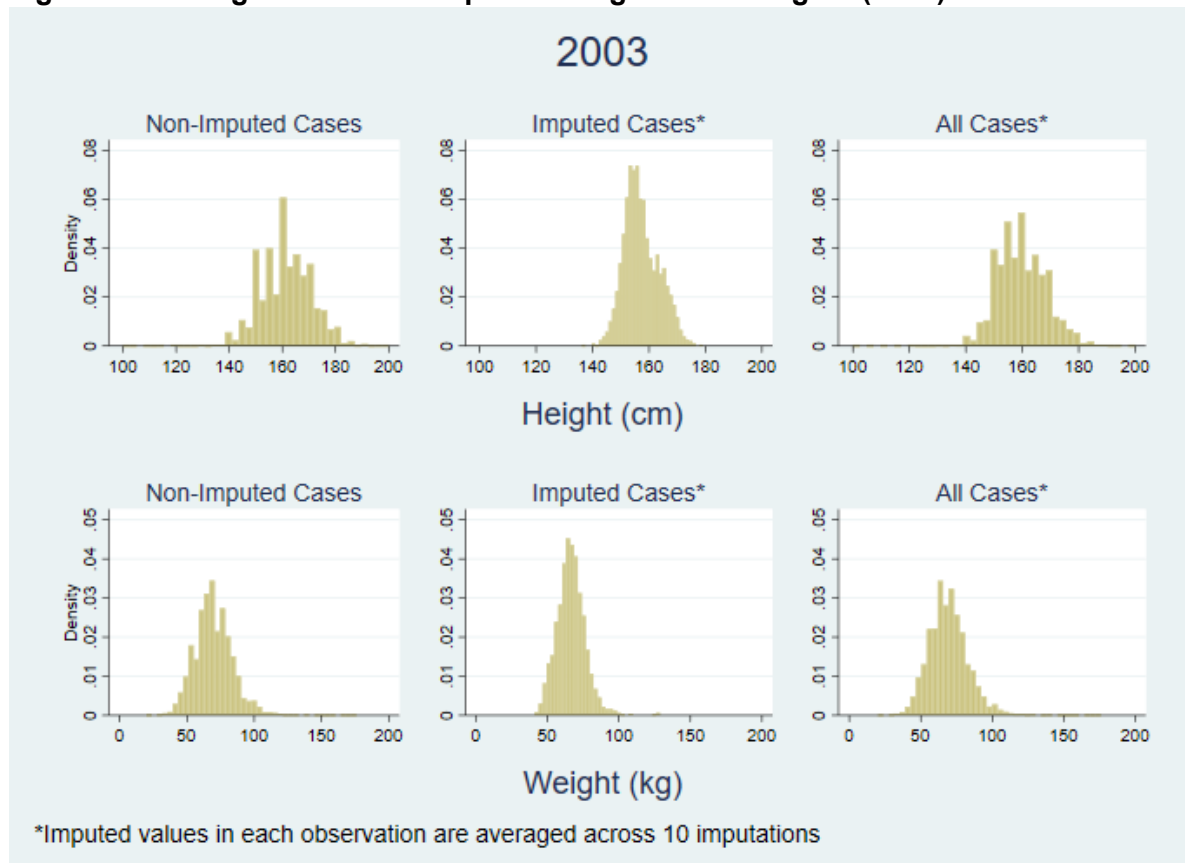


Figure 6c. Histograms of Self-Reported Heights and Weights (2012)

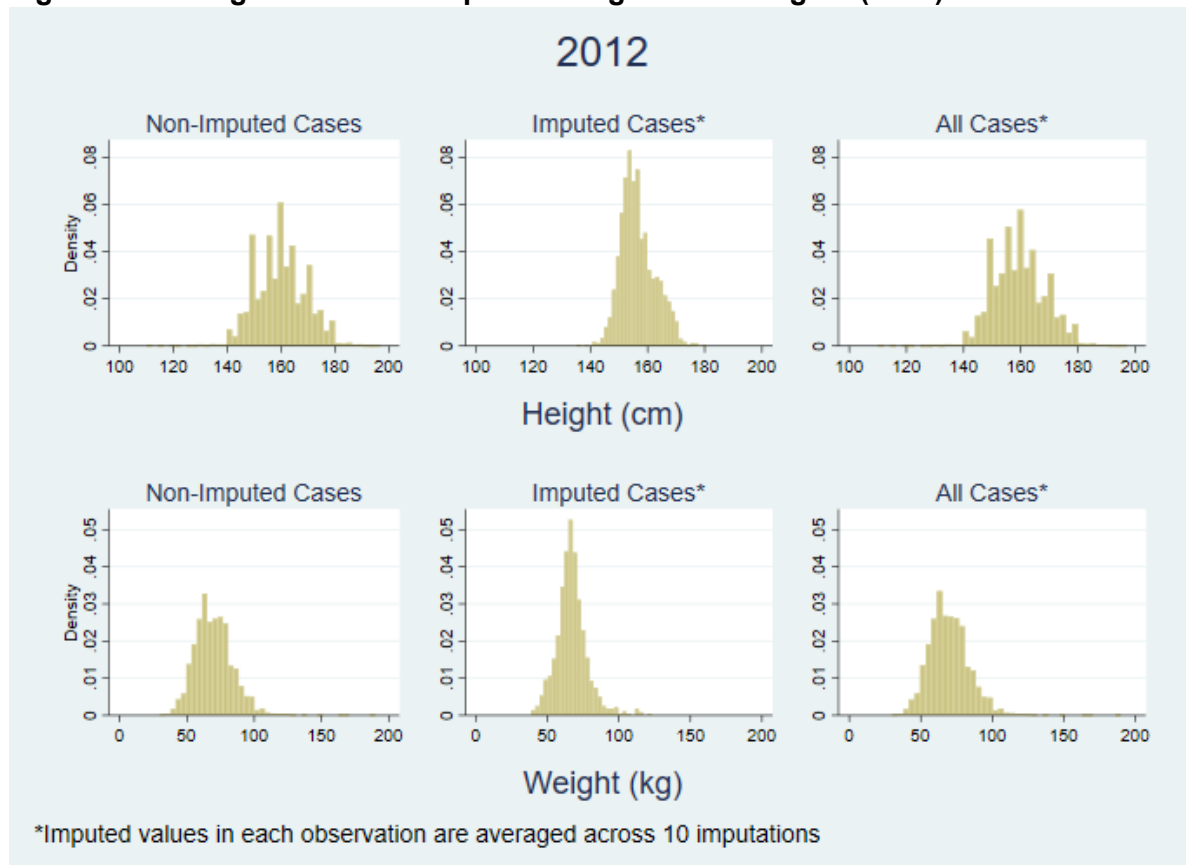


Figure 6d. Histograms of Self-Reported Heights and Weights (2015)

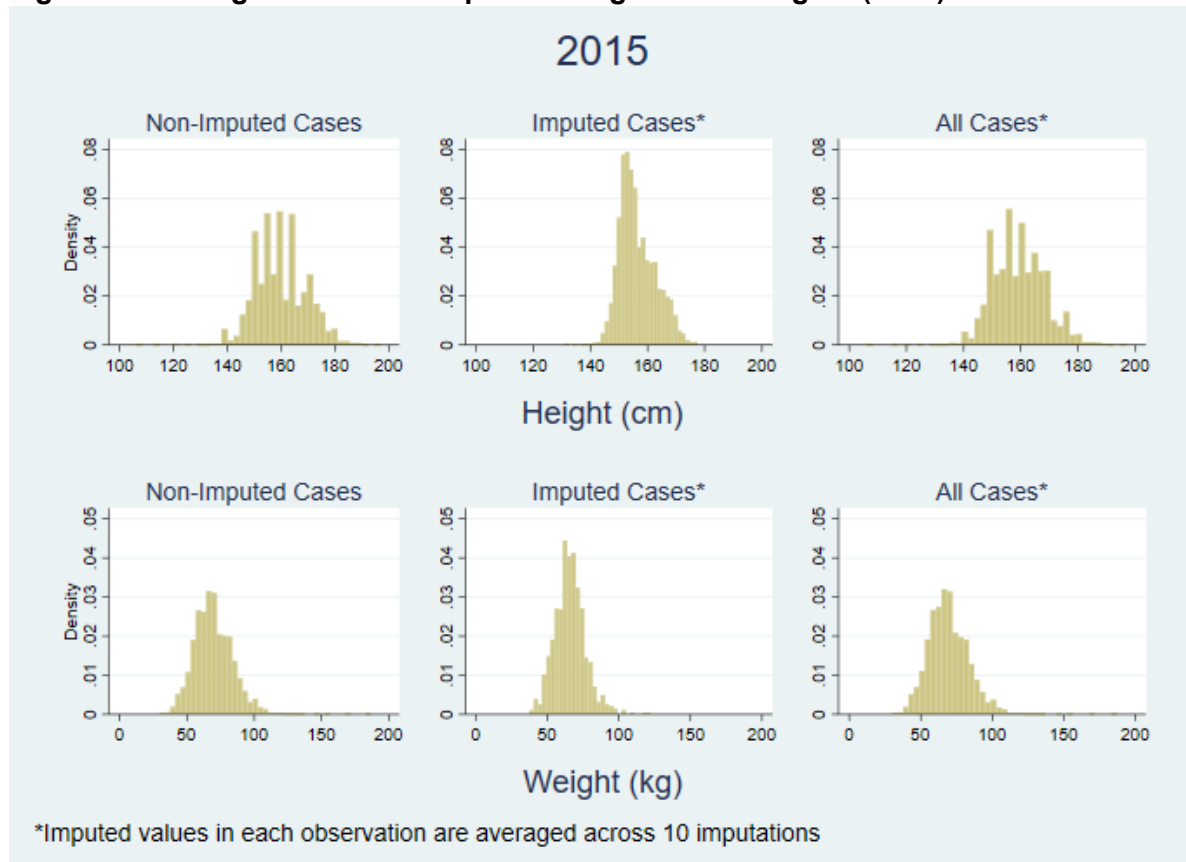


Figure 6e. Histograms of Self-Reported Heights and Weights (2018)

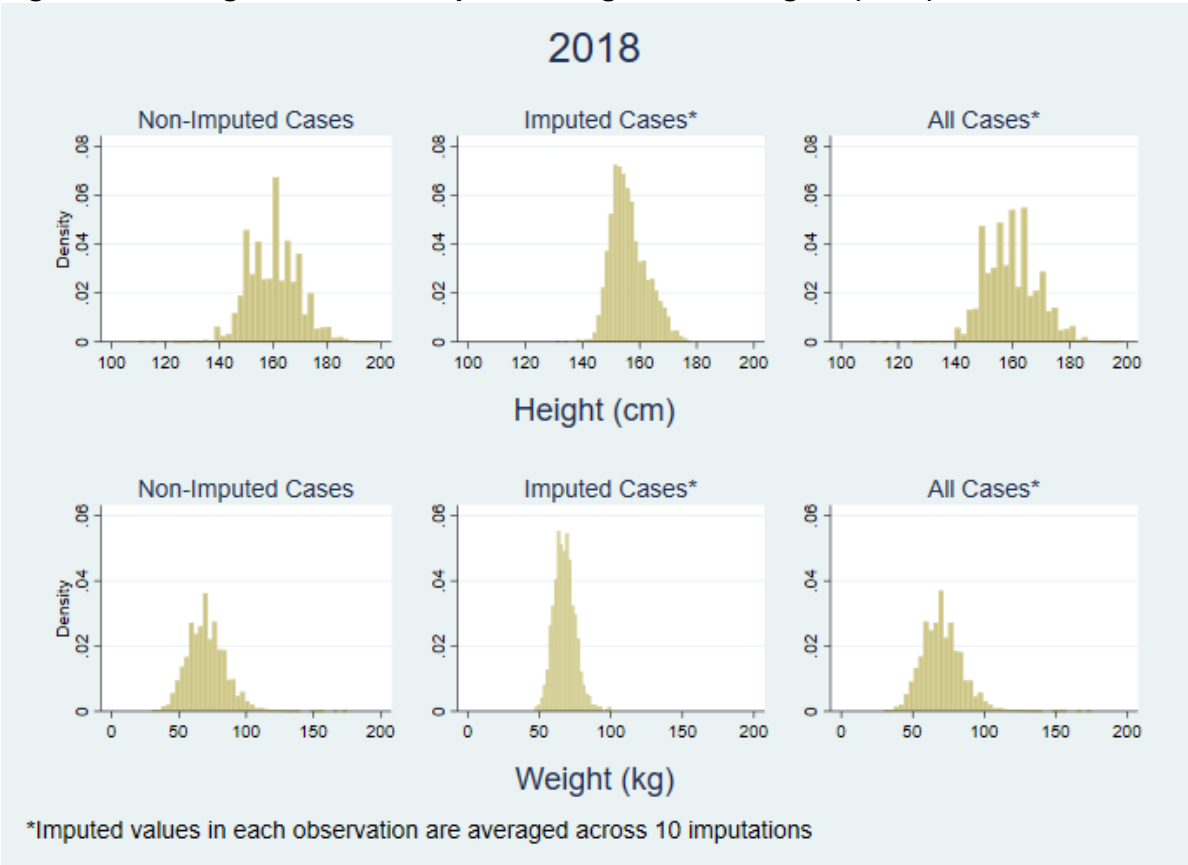


Figure 6f. Histograms of Self-Reported Heights and Weights (2021)

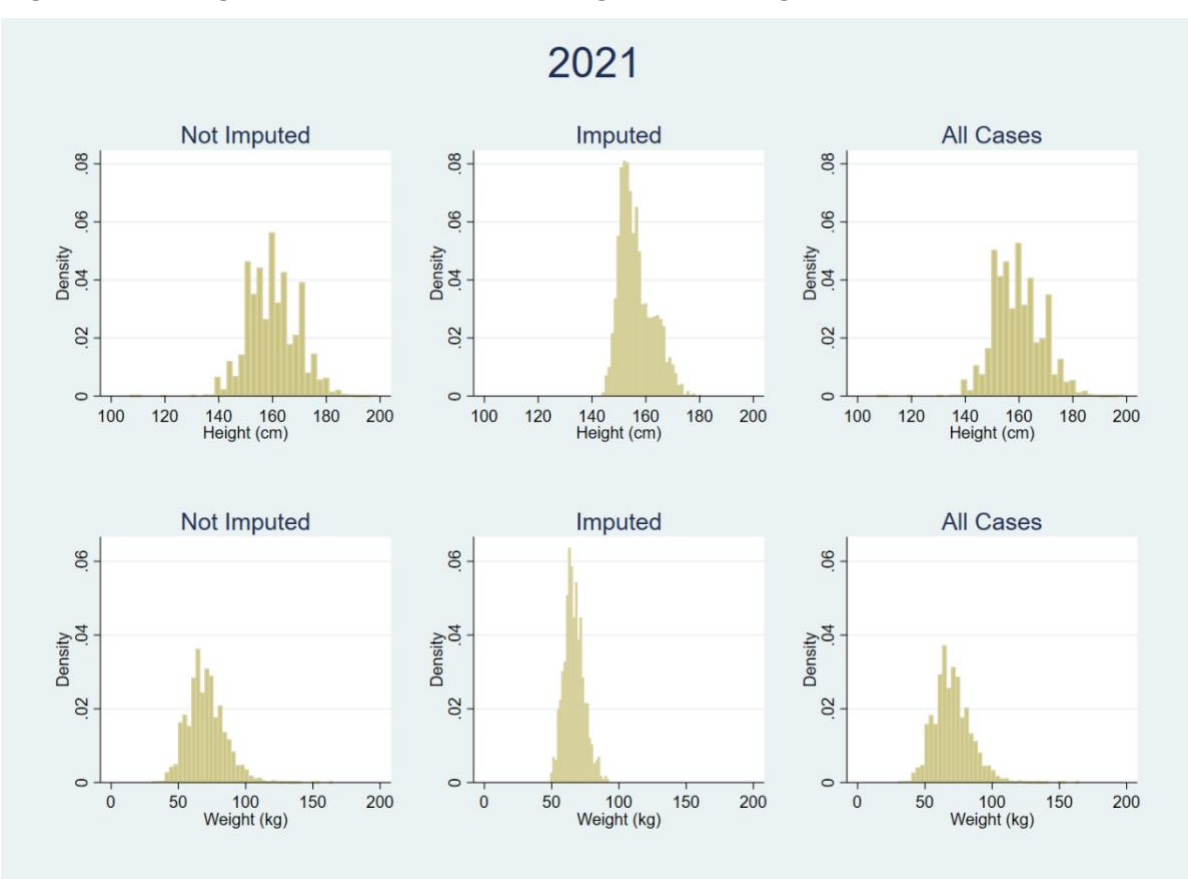
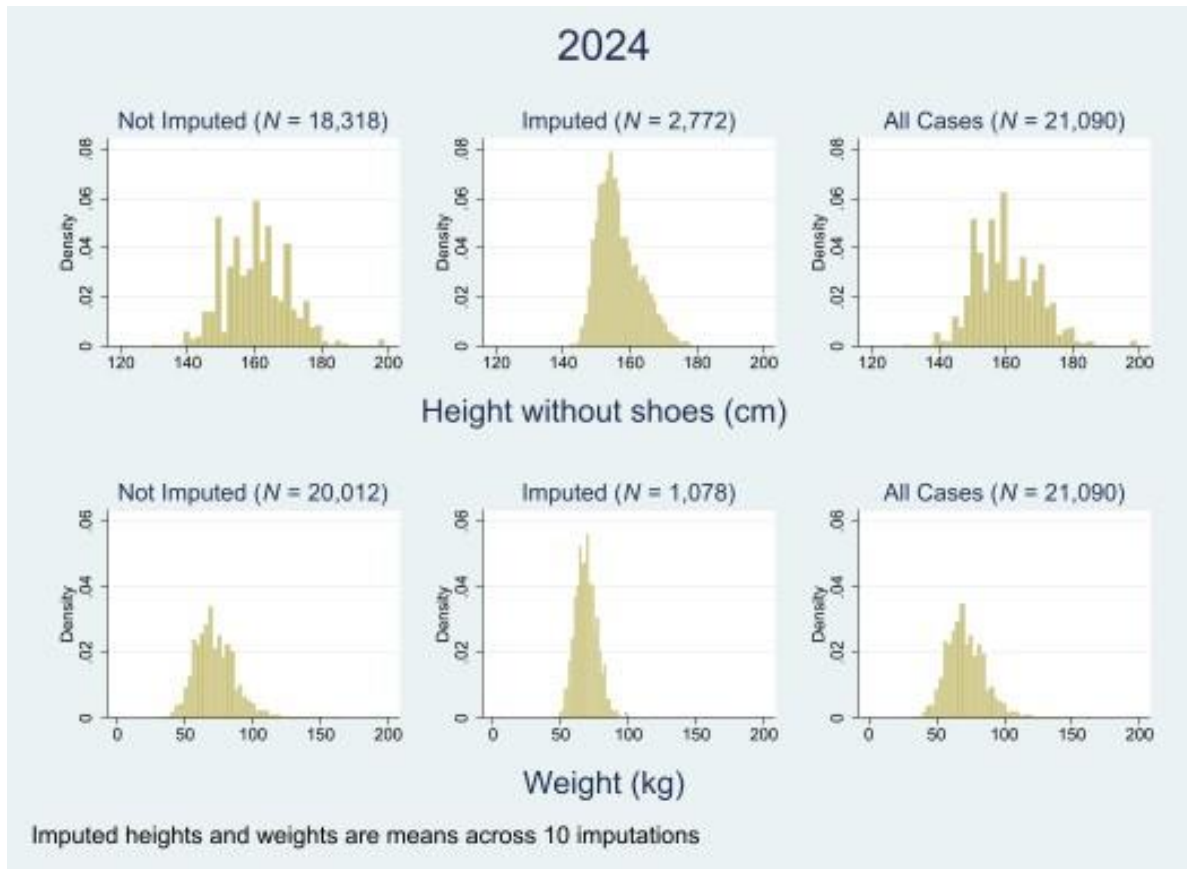


Figure 6g. Histograms of Self-Reported Heights and Weights (2024)



Next, we show how excluding observations with missing data outright can bias results. Table 8 presents the slopes of models of log-transformed BMI along with standard errors and p -values. For example, in 2001 and 2003, when the imputed data are included, we observe a statistically significant positive association between log-BMI and education. On the other hand, in the models that exclude the observations with missing values, this association is smaller in magnitude and not statistically significant. Also, in every wave the models with education or locality size as independent variables had smaller coefficients when observations with missing data are excluded than when their imputed values are included. In the models of log-BMI versus diabetic status, the regression coefficient is smaller when imputed values are included than when they are excluded. In summary, although there are differences in the regression coefficients in different waves, the fact that certain patterns were consistently evident across waves shows the potential impact that listwise deletion of observations with missing information has on analysis of these data.

Table 8. Marginal Associations Between Log-Transformed BMI on Years of Education, Locality Size*, and Diabetic Status

	Non-Imputed Cases Only				All Cases**			
	N^{***}	$\hat{\beta}$	s.e. ($\hat{\beta}$)	p-value	N^{***}	$\hat{\beta}$	s.e. ($\hat{\beta}$)	p-value
2001								
Years of Education	11,092	3.171×10^{-4}	3.64×10^{-4}	0.384	15,186	1.4988×10^{-3}	3.663×10^{-4}	****
Locality Size (2,500–14,999)		0.026	7.5639×10^{-3}	0.001		0.038	7.6339×10^{-3}	****
Locality Size (15,000–99,999)	11,102	0.034	6.4166×10^{-3}	****	15,186	0.043	6.4598×10^{-3}	****
Locality Size (100,000+)		0.044	5.1274×10^{-3}	****		0.052	4.8687×10^{-3}	****
Diabetes (Yes)	10,815	0.017	4.6903×10^{-3}	****	14,721	0.016	4.4853×10^{-3}	****
2003								
Years of Education	8,855	6.158×10^{-4}	4.031×10^{-4}	0.127	13,704	2.4013×10^{-3}	3.838×10^{-4}	****
Locality Size in 2001 (2,500–14,999)		0.030	5.6446×10^{-3}	****		0.042	7.8703×10^{-3}	****
Locality Size in 2001 (15,000–99,999)	8,897	0.041	7.02×10^{-3}	****	13,704	0.052	6.5326×10^{-3}	****
Locality Size in 2001 (100,000+)		0.044	8.5633×10^{-3}	****		0.062	4.876×10^{-3}	****
Diabetes (Yes)	8,840	0.024	4.9849×10^{-3}	****	13,648	0.023	4.5264×10^{-3}	****
2012								
Years of Education	13,172	2.05×10^{-3}	3.123×10^{-4}	****	15,723	2.6027×10^{-3}	3.154×10^{-4}	****
Locality Size (2,500–14,999)		0.025	5.8681×10^{-3}	****		0.028	5.6776×10^{-3}	****
Locality Size (15,000–99,999)	13,172	0.037	5.7513×10^{-3}	****	15,723	0.039	5.6581×10^{-3}	****
Locality Size (100,000+)		0.037	4.0076×10^{-3}	****		0.041	4.2003×10^{-3}	****
Diabetes (Yes)	13,149	0.034	3.5847×10^{-3}	****	15,689	0.032	3.3959×10^{-3}	****
2015								
Years of Education	11,802	2.415×10^{-3}	3.332×10^{-4}	****	14,779	3.6563×10^{-3}	3.338×10^{-4}	****
Locality Size (2,500–14,999)		0.026	6.524×10^{-3}	****		0.034	6.429×10^{-3}	****
Locality Size (15,000–99,999)	11,963	0.022	5.845×10^{-3}	****	14,779	0.029	5.659×10^{-3}	****
Locality Size (100,000+)		0.033	4.316×10^{-3}	****		0.034	4.113×10^{-3}	****
Diabetes (Yes)	11,952	0.034	3.678×10^{-3}	****	14,753	0.032	3.542×10^{-3}	****
2018								
Years of Education	14,281	1.5799×10^{-3}	2.986×10^{-4}	****	17,114	2.6326×10^{-3}	2.921×10^{-4}	****
Locality Size (2,500–14,999)		0.019	5.5811×10^{-3}	0.001		0.027	5.2266×10^{-3}	****
Locality Size (15,000–99,999)	14,424	0.018	5.2866×10^{-3}	0.001	17,114	0.026	5.0327×10^{-3}	****
Locality Size (100,000+)		0.022	3.8996×10^{-3}	****		0.030	3.6302×10^{-3}	****
Diabetes (Yes)	14,404	0.027	3.363×10^{-3}	****	17,084	0.026	3.542×10^{-3}	****

2021								
Years of Education	13,011	1.8258×10^{-3}	2.924×10^{-4}	****	15,739	2.6376×10^{-3}	3.008×10^{-4}	****
Locality Size (2,500–14,999)		7.4749×10^{-3}	5.6029×10^{-3}	0.182		0.015	5.308×10^{-3}	0.006
Locality Size (15,000–99,999)	13,304	0.016	5.687×10^{-3}	0.004	15,739	0.023	5.6692×10^{-3}	****
Locality Size (100,000+)		0.021	4.0304×10^{-3}	****		0.027	3.9955×10^{-3}	****
Diabetes (Yes)	13,293	0.027	3.4005×10^{-3}	****	15,724	0.025	3.3517×10^{-3}	****
2024								
Years of Education	17,611	2.1615×10^{-3}	2.775×10^{-4}	****	21,090	3.1298×10^{-3}	2.747×10^{-4}	****
Locality Size (2,500–14,999)		0.015	3.5158×10^{-3}	****		0.020	4.8174×10^{-3}	****
Locality Size (15,000–99,999)	17,866	8.8212×10^{-3}	4.836×10^{-3}	0.068	21,090	0.014	4.6104×10^{-3}	0.002
Locality Size (100,000+)		0.018	4.9901×10^{-3}	****		0.020	3.3641×10^{-3}	****
Diabetes (Yes)	17,838	0.022	3.0112×10^{-3}	****	21,052	0.020	2.9166×10^{-3}	****

*Versus locality size < 2,500

**Models pooled across 10 imputations

***Number of observations in which neither log-transformed BMI nor independent variable is missing

**** p -value appears as 0.000 in coefficients table

Table 6 and Figures 7a–7g show that—although exclusion of cases with missing values decreases the slopes of linear associations between log-transformed BMI and each of education, locality size, and diabetic status towards zero—this effect is less pronounced in the 2012– 2021 waves than in the 2001 and 2003 waves. This result may be because there were greater proportions of missing values in 2001 and 2003 than in later waves.

Figure 7a. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2001 Wave)

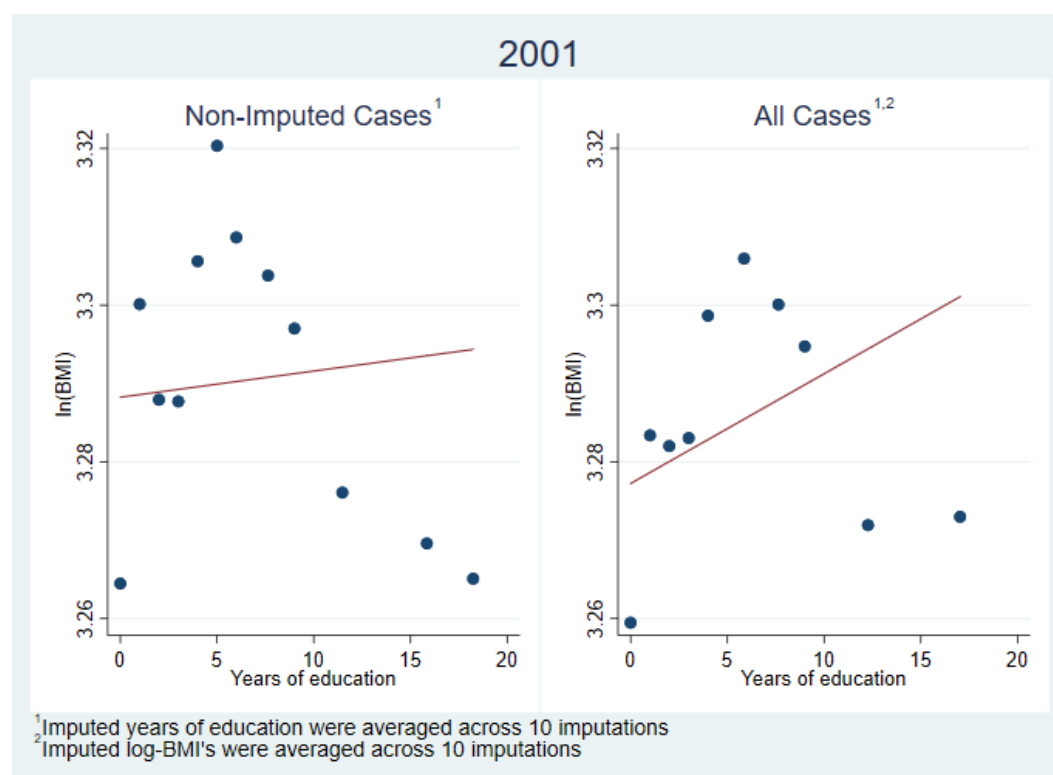


Figure 7b. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2003 Wave)

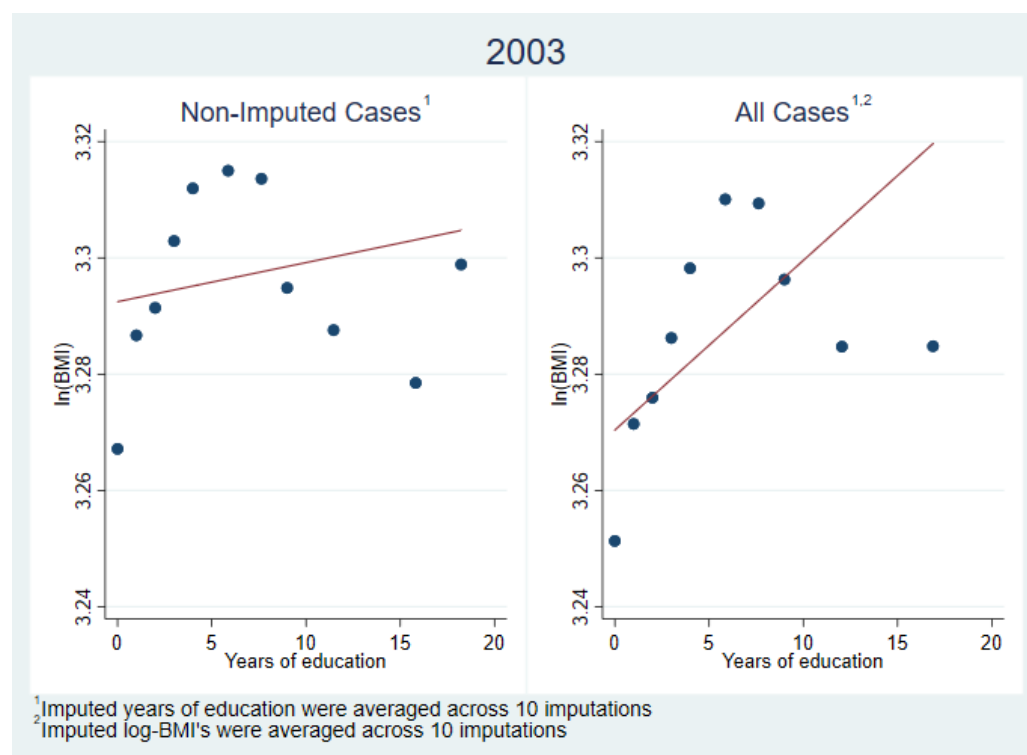


Figure 7c. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2012 Wave)

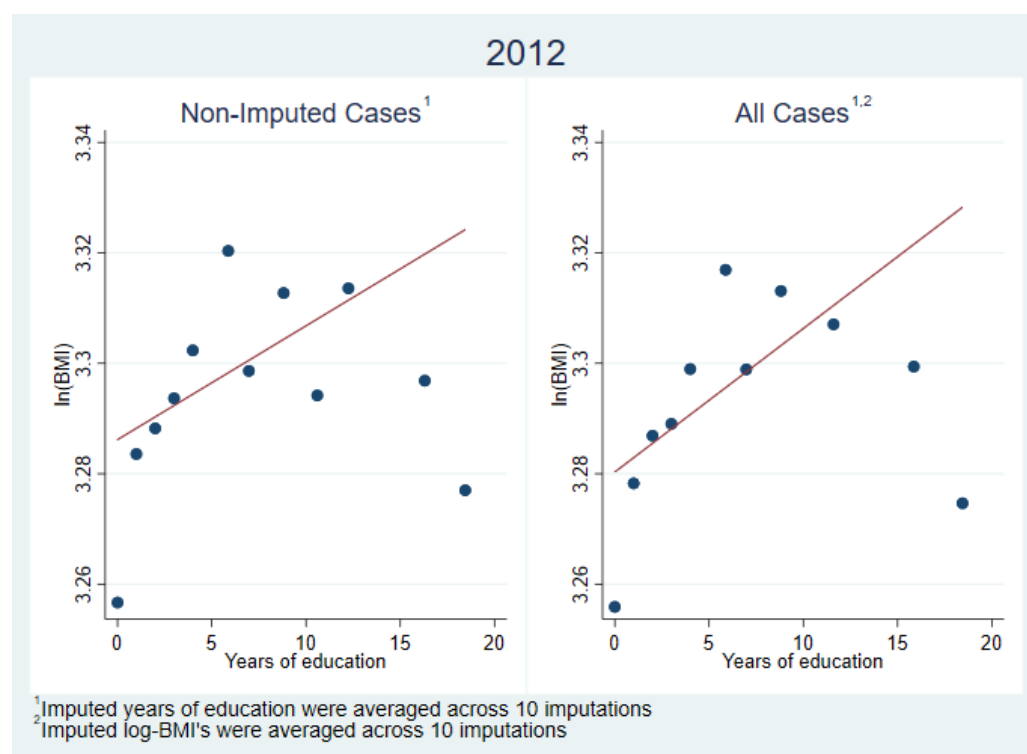


Figure 7d. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2015 Wave)

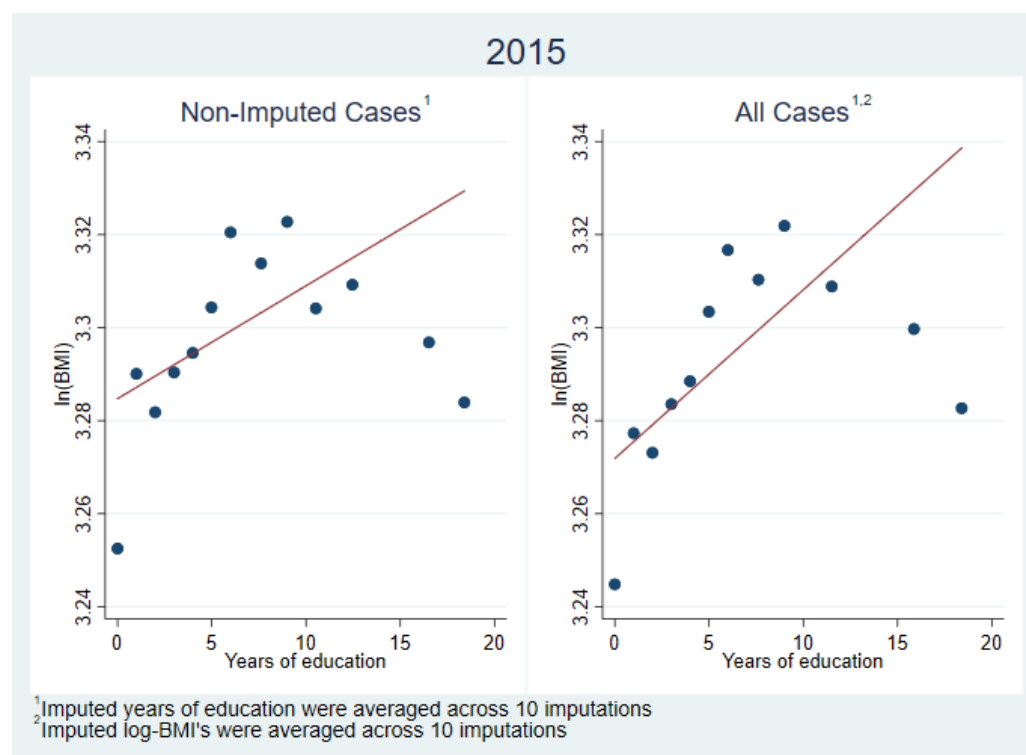


Figure 7e. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2018 Wave)

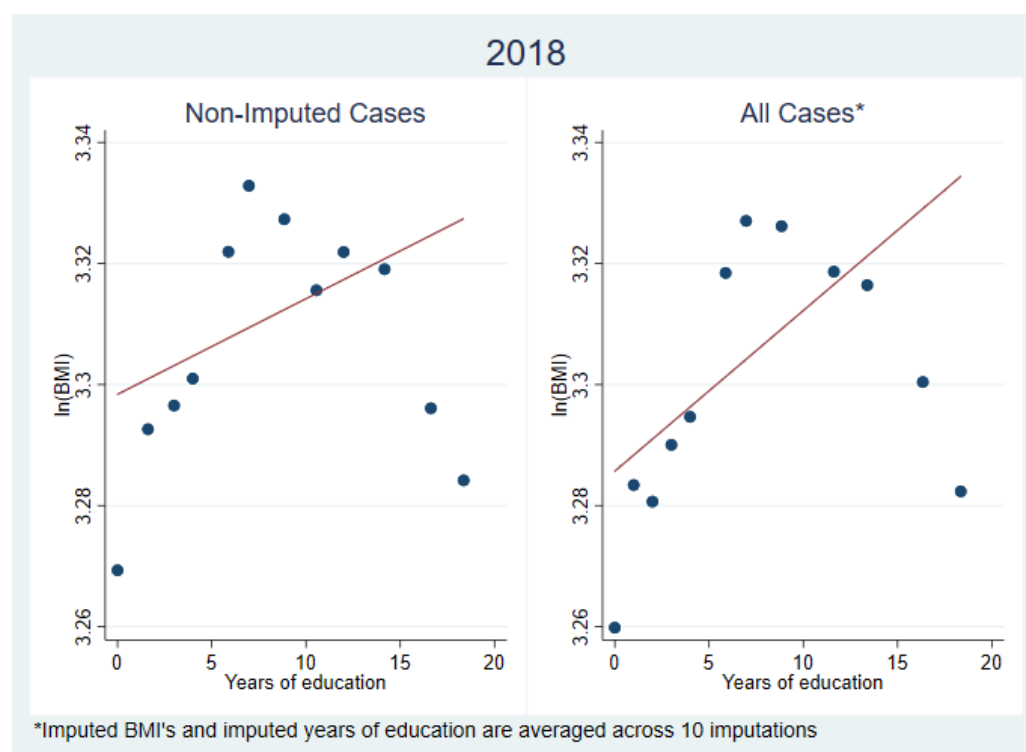


Figure 7f. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2021 Wave)

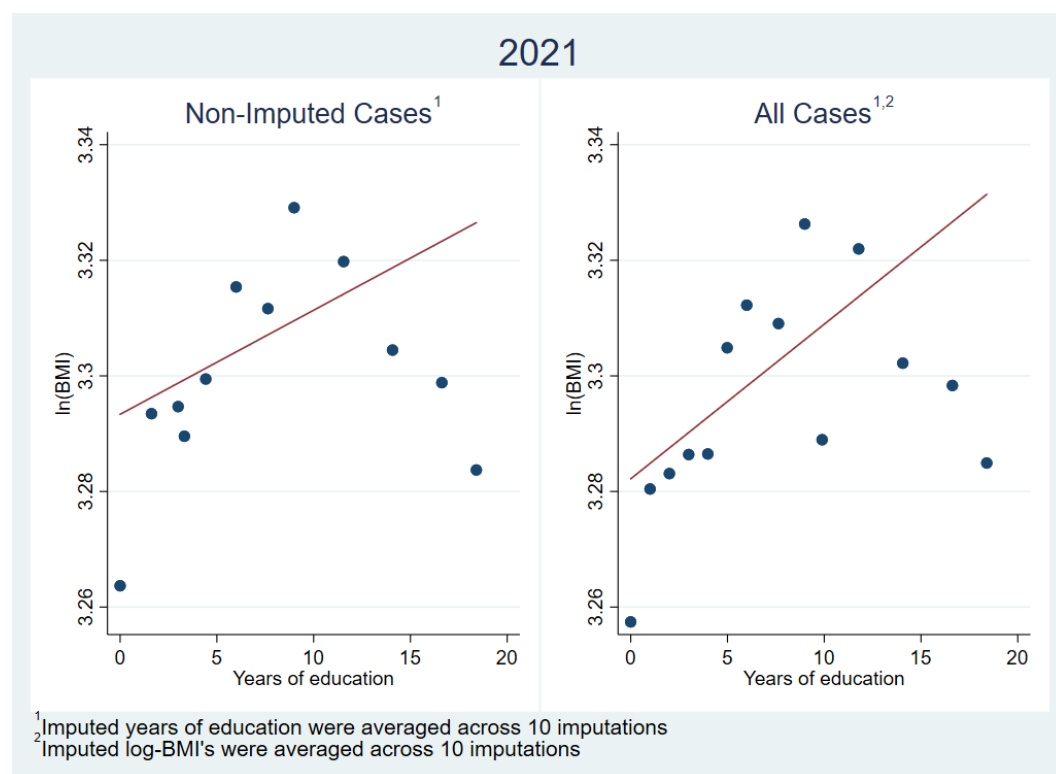
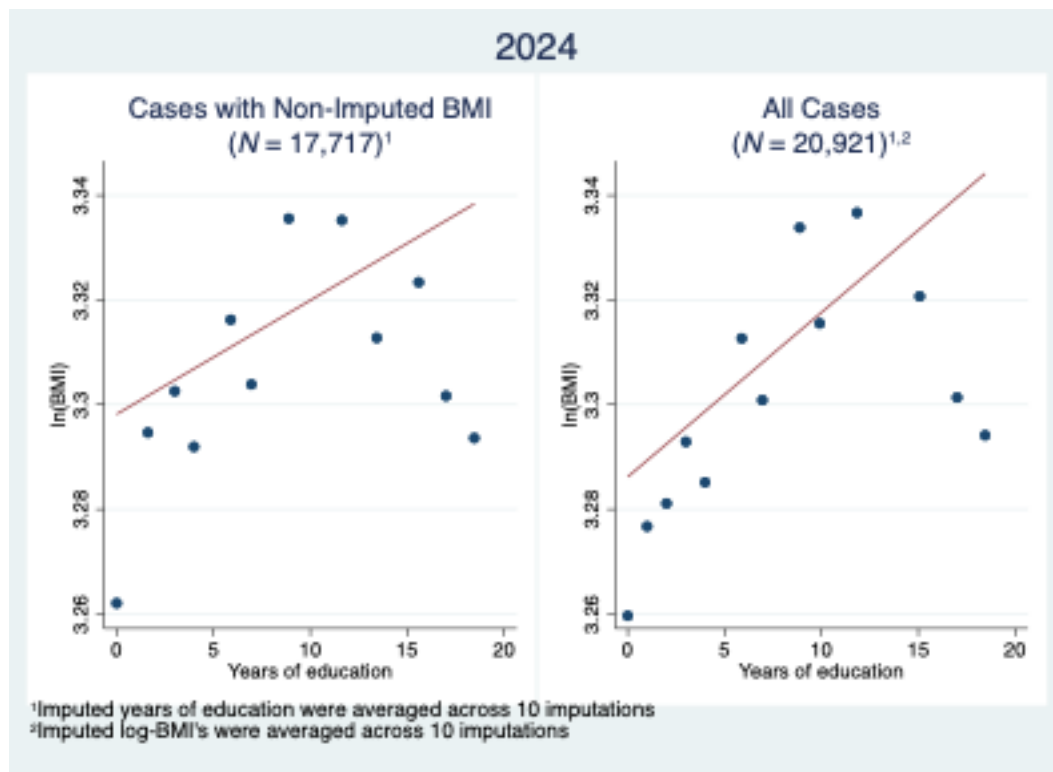


Figure 7g. Binned Scatter Plots of Mean Log-BMI by Years of Education among Non-Imputed Cases and All Cases (2024 Wave)



V. References

1. MHAS Mexican Health and Aging Study, (2001–2018). Data Files and Documentation (public use): Mexican Health and Aging Study, (Codebooks). Retrieved from www.MHASweb.org on December 7, 2020.
2. Rässler, S., Rubin, D. B., & Zell, E. R. (2012). Imputation. *WIREs Computational Statistics*, 5(1), 20–29. doi: [10.1002/wics.1240](https://doi.org/10.1002/wics.1240)
3. Van Buuren, S., Boshuizen, H.C., & Knook, D.L. (1999). Multiple imputation of missing blood pressure covariates in survival analysis. *Statistics in Medicine*, 18(6), 681–694. doi: [10.1002/\(SICI\)1097-0258\(19990330\)18:6<681::AID-SIM71>3.0.CO;2-R](https://doi.org/10.1002/(SICI)1097-0258(19990330)18:6<681::AID-SIM71>3.0.CO;2-R)
4. Van Buuren, S. (2007). Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical Methods in Medical Research*, 16(3), 219–242. doi: [10.1177/0962280206074463](https://doi.org/10.1177/0962280206074463)

Appendix A – STATA Program

2001

```
*** ENASEM 2001 – IMPUTACIÓN DE ALTURA Y PESO AUTO-REPORTADOS Y DE ÍNDICE DE ***
*** MASA CORPORAL (IMC)
***

*** MHAS 2001 – IMPUTATION OF SELF-REPORTED HEIGHT AND WEIGHT AND OF BODY ***
*** MASS INDEX (BMI)
***

*** Versión agosto 2020 (August 2020 Version) ***

version 15

local yr "01"

*** PASO 1: DEFINIR UBICACIONES DE CARPETAS ***
*** STEP 1: DEFINE LOCATIONS OF FILES ***

*** Convertir los siguientes macros en los directorios donde se guardan las ***
*** petas descritas abajo (Convert the following macros to the directories ***
*** where the files described below are stored)
***

local maestro "||maestro||" // Archivo maestro de seguimiento más reciente
                                // (Most recent master follow-up file)

local indiv "||individual||" // Bases de datos crudos al nivel individual
                                // (Individual-level databases)

local gph_out "||graph_output||"
                                // Figuras construidas en este programa (Figures constructed in this
                                // program)

local output "||output||" // Base de datos final con medias y desviaciones
                                // estándares entre imputaciones (Final dataset with
                                // means and standard deviations across imputations)

*** Cargar datos brutos (Load raw data) ***

local rmr "18" // Año de la ronda más reciente (Year of most recent wave)

local maestro "`maestro'\master_follow_up_file_20`rmr'.dta'"

use unhhid cunicah np codent`yr' subhog`yr' antro`yr' tamloc`yr' ///
    edad`yr' sexo`yr' tipent`yr' if inlist(tipent`yr',11,12,21,22) ///
    using ``maestro'', clear

** Cargar otras variables para imputación (Load other variables for imputation)
```

```
generate ps3 = codent`yr'
```

```
* Educación
```

```
merge 1:1 unhhid ps3 using ``indiv'\sect_a.dta'', assert(match master) ///  
    nogenerate keepusing(escola) keep(match)  
    // 216 observaciones tienen valores válidos de tipent_01, aunque los  
    // sujetos no completaron la entrevista. (216 observations have valid  
    // values of tipent_01, even though the subjects did not complete the  
    // interview.)
```

```
merge 1:1 unhhid ps3 using ``indiv'\sect_c.dta'', assert(match master) ///  
    nogenerate keepusing(c6 c71 c72)  
    // Altura y peso auto-reportados y diabetes (Self-reported height and  
    // weight and diabetes)
```

```
merge 1:1 unhhid ps3 using ``indiv'\sect_l.dta'', assert(match master) ///  
    nogenerate keepusing(l3 l4) // Altura y peso medidos (Measured height and weight)
```

```
*** PASO 2: PREPARAR DATOS CRUDOS PARA IMPUTACIÓN ***
```

```
*** STEP 2: PREPARE RAW DATA FOR IMPUTATION ***
```

```
*** Sustituir valores que indican datos faltantes por . (Replace values that ***
```

```
*** indicate missing data with .)  
***
```

```
replace c71 = . if inlist(c71,888,999)  
replace c72 = . if inrange(c72,888,999)  
replace escola = . if escola == 99  
replace l3 = . if antrop_`yr' == 1 & inlist(l3,995,996,997,999)  
replace l4 = . if antrop_`yr' == 1 & inlist(l4,995,996,997,999)
```

```
*** Sustituir alturas y pesos auto-reportados que no emparejan con valores ***
```

```
*** medidos por . (Replace self-reported heights and weights that do not ***
```

```
*** match measured values with .)
```

```
***
```

```
replace c71 = . if antrop_`yr' == 1 & !(missing(c71) | missing(l3)) & ///  
    abs((c71 - l3) / l3) > 0.1  
replace c72 = . if antrop_`yr' == 1 & !(missing(c72) | missing(l4)) & ///  
    abs((c72 - l4) / l4) > 0.1
```

```
*** Crear indicadores de alturas y pesos faltantes (Create indicator of mis- ***
```

```

*** sing heights and weights)
***

generate byte ht_miss_`yr' = missing(c72)
generate byte wt_miss_`yr' = missing(c71)

*** Sustituir alturas y pesos medidos que faltan porque los sujetos ***
*** correspondientes no fueron seleccionados para medidas por 0 ***
*** antropometricas (Substitute 0's for height and weight measure- ***
*** ments that are missing because subject was not selected for an- ***
*** thropometric measurements) ***

replace l3 = 0 if antro_`yr' == 0
replace l4 = 0 if antro_`yr' == 0

*** PASO 3: IMPUTAR DATOS FALTANTES ***
*** STEP 3: IMPUTE MISSING DATA ***

*** Definir parámetros de imputación (cambiar los siguientes macros para modificar las imputaciones ***
*** Define parameters of imputation (change the following macros to modify imputations ***
local n_imp = 10 // Número de imputaciones (Number of imputations)
local n_iter = 450 // Número de iteraciones en imputación multivariada usanda ecuaciones ligadas
// Number of iterations in multivariate imputation using chained
equations (MICE)
local k = 5 // Número de "vecinos más cercanos" para emparejar medias predictivas
// Number of "nearest neighbors" for predictive mean matching (PMM)
local rng_seed = 101 // "Semilla" para generar números aleatorios ("Seed" to
// generate random numbers)

*** Declarar base de datos como "datos de imputación" y variables como ***
*** "imputadas" o "regulares" (ningún valor se imputará) ***
*** Declare database as "imputation data" and variables as "imputed or ***
*** "regular" (no values will be imputed) ***

mi set wide

mi register imputed c71 c72 l3 l4 escola

mi register regular unhhid subhog_`yr' tipent_`yr' tamloc_`yr' sexo_`yr' ///
codent_`yr' cunicah np ht_miss_`yr' wt_miss_`yr' antro_`yr' ps3 c6 ///
edad_`yr'

*** Imputar valores faltantes (Impute missing values) ***

```

```

mi impute chained (pmm, knn(`k')) escola c71 c72 (pmm, knn(`k')) ///
    conditional(if antro_`yr' == 1)) l3 l4 = i.sexo_`yr' i.tamloc_`yr' ///
    edad_`yr', burnin(`n_iter') rseed(`rng_seed') add(`n_imp')

*** Crear variables pasivas (definidas como funciones de variables ***
*** imputadas)
***

*** Create passive variables (defined as functions of imputed ***
*** variables)
***

mi passive: generate bmi_`yr' = c71 / ((c72 / 100)^2)
mi passive: generate log_bmi_`yr' = log(bmi_`yr')

*** PASO 4: ESTIMAR CONEXIONES MARGINALES ENTRE EL LOGARITMO DE ÍMC Y ***
*** EDUCACIÓN, TAMAÑO DE LOCALIDAD Y DIABETES USANDO DATOS IMPUTADOS ***
*** STEP 4: ESTIMATE MARGINAL ASSOCIATIONS BETWEEN LOG-BMI AND EDUCATION, ***
*** LOCALITY SIZE, AND DIABETES USING IMPUTED DATA
***

mi estimate: regress log_bmi escola // Educación
mi estimate: regress log_bmi ib4.tamloc_`yr' // Tamaño de localidad (Locality size)
mi estimate: regress log_bmi ib2.c6 if inlist(c6,1,2) // Diabetes

*** PASO 5: PLEGAR BASE DE DATOS EN ESTADÍSTICAS SUMARIAS NECESARIAS PARA ***
*** BASE DE DATOS FINAL Y PARA TABLAS Y FIGURAS
***

*** STEP 5: COLLAPSE DATASET INTO SUMMARY STATISTICS NECESSARY FOR FINAL ***
*** DATASET AND FOR TABLES AND FIGURES
***

mi convert mlong, clear
collapse (mean) imp_ht_`yr'=c72 imp_wt_`yr'=c71 imp_bmi_`yr'=bmi_`yr' ///
    log_bmi_mean_`yr'=log_bmi_`yr' escola_mean=escola (sd) ///
    imp_ht_sd_`yr'=c72 imp_wt_sd_`yr'=c71 imp_bmi_sd_`yr'=bmi_`yr' ///
    escola_sd=escola (median) bmi_med_`yr'=bmi_`yr' (firstnm) unhhid ///
    codent_`yr' subhog_`yr' _mi_m _mi_id _mi_miss ht_miss_`yr' ///
    wt_miss_`yr' c6 tamloc_`yr', by(cunicah np)

mi extract 0, clear // Stata ya no reconoce estos datos como datos de
    // imputación. (Stata no longer recognizes these data as im-
    // putation data.)

*** Sustituir desviaciones estándares por 0 en observaciones completas (Re- ***
*** place standard deviations with 0 in complete observations)
***

replace imp_ht_sd_`yr' = 0 if ht_miss_`yr' == 0

```

```

replace imp_wt_sd`yr' = 0 if wt_miss`yr' == 0
replace imp_bmi_sd`yr' = 0 if ht_miss`yr' == 0 & wt_miss`yr' == 0

*** Reetiquetar variables que se les enviarán a usuarios (Relabel variables ***
*** that will be sent to users)
***
label variable imp_ht`yr' "Imputed/observed height without shoes (cm) 20`yr'"
label variable imp_wt`yr' "Imputed/observed weight (kg) 20`yr'"
label variable imp_bmi`yr' "Imputed/observed BMI (kg/m^2) 20`yr'"
label variable imp_ht_sd`yr' "Between-imputation standard deviation of height 20`yr'"
label variable imp_wt_sd`yr' "Between-imputation standard deviation of weight 20`yr'"
label variable imp_bmi_sd`yr' "Between-imputation standard deviation of BMI 20`yr'"
label variable codent`yr' "Person identification code 20`yr' (=ps3)"
label variable subhog`yr' "20`yr' sub-household identifier"
label variable ht_miss`yr' "Height was imputed 20`yr'"
label variable wt_miss`yr' "Weight was imputed 20`yr'"

*** Volver a aplicar etiquetas de valores (Reapply value labels) ***
label values subhog`yr' clavesubhog
label values codent`yr' codent`yr'

*** Agregarle notas sobre valores imputados a base de datos (Add note about ***
*** imputed values to dataset)
***
notes: Values of imp_ht`yr' and imp_wt`yr' where height and weight were imputed (ht_miss`yr' = 1
and wt_miss`yr' = 1, respectively) were averaged across 10 imputations.

notes: Values of imp_bmi`yr' where BMI was imputed (ht_miss`yr' = 1 and/or wt_miss`yr' = 1) were
averaged across 10 imputations.

*** Agregarle nota sobre desviaciones estándares entre imputaciones a base ***
*** de datos (Add note about between-imputation standard deviations to ***
*** dataset)
***
notes: imp_ht_sd`yr', imp_wt_sd`yr', and imp_bmi_sd`yr' are 0 where height, weight, and BMI,
respectively, were observed (i.e., ht_miss`yr' = 0, wt_miss`yr' = 0, and both ht_miss`yr' and
wt_miss`yr' are 0, respectively) because observed values are constant across imputations.

*** PASO 6: COMPARAR RESULTADOS DE ANALIZAR DATOS IMPUTADOS CON LOS DE ***
*** ANALIZAR DATOS BRUTOS
***
*** STEP 6: COMPARE RESULTS OF ANALYZING IMPUTED DATA WITH THOSE FROM ANALY- ***

```

```
*** ZING RAW DATA
***
```

```
*** Estimar conexiones marginales entre el logaritmo de ÍMC y educación, ***
*** tamaño de localidad y diabetes usando datos brutos (Estimate margi- ***
*** nal associations between log-BMI and education, locality size, and ***
*** diabetes using raw data
***
```

```
regress log_bmi_mean_`yr' escola_mean if ht_miss_`yr' == 0 & wt_miss_`yr' ///
      == 0 & (escola_sd == 0 | missing(escola_sd))
regress log_bmi_mean_`yr' ib4.tamloc_`yr' if ht_miss_`yr' == 0 & ///
      wt_miss_`yr' == 0
regress log_bmi_mean_`yr' ib2.c6 if ht_miss_`yr' == 0 & wt_miss_`yr' == 0 & ///
      inlist(c6,1,2)
```

```
*** Computar estadísticas sumarias y percentiles de alturas y peso auto- ***
*** reportados entre observaciones donde fueron observados, entre las ***
*** donde fueron imputados y entre todas las observaciones (Compute sum- ***
*** mary statistics and percentiles of self-reported heights and weights ***
*** among observations where they were observed, among those where they ***
*** were imputed, and among all observations) ***
```

```
foreach i in ht wt {
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 0, detail
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 1, detail
    summarize imp_`i'_`yr', detail
}
```

```
*** Construir histogramas de alturas y pesos auto-reportados entre ***
*** observaciones donde fueron observados, entre las donde fueron ***
*** imputados y entre todas las observaciones (Construct histograms ***
*** of self-reported heights and weights among observations where ***
*** they were observed, among those where they were imputed, and ***
*** among all observations) ***
```

```
graph drop _all
```

```
* Altura (Height)
```

```
histogram imp_ht_`yr' if ht_miss_`yr' == 0, xtitle("") ///
      title("Non-Imputed Cases") name(gr1)
histogram imp_ht_`yr' if ht_miss_`yr' == 1, xtitle("") ///
      title("Imputed Cases") name(gr2) ytitle("")
```

```

histogram imp_ht_`yr', xtitle("") title("All Cases*") name(gr3) ///
    ytitle("")
graph combine gr1 gr2 gr3, xcommon ycommon title("Height (cm)", position(6)) ///
    rows(1) name(gr_ht) // Combina histogramas de altura en una figura

* Peso (Weight)
histogram imp_wt_`yr' if wt_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr4)
histogram imp_wt_`yr' if wt_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases*") name(gr5) ytitle("")
histogram imp_wt_`yr', xtitle("") title("All Cases*") name(gr6) ytitle("")
graph combine gr4 gr5 gr6, xcommon ycommon title("Weight (kg)", position(6)) ///
    rows(1) name(gr_wt) // Combina histogramas de peso en una figura

* Combinar histogramas de altura con histogramas de peso (Combine histograms of
* height with hitsograms of weigth)
graph combine gr_ht gr_wt, cols(1) title("`20`yr'") ///
    note("*Imputed values in each observation are averaged across 10 imputations")

* Guardar histogramas (Save histograms)
graph save ``gph_out'\ht_wt_hist`yr'.gph"

*** Construir diagramas de caja de ÍMC por tamaño de localidad entre ***
*** observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para ***
*** observaciones donde ÍMC fue imputado (Construct box plots of BMI ***
*** by locality size among observations where BMI was observed and ***
*** among all observations, using medians across imputations for ob- ***
*** servations where BMI was imputed) ***

graph drop _all

* Casos no imputados (Non-imputed cases)
graph hbox bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0, ///
    over(tamloc_`yr') title("Non-Imputed Cases") ///
    ytitle("BMI (kg/m2)") name(gr1)

* Todos los casos (All cases)
graph hbox bmi_med_`yr', over(tamloc_`yr') title("All Cases*") ///
    ytitle("Median BMI (kg/m2)") name(gr2)

```

```

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'`) ycommon ///
    note("*BMI's for imputed cases are medians among 10 imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_tam_loc_box`yr'.gph''

*** Construir diagramas de caja de ÍMC por presencia o ausencia de diabetes ***
*** entre observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para observaciones ***
*** donde ÍMC fue imputado y excluyendo observaciones donde faltan datos ***
*** sobre diabetes (Construct box plots of BMI by presence or absence of ***
*** diabetes among observations where BMI was observed and among all obser- ***
*** vations, using medians across imputations for observations wher BMI was ***
*** imputed and excluding observations where data on diabetes are missing) ***
graph drop all

* Casos no imputados (Non-imputed cases)
graph box bmi_med`yr' if ht_miss`yr' == 0 & wt_miss`yr' == 0 & ///
    inlist(c6,1,2), over(c6, relabel(1 "Diabetic" 2 "Not Diabetic")) ///
    title("Non-Imputed Cases{superscript:1}") ///
    ytitle("BMI (kg/m{superscript:2})") name(gr1)

* Todos los casos (All cases)
graph box bmi_med`yr' if inlist(c6,1,2), over(c6, relabel(1 "Diabetic" 2 ///
    "Not Diabetic")) ///
    title("All Cases{superscript:1,2}") ///
    ytitle("Median BMI (kg/m{superscript:2})") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'`) ycommon ///
    note("{superscript:1}Excludes observations where diabetic status is missing" ///
    "{superscript:2}BMI's for imputed cases are medians among 10 imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_diab_box`yr'.gph''

```

```

*** Construir diagramas de dispersión "agrupados" de logaritmos de ÍMC por ***
*** años de educación entre observaciones donde ÍMC fue observado y entre ***
*** todas las observaciones (Construct "binned" scatter plots of log-BMI ***
*** vs. years of education among observations where BMI was observed and ***
*** among all observations)
***
graph drop _all

* Casos no imputados (Non-imputed cases)
binscatter log_bmi_mean_`yr' escola_mean if ht_miss_`yr' == 0 & ///
           wt_miss_`yr' == 0, xtitle("Years of education") ytitle("ln(BMI)") ///
           title("Non-Imputed Cases") name(gr1)

* Todos los casos (All cases)
binscatter log_bmi_mean_`yr' escola_mean, xtitle("Years of education") ///
           ytitle("ln(BMI)") title("All Cases*") name(gr2)

* Combinar diagramas de dispersión agrupados entre casos no imputados y entre
* todos los casos (Combine binned scatter plots among non-imputed cases and
* among all cases)
graph combine gr1 gr2, xcommon ycommon rows(1) title("`20`yr'") ///
           note("*Imputed BMI's and imputed years of education are averaged across 10
imputations")

* Guardar diagramas de dispersión (Save scatter plots)
graph save "`gph_out'\log_bmi_edu_bin`yr'.gph"

*** PASO 7: GUARDAR BASE DE DATOS FINAL ***
*** STEP 7: SAVE FINAL DATASET ***

*** Borrar escalares (Delete scalars) ***
scalar drop _all

*** Borrar variables no necesarias (Delete unnecessary variables) ***
keep cunicah np codent_`yr' subhog_`yr' tamloc_`yr' imp_ht_`yr' ///
           imp_ht_sd_`yr' imp_wt_`yr' imp_wt_sd_`yr' imp_bmi_`yr' ///
           imp_bmi_sd_`yr' ht_miss_`yr' wt_miss_`yr'

*** Cargar otras variables importantes, por ejemplo tipo de entrevista y ***

```

```

*** peso de muestreo (Load other important variables, e.g. type of      ***
*** interview and sampling weight)
***
merge 1:m unhhid codent`yr' using `"maestro"', nogenerate assert(match ///
        using) keep(match) keepusing(UNHHID CUNICAH NP UNHHIDNP tipent`yr' ///
        tamloc`yr' factori`yr' edad`yr' sexo`yr')

*** Guardar datos finales (Save final data) ***
save `"output"\MHAS_20`yr'_Weight_and_Height_IMPUTED.dta", replace

```

2003

```
*** ENASEM 2003 — IMPUTACIÓN DE ALTURA Y PESO AUTO-REPORTADOS Y DE ÍNDICE DE ***
*** MASA CORPORAL (ÍMC)
***
*** MHAS 2003 — IMPUTATION OF SELF-REPORTED HEIGHT AND WEIGHT AND OF BODY ***
*** MASS INDEX (BMI)
***
*** Versión septiembre 2020 (September 2020 Version) ***

version 15
local yr "03"

*** PASO 1: DEFINIR UBICACIONES DE CARPETAS ***
*** STEP 1: DEFINE LOCATIONS OF FILES ***

*** Convertir los siguientes macros en los directorios donde se guardan las ***
*** petas descritas abajo (Convert the following macros to the directories ***
*** where the files described below are stored)
***

local maestro "||maestro||" // Archivo maestro de seguimiento (Master follow-up
file)

local indiv "||individual||" // Bases de datos crudos al nivel individual de
2003
                                // (Individual-level databases from
2003)

local indiv01 "||individual_01||"
                                // Datos al nivel individual de 2001 (Individual-level data from 2001)

local gph_out "||graph_output||"
                                // Figuras construidas en este programa (Figures constructed in this
                                // program)

local output "||output||" // Base de datos final con medias y desviaciones
                                // estándares entre imputaciones (Final dataset
with
                                // means and standard deviations across
imputations)

*** Cargar datos brutos (Load raw data) ***
```

```

local rmr "18" // Año de la ronda más reciente (Year of most recent wave)
local maestro ``maestro'\master_follow_up_file_20`rmr'.dta''

    // Se recomienda que se usa el archivo maestro de seguimiento
disponible

    // más reciente. (It is recommended that the most recent available
mas-

    // the follow-up file is used.)
use cunichah np acthog subhog`yr' codent`yr' antropo`yr' tipent`yr' ///
    tamloc_01 edad`yr' sexo`yr' unhhid codent01 new_sample`yr' if ///
    inlist(tipent`yr',11,12,21,22) using ``maestro'', clear

*** Cargar otras variables para imputación y figuras (Load other variables ***
*** for imputation and figures)
***

* Altura y peso (Height and weight)
generate ent2 = codent03
tostring acthog, replace format(%02.0f) force

    // acthog es cadena de caracteres en bases de datos de 2003 (acthog is
    // a string of characters in datasets from 2003)
merge 1:1 cunichah acthog ent2 using ``indiv'\sect_c.dta'', assert(match) ///
    keepusing(c6 c66 c67) nogenerate
merge 1:1 cunichah acthog ent2 using ``indiv'\sect_1.dta'', assert(master ///
    match) keepusing(l3 l4) nogenerate

* Educación
merge 1:1 cunichah acthog ent2 using ``indiv'\sect_aa.dta'', assert(master ///
    match) keepusing(aa4a_3) nogenerate

tempfile a01
preserve
use ``indiv01'\sect_a.dta'', clear
rename a3_3 a3_3_01
save `a01'
restore
generate ps3 = codent01

```

```

merge m:1 unhhid ps3 using `a01', keep(match master) keepusing(a3_3_01) ///
      nogenerate

generate yrschool = aa4a_3 if new_sample_`yr' == 2
replace yrschool = a3_3_01 if new_sample_`yr' == 1

*** PASO 2: PREPARAR DATOS CRUDOS PARA IMPUTACIÓN ***
*** STEP 2: PREPARE RAW DATA FOR IMPUTATION ***

*** Sustituir valores que indican datos faltantes por . (Replace values that ***
*** indicate missing data with .)
***
replace c66 = . if inlist(c66,888,999)
replace c67 = . if inlist(c67,888,999)
replace yrschool = . if yrschool == 99
replace edad_03 = . if edad_03 == 999
replace l3 = . if antrop_`yr' == 1 & inlist(l3,995,996,997,999)
replace l4 = . if antrop_`yr' == 1 & inlist(l4,995,996,997,999)

*** Sustituir alturas y pesos auto-reportados que no emparejan con valores ***
*** medidos por . (Replace self-reported heights and weights that do not ***
*** match measured values with .)
***
replace c66 = . if antrop_`yr' == 1 & !(missing(c66) | missing(l3)) & ///
      abs((c66 - l3) / l3) > 0.1
replace c67 = . if antrop_`yr' == 1 & !(missing(c67) | missing(l4)) & ///
      abs((c67 - l4) / l4) > 0.1

*** Sustituir alturas reportados en 2003 que no emparejan con las reportadas ***
*** en 2001 por . (Replace heights reported in 2003 that do not match those ***
*** reported in 2001 with .)
***
tempfile c01
preserve
use ``\indiv01'\sect_c.dta"', clear

```

```

rename c72 c72_01
save `c01'
restore
merge m:1 unhhid ps3 using `c01', keep(match master) keepusing(c72_01) ///
      nogenerate
replace c67 = . if new_sample_03 == 1 & !(missing(c67) | missing(c72_01) | ///
      inrange(c72_01,888,999)) & abs((c67 - c72_01) / c72_01) > 0.1

*** Crear indicadores de alturas y pesos faltantes (Create indicator of mis- ***
*** sing heights and weights)
***
generate byte ht_miss_`yr' = missing(c67)
generate byte wt_miss_`yr' = missing(c66)

*** Sustituir alturas y pesos medidos que faltan porque los sujetos ***
*** correspondientes no fueron seleccionados para medidas por 0 ***
*** antropometricas (Substitute 0's for height and weight measure- ***
*** ments that are missing because subject was not selected for an- ***
*** thropometric measurements)
***
replace l3 = 0 if antropo_`yr' == 0
replace l4 = 0 if antropo_`yr' == 0

*** PASO 3: IMPUTAR DATOS FALTANTES ***
*** STEP 3: IMPUTE MISSING DATA ***

*** Definir parámetros de imputación (cambiar los siguientes macros para ***
*** modificar las imputaciones)
***
*** Define parameters of imputation (change the following macros to modify ***
*** imputations)
***
local n_imp = 10 // Número de imputaciones (Number of imputations)
local n_iter = 350 // Número de iteraciones en imputación multivariada usanda
// ecuaciones ligadas

```

```

// Number of iterations in multivariate imputation using
// chained equations (MICE)

local k = 5 // Número de "vecinos más cercanos" para emparejar medias predictivas
// Number of "nearest neighbors" for predictive mean matching
(PMM)

local rng_seed = 101 // "Semilla" para generar números aleatorios ("Seed" to
// generate random numbers)

*** Declarar base de datos como "datos de imputación" y variables como ***
*** "imputadas" o "regulares" (ningún valor se imputará) ***
*** Declare database as "imputation data" and variables as "imputed or ***
*** "regular" (no values will be imputed)
***

mi set wide
mi register imputed c66 c67 l3 l4 edad_`yr' yrschool
mi register regular tamloc_01 sexo_`yr' acthog codent_`yr' cunichah np ///
    ht_miss_`yr' wt_miss_`yr' antro_`yr' c6

*** Imputar valores faltantes (Impute missing values) ***
mi impute chained (pmm, knn(`k')) edad_`yr' yrschool c66 c67 (pmm, knn(`k') ///
    conditional(if antro_`yr' == 1)) l3 l4 = i.sexo_`yr' i.tamloc_01, ///
    burnin(`n_iter') rseed(`rng_seed') add(`n_imp')

*** Crear variables pasivas (definidas como funciones de variables ***
*** imputadas)
***

*** Create passive variables (defined as functions of imputed ***
*** variables) ***

mi passive: generate bmi_`yr' = c66 / ((c67 / 100)^2)
mi passive: generate log_bmi_`yr' = log(bmi_`yr')

*** PASO 4: ESTIMAR CONEXIONES MARGINALES ENTRE EL LOGARITMO DE ÍMC Y ***
*** EDUCACIÓN, TAMAÑO DE LOCALIDAD Y DIABETES USANDO DATOS IMPUTADOS ***
*** STEP 4: ESTIMATE MARGINAL ASSOCIATIONS BETWEEN LOG-BMI AND EDUCATION, ***

```

```

*** LOCALITY SIZE, AND DIABETES USING IMPUTED DATA
***
mi estimate: regress log_bmi_`yr' yrschool // Educación
mi estimate: regress log_bmi_`yr' ib4.tamloc_01 // Tamaño de localidad (Locality
size)
mi estimate: regress log_bmi_`yr' ib2.c6 if inlist(c6,1,2) // Diabetes

*** PASO 5: PLEGAR BASE DE DATOS EN ESTADÍSTICAS SUMARIAS NECESARIAS PARA ***
*** BASE DE DATOS FINAL Y PARA TABLAS Y FIGURAS
***
*** STEP 5: COLLAPSE DATASET INTO SUMMARY STATISTICS NECESSARY FOR FINAL ***
*** DATASET AND FOR TABLES AND FIGURES
***
mi convert mlong, clear
collapse (mean) imp_ht_`yr'=c67 imp_wt_`yr'=c66 imp_bmi_`yr'=bmi_`yr' ///
      yrschool_mean=yrschool log_bmi_mean_`yr'=log_bmi_`yr' ///
      (sd) imp_ht_sd_`yr'=c67 imp_wt_sd_`yr'=c66 imp_bmi_sd_`yr'=bmi_`yr'
///
      yrschool_sd=yrschool (median) bmi_med_`yr'=bmi_`yr' (firstnm) ///
      subhog_`yr' acthog codent`yr' ht_miss_`yr' wt_miss_`yr' _mi_id _mi_m
///
      _mi_miss c6 tamloc_01, by(cunicah np)
mi extract 0, clear // Stata ya no reconoce estos datos como datos de
                        // imputación. (Stata no longer recognizes these data
as im-
                        // putation data.)

* Sustituir desviaciones estándares por 0 en observaciones completas (Replace
* standard deviations with 0 in complete observations)
replace imp_ht_sd_`yr' = 0 if ht_miss_`yr' == 0
replace imp_wt_sd_`yr' = 0 if wt_miss_`yr' == 0
replace imp_bmi_sd_`yr' = 0 if ht_miss_`yr' == 0 & wt_miss_`yr' == 0

*** PASO 6: COMPARAR RESULTADOS DE ANALIZAR DATOS IMPUTADOS CON LOS DE ***
*** ANALIZAR DATOS BRUTOS
***
*** STEP 6: COMPARE RESULTS OF ANALYZING IMPUTED DATA WITH THOSE FROM ANALY- ***

```

```
*** ZING RAW DATA
***
```

```
*** Estimar conexiones marginales entre el logaritmo de ÍMC y educación, ***
*** tamaño de localidad y diabetes usando datos brutos (Estimate margi- ***
*** nal associations between log-BMI and education, locality size, and ***
*** diabetes using raw data
***
```

```
regress log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & wt_miss_`yr' ///
      == 0 & (yrschool_sd == 0 | missing(yrschool_sd))
regress log_bmi_mean_`yr' ib4.tamloc_01 if ht_miss_`yr' == 0 & wt_miss_`yr' ///
      == 0
regress log_bmi_mean_`yr' ib2.c6 if ht_miss_`yr' == 0 & wt_miss_`yr' == 0 & ///
      inlist(c6,1,2)
```

```
*** Computar estadísticas sumarias y percentiles de alturas y peso auto- ***
*** reportados entre observaciones donde fueron observados, entre las ***
*** donde fueron imputados y entre todas las observaciones (Compute sum- ***
*** mary statistics and percentiles of self-reported heights and weights ***
*** among observations where they were observed, among those where they ***
*** were imputed, and among all observations)
***
```

```
foreach i in ht wt {
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 0, detail
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 1, detail
    summarize imp_`i'_`yr', detail
}
```

```
*** Construir histogramas de alturas y pesos auto-reportados entre ***
*** observaciones donde fueron observados, entre las donde fueron ***
*** imputados y entre todas las observaciones (Construct histograms ***
*** of self-reported heights and weights among observations where ***
*** they were observed, among those where they were imputed, and ***
*** among all observations)
***
```

```
graph drop _all
```

```

* Altura (Height)
histogram imp_ht_`yr' if ht_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr1)
histogram imp_ht_`yr' if ht_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases*") name(gr2) ytitle("")
histogram imp_ht_`yr', xtitle("") title("All Cases*") name(gr3) ///
    ytitle("")
graph combine gr1 gr2 gr3, xcommon ycommon title("Height (cm)", position(6)) ///
    rows(1) name(gr_ht) // Combina histogramas de altura en una figura

* Peso (Weight)
histogram imp_wt_`yr' if wt_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr4)
histogram imp_wt_`yr' if wt_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases*") name(gr5) ytitle("")
histogram imp_wt_`yr', xtitle("") title("All Cases*") name(gr6) ytitle("")
graph combine gr4 gr5 gr6, xcommon ycommon title("Weight (kg)", position(6)) ///
    rows(1) name(gr_wt) // Combina histogramas de peso en una figura

* Combinar histogramas de altura con histogramas de peso (Combine histograms of
* height with hitsograms of weigth)
graph combine gr_ht gr_wt, cols(1) title("`20`yr'") ///
    note("*Imputed values in each observation are averaged across 10
imputations")

* Guardar histogramas (Save histograms)
graph save ``gph_out'\ht_wt_hist`yr'.gph'', replace

*** Construir diagramas de caja de ÍMC por tamaño de localidad entre ***
*** observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para ***
*** observaciones donde ÍMC fue imputado (Construct box plots of BMI ***
*** by locality size among observations where BMI was observed and ***
*** among all observations, using medians across imputations for ob- ***

```

```

*** servations where BMI was imputed) ***

graph drop _all

* Casos no imputados (Non-imputed cases)
graph hbox bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0, ///
    over(tamloc_01) title("Non-Imputed Cases") ///
    ytitle("BMI (kg/m2)") name(gr1)

* Todos los casos (All cases)
graph hbox bmi_med_`yr', over(tamloc_01) title("All Cases*") ///
    ytitle("Median BMI (kg/m2)") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'") ycommon ///
    note("*BMI's for imputed cases are medians among 10 imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_tam_loc_box`yr'.gph", replace

*** Construir diagramas de caja de ÍMC por presencia o ausencia de diabetes ***
*** entre observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para observaciones ***
*** donde ÍMC fue imputado y excluyendo observaciones donde faltan datos ***
*** sobre diabetes (Construct box plots of BMI by presence or absence of ***
*** diabetes among observations where BMI was observed and among all obser- ***
*** vations, using medians across imutations for observations wher BMI was ***
*** imputed and excluding observations where data on diabetes are missing) ***
graph drop _all

* Casos no imputados (Non-imputed cases)
graph box bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0 & ///
    inlist(c6,1,2), over(c6, relabel(1 "Diabetic" 2 "Not Diabetic")) ///
    title("Non-Imputed Cases1") ///

```

```

        ytitle("BMI (kg/m2)") name(gr1)

* Todos los casos (All cases)
graph box bmi_med_`yr' if inlist(c6,1,2), over(c6, relabel(1 "Diabetic" 2 ///
    "Not Diabetic")) ///
    title("All Cases1,2") ///
    ytitle("Median BMI (kg/m2)") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'") ycommon ///
    note("{1Excludes observations where diabetic status is
missing" ///
    "{2BMI's for imputed cases are medians among 10
imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_diab_box`yr'.gph"', replace

*** Construir diagramas de dispersión "agrupados" de logaritmos de ÍMC por ***
*** años de educación entre observaciones donde ÍMC fue observado y entre ***
*** todas las observaciones (Construct "binned" scatter plots of log-BMI ***
*** vs. years of education among observations where BMI was observed and ***
*** among all observations)
***
graph drop _all

* Casos no imputados (Non-imputed cases)
binscatter log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & ///
    wt_miss_`yr' == 0, xtitle("Years of education") ytitle("ln(BMI)") ///
    title("Non-Imputed Cases") name(gr1)

* Todos los casos (All cases)
binscatter log_bmi_mean_`yr' yrschool_mean, xtitle("Years of education") ///
    ytitle("ln(BMI)") title("All Cases*") name(gr2)

```

```

* Combinar diagramas de dispersión agrupados entre casos no imputados y entre
* todos los casos (Combine binned scatter plots among non-imputed cases and
* among all cases)
graph combine gr1 gr2, xcommon ycommon rows(1) title(`"20`yr'") ///
    note("**Imputed BMI's and imputed years of education are averaged
across 10 imputations")

* Guardar diagramas de dispersión (Save scatter plots)
graph save ``gph_out'\log_bmi_edu_bin`yr'.gph", replace

*** PASO 7: GUARDAR BASE DE DATOS FINAL ***
*** STEP 7: SAVE FINAL DATASET ***

*** Borrar variables no necesarias (Delete unnecessary variables) ***
keep cunichah subhog_`yr' np acthog codent`yr' imp_ht_`yr' imp_ht_sd_`yr' ///
    imp_wt_`yr' imp_wt_sd_`yr' imp_bmi_`yr' imp_bmi_sd_`yr' ///
    ht_miss_`yr' wt_miss_`yr'

*** Reetiquetar variables (Relabel variables) ***

* Identificadores de hogar e individuales (Household and individual identifiers)
label variable cunichah "Clave Unica del Hogar (=unhhid)"
label variable subhog_`yr' `"20`yr' sub-household identifier"
label variable np "Person Number/ Numero de Persona"
label variable acthog `"Updated household code 20`yr'"
label variable codent`yr' `"Person identification code 20`yr' (=ent2)"

* Valores imputados (promedios), valores observados y desviaciones estándares
* entre imputaciones
* (Average) imputed values, observed values, and standard deviations across im-
* putations
foreach i in `"'ht"'height without shoes (cm)"'height"' ///
    `"'wt"'weight (kg)"'weight"' ///
    `"'bmi"'body mass index (kg/m^2)"'BMI"' {
    tokenize `"'i"'

```

```

    label variable imp_`1'`yr' `"'Imputed/observed `2' 20`yr'"
    label variable imp_`1'_sd_`yr' `"'Between-imputation standard deviation of
`3' 20`yr'"
}

* Indicadores de imputación de altura y peso (Indicators of imputation of height
* and weight)
label variable ht_miss_`yr' `"'Height was imputed 20`yr'"
label variable wt_miss_`yr' `"'Weight was imputed 20`yr'"

*** Volver a aplicar etiquetas de valores (Reapply value labels) ***
label values subhog_`yr' clavesubhog
label values codent_`yr' codent_`yr'

*** Cargar otras variables importantes, por ejemplo tipo de entrevista y ***
*** peso de muestreo (Load other important variables, e.g. type of      ***
*** interview and sampling weight)
***
merge 1:1 cunicah np using `"'maestro'"', nogenerate ///
    assert(match using) keep(match) keepusing(unhhid UNHHID CUNICAH NP ///
    UNHHIDNP tipent_`yr' tamloc_01 factori_`yr' edad_`yr' sexo_`yr')

*** Guardar datos finales (Save final data) ***
save `"'output'\MHAS_20`yr'_Weight_and_Height_IMPUTED.dta"', replace

```

2012

```
*** ENASEM 2012 — IMPUTACIÓN DE ALTURA Y PESO AUTO-REPORTADOS Y DE ÍNDICE DE ***
*** MASA CORPORAL (IMC)
***
*** MHAS 2012 — IMPUTATION OF SELF-REPORTED HEIGHT AND WEIGHT AND OF BODY ***
*** MASS INDEX (BMI)
***

*** Versión agosto 2020 (August 2020 Version) ***

version 15
local yr "12"

*** PASO 1: DEFINIR UBICACIONES DE CARPETAS ***
*** STEP 1: DEFINE LOCATIONS OF FILES ***

*** Convertir los siguientes macros en los directorios donde se guardan las ***
*** petas descritas abajo (Convert the following macros to the directories ***
*** where the files described below are stored)
***

local maestro "||maestro||" // Archivo maestro más reciente de seguimiento
                                // (Most recent master follow-up file)

local maestro`yr' `'||maestro`yr'||'"
                                // Archivo maestro de seguimiento de 2012 (fuente de indicador de
                                // selección para medidas antropométricas)
                                // 2012 master follow-up file (Source of indicator of selection for
an-
                                // thropometric measurements)

local indiv "||individual||" // Bases de datos crudos al nivel individual
                                // (Individual-level databases)

local indiv01 "||individual_2001||"
                                // Datos al nivel individual de 2001 (Individual-level data from 2001)

local indiv03 "||individual_2003||"
                                // Datos al nivel individual de 2003 (Individual-level data from 2003)

local gph_out "||graph_output||"
                                // Figuras construidas en este programa (Figures constructed in this
```

```

        // program)
local output "||output||" // Base de datos final con medias y desviaciones
                                // estándares entre imputaciones (Final dataset
with
                                // means and standard deviations across
imputations)

*** Cargar datos brutos (Load raw data) ***
local maestro`yr' ``maestro`yr'\master_follow_up_file_20`yr'.dta"
use cunichah np tipent`yr' biomarkers`yr' if inlist(tipent`yr',1,2,3,4) ///
    using ``maestro`yr'","", clear

** Cargar otras variables para imputación (Load other variables for imputation)
local rmr "18" // Año de la ronda más reciente
local maestro ``maestro'\master_follow_up_file_20`rmr'.dta"
merge 1:1 cunichah np using ``maestro'", keepusing(subhog`yr' tipent`yr' ///
    tam_loc`yr' age`yr' sex`yr') nogenerate assert(match using) ///
    keep(match)

merge 1:1 cunichah np using ``indiv'\sect_a_c_d_e_pc_f_h_i_em_20`yr'.dta"', ///
    assert(match) nogenerate keepusing(c6`yr' c66`yr' ///
    c67_1`yr' c67_2`yr' yrschool`yr')
    // Educación para nuevas personas, altura y peso auto-reportados y
    // diabetes (Education for new persons, self-reported height and
weight,
    // and diabetes)
merge 1:1 cunichah np using ``indiv'\sect_1_biomarkers_20`yr'","", nogenerate ///
    assert(match master) keepusing(estatu1`yr' estatu2`yr' peso1`yr'
///
    peso2`yr') // Altura y peso medidos (Measured height and weight)

*** PASO 2: PREPARAR DATOS CRUDOS PARA IMPUTACIÓN ***
*** STEP 2: PREPARE RAW DATA FOR IMPUTATION ***

```

```

*** Crear una sola variable de altura auto-reportada (Create single self- ***
*** reported height variable)
***
generate c67_`yr' = c67_1_`yr' * 100 + c67_2_`yr'
        // Altura auto-reportada sin zapatos en centímetros (Self-reported
        // height without shoes in centimeters)

*** Crear variable por educación para todos los sujetos (Create education ***
*** variable for all subjects)
***

* Comenzar con datos sobre educación para sujetos nuevos en 2012
generate yrschool = yrschool_`yr' if inlist(tipent_`yr',2,4) & ///
        !inlist(yrschool_`yr',88,99)

* Cargar datos sobre educación de olas previas (Load data on education from pre-
* vious waves
tempfile a01 aa03
merge 1:1 cunicah np using ``maestro'', assert(match using) keep(match) ///
        nogenerate keepusing(unhhid codent01 acthog codent03)

preserve
use ``indiv01'\sect_a.dta'', clear
rename (ps3 a3_3) (codent01 a3_3_01)
replace a3_3_01 = . if a3_3_01 == 99
save `a01'
use ``indiv03'\sect_aa.dta''
rename(ent2 aa4a_3) (codent03 aa4a_3_03)
replace aa4a_3_03 = . if aa4a_3_03 == 99
destring acthog, replace
save `aa03'
restore

merge m:1 unhhid codent01 using `a01', keep(match master) keepusing(a3_3_01) ///
        generate(merge01)

```

```

merge m:1 cunicah acthog codent03 using `aa03', keep(match master) ///
      keepusing(aa4a_3_03) generate(merge03)

* Construir variable yrschool (Construct variable yrschool)
replace yrschool = aa4a_3_03 if merge03 == 3 & missing(yrschool) & ///
      !missing(aa4a_3_03)
replace yrschool = a3_3_01 if merge01 == 3 & missing(yrschool) & ///
      !missing(a3_3_01)

*** Computar medias de alturas y pesos medidos (Compute averages of measured ***
*** heights and weights)
***
generate estatu_`yr' = (estatu1_`yr' + estatu2_`yr') / 2 if biomarkers_`yr' == 1
generate peso_`yr' = (peso1_`yr' + peso2_`yr') / 2 if biomarkers_`yr' == 1

*** Sustituir valores que indican datos faltantes por . (Replace values that ***
*** indicate missing data with .)
***
replace c66_`yr' = . if inlist(c66_`yr',888,999)
replace c67_`yr' = . if inlist(c67_1_`yr',8,9)
replace peso_`yr' = . if peso1_`yr' == 995 | peso2_`yr' == 995
replace age_`yr' = . if age_`yr' == 999

*** Sustituir alturas y pesos auto-reportados que no emparejan con valores ***
*** medidos por . (Replace self-reported heights and weights that do not ***
*** match measured values with .)
***
replace c66_`yr' = . if biomarkers_`yr' == 1 & !(missing(c66_`yr') | ///
      missing(peso_`yr')) & abs((c66_`yr' - peso_`yr') / peso_`yr') > 0.1
replace c67_`yr' = . if biomarkers_`yr' == 1 & !(missing(c67_`yr') | ///
      missing(estatu_`yr')) & abs((c67_`yr' - estatu_`yr') / estatu_`yr')
///
      > 0.1

*** Sustituir alturas reportados en 2012 que no emparejan con la altura ***
*** reportada en cualquier ola previa por . (Replace heights reported ***

```

```

*** in 2012 that do not match the height reported in any previous wave ***
*** with .)
***
drop merge01 merge03
tempfile c01 c03
preserve
use ``indiv01'\sect_c.dta'', clear
rename (ps3 c72) (codent01 c72_01)
save `c01'
use ``indiv03'\sect_c.dta''
rename (ent2 c67) (codent03 c67_03)
destring acthog, replace
save `c03'
restore

merge m:1 unhhid codent01 using `c01', keep(match master) keepusing(c72_01) ///
      generate(merge01)
merge m:1 cunicah acthog codent03 using `c03', keep(match master) ///
      keepusing(c67_03) generate(merge03)
replace c67_`yr' = . if (merge01 == 3 & !(missing(c72_01) | ///
      missing(c67_`yr') | inrange(c72_01,888,999)) & abs((c67_`yr' - ///
      c72_01) / c72_01) > 0.1) | (merge03 == 3 & !(missing(c67_03) | ///
      missing(c67_`yr') | inlist(c67_03,888,999)) & abs((c67_`yr' - ///
      c67_03) / c67_03) > 0.1)

*** Crear indicadores de alturas y pesos faltantes (Create indicator of mis- ***
*** sing heights and weights)
***
generate byte ht_miss_`yr' = missing(c67_`yr')
generate byte wt_miss_`yr' = missing(c66_`yr')

*** Sustituir alturas y pesos medidos que faltan porque los sujetos ***
*** correspondientes no fueron seleccionados para medidas por 0 ***
*** antropometricas (Substitute 0's for height and weight measure- ***
*** ments that are missing because subject was not selected for an- ***

```

```

*** thropometric measurements)
***

replace peso_`yr' = 0 if biomarkers_`yr' == 0
replace estatu_`yr' = 0 if biomarkers_`yr' == 0


*** PASO 3: IMPUTAR DATOS FALTANTES ***
*** STEP 3: IMPUTE MISSING DATA ***


*** Definir parámetros de imputación (cambiar los siguientes macros para ***
*** modificar las imputaciones)
***
*** Define parameters of imputation (change the following macros to modify ***
*** imputations)
***

local n_imp = 10 // Número de imputaciones (Number of imputations)
local n_iter = 300 // Número de iteraciones en imputación multivariada usanda
                    // ecuaciones ligadas
                    // Number of iterations in multivariate imputation using
                    // chained equations (MICE)

local k = 5 // Número de "vecinos más cercanos" para emparejar medias predictivas
            // Number of "nearest neighbors" for predictive mean matching
(PMM)

local rng_seed = 101 // "Semilla" para generar números aleatorios ("Seed" to
                    // generate random numbers)


*** Declarar base de datos como "datos de imputación" y variables como ***
*** "imputadas" o "regulares" (ningún valor se imputará) ***
*** Declare database as "imputation data" and variables as "imputed or ***
*** "regular" (no values will be imputed)
***

mi set wide

mi register imputed c66_`yr' c67_`yr' peso_`yr' estatu_`yr' age_`yr' yrschool
mi register regular tam_loc_`yr' sex_`yr' subhog_`yr' cunicah np ///
            ht_miss_`yr' wt_miss_`yr' biomarkers_`yr' c6_`yr'

```

```

*** Imputar valores faltantes (Impute missing values) ***
mi impute chained (pmm, knn(`k')) c66_`yr' c67_`yr' age_`yr' yrschool (pmm, ///
    knn(`k') conditional(if biomarkers_`yr' == 1)) peso_`yr' estatu_`yr'
///
    = i.sex_`yr' i.tam_loc_`yr', burnin(`n_iter') rseed(`rng_seed') ///
    add(`n_imp')

*** Crear variables pasivas (definidas como funciones de variables ***
*** imputadas)
***

*** Create passive variables (defined as functions of imputed ***
*** variables) ***

mi passive: generate bmi_`yr' = c66_`yr' / ((c67_`yr' / 100)^2)
mi passive: generate log_bmi_`yr' = log(bmi_`yr')

*** PASO 4: ESTIMAR CONEXIONES MARGINALES ENTRE EL LOGARITMO DE ÍMC Y ***
*** EDUCACIÓN, TAMAÑO DE LOCALIDAD Y DIABETES USANDO DATOS IMPUTADOS ***
*** STEP 4: ESTIMATE MARGINAL ASSOCIATIONS BETWEEN LOG-BMI AND EDUCATION, ***
*** LOCALITY SIZE, AND DIABETES USING IMPUTED DATA
***

mi estimate: regress log_bmi_`yr' yrschool // Educación
mi estimate: regress log_bmi_`yr' ib4.tam_loc_`yr'
    // Tamaño de localidad (Locality size)
mi estimate: regress log_bmi_`yr' ib2.c6_`yr' if inlist(c6_`yr',1,2) // Diabetes

*** PASO 5: PLEGAR BASE DE DATOS EN ESTADÍSTICAS SUMARIAS NECESARIAS PARA ***
*** BASE DE DATOS FINAL Y PARA TABLAS Y FIGURAS
***

*** STEP 5: COLLAPSE DATASET INTO SUMMARY STATISTICS NECESSARY FOR FINAL ***
*** DATASET AND FOR TABLES AND FIGURES
***

mi convert mlong, clear
local yr "12"
collapse (mean) imp_ht_`yr'=c67_`yr' imp_wt_`yr'=c66_`yr' ///
    imp_bmi_`yr'=bmi_`yr' log_bmi_mean_`yr'=log_bmi_`yr' ///

```

```

yrschool_mean=yrschool (sd) imp_ht_sd`yr`=c67`yr' ///
imp_wt_sd`yr`=c66`yr' imp_bmi_sd`yr`=bmi`yr' ///
yrschool_sd=yrschool (median) bmi_med`yr`=bmi`yr' (firstnm) ///
subhog`yr' _mi_m _mi_id _mi_miss ht_miss`yr' wt_miss`yr' c6`yr'
///

tam_loc`yr', by(cunichah np)

mi extract 0, clear // Stata ya no reconoce estos datos como datos de
                    // imputación. (Stata no longer recognizes these data
as im-
                    // putation data.)

*** Sustituir desviaciones estándares por 0 en observaciones completas (Re- ***
*** place standard deviations with 0 in complete observations)
***
replace imp_ht_sd`yr' = 0 if ht_miss`yr' == 0
replace imp_wt_sd`yr' = 0 if wt_miss`yr' == 0
replace imp_bmi_sd`yr' = 0 if ht_miss`yr' == 0 & wt_miss`yr' == 0

*** Reetiquetar variables que se les enviarán a usuarios (Relabel variables ***
*** that will be sent to users)
***
label variable imp_ht`yr' "Imputed/observed height without shoes (cm) 20`yr'"
label variable imp_wt`yr' "Imputed/observed weight (kg) 20`yr'"
label variable imp_bmi`yr' "Imputed/observed BMI (kg/m^2) 20`yr'"
label variable imp_ht_sd`yr' "Between-imputation standard deviation of height
20`yr'"
label variable imp_wt_sd`yr' "Between-imputation standard deviation of weight
20`yr'"
label variable imp_bmi_sd`yr' "Between-imputation standard deviation of BMI
20`yr'"
label variable subhog`yr' "20`yr' sub-household identifier"
label variable ht_miss`yr' "Height was imputed 20`yr'"
label variable wt_miss`yr' "Weight was imputed 20`yr'"

*** Volver a aplicar etiquetas de valores (Reapply value labels) ***
label values subhog`yr' clavesubhog

```

```

*** Agregarle notas sobre valores imputados a base de datos (Add note about ***
*** imputed values to dataset)
***
notes: Values of imp_ht_`yr' and imp_wt_`yr' where height and weight were imputed
(ht_miss_`yr' = 1 and wt_miss_`yr' = 1, respectively) were averaged across 10
imputations.

notes: Values of imp_bmi_`yr' where BMI was imputed (ht_miss_`yr' = 1 and/or
wt_miss_`yr' = 1) were averaged across 10 imputations.

*** Agregarle nota sobre desviaciones estándares entre imputaciones a base ***
*** de datos (Add note about between-imputation standard deviations to ***
*** dataset)
***
notes: imp_ht_sd_`yr', imp_wt_sd_`yr', and imp_bmi_sd_`yr' are 0 where height,
weight, and BMI, respectively, were observed (i.e., ht_miss_`yr' = 0,
wt_miss_`yr' = 0, and both ht_miss_`yr' and wt_miss_`yr' are 0, respectively)
because observed values are constant across imputations.

*** PASO 6: COMPARAR RESULTADOS DE ANALIZAR DATOS IMPUTADOS CON LOS DE ***
*** ANALIZAR DATOS BRUTOS
***
*** STEP 6: COMPARE RESULTS OF ANALYZING IMPUTED DATA WITH THOSE FROM ANALY- ***
*** ZING RAW DATA
***

*** Estimar conexiones marginales entre el logaritmo de ÍMC y educación, ***
*** tamaño de localidad y diabetes usando datos brutos (Estimate margi- ***
*** nal associations between log-BMI and education, locality size, and ***
*** diabetes using raw data
***

regress log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & wt_miss_`yr' ///
      == 0 & (yrschool_sd == 0 | missing(yrschool_sd))
regress log_bmi_mean_`yr' ib4.tam_loc_`yr' if ht_miss_`yr' == 0 & ///
      wt_miss_`yr' == 0
regress log_bmi_mean_`yr' ib2.c6_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == ///
      0 & inlist(c6_`yr',1,2)

*** Computar estadísticas sumarias y percentiles de alturas y peso auto- ***

```

```

*** reportados entre observaciones donde fueron observados, entre las ***
*** donde fueron imputados y entre todas las observaciones (Compute sum- ***
*** mary statistics and percentiles of self-reported heights and weights ***
*** among observations where they were observed, among those where they ***
*** were imputed, and among all observations)
***

foreach i in ht wt {
    summarize imp_`i'`_yr' if `i'`_miss_`yr' == 0, detail
    summarize imp_`i'`_yr' if `i'`_miss_`yr' == 1, detail
    summarize imp_`i'`_yr', detail
}

*** Construir histogramas de alturas y pesos auto-reportados entre ***
*** observaciones donde fueron observados, entre las donde fueron ***
*** imputados y entre todas las observaciones (Construct histograms ***
*** of self-reported heights and weights among observations where ***
*** they were observed, among those where they were imputed, and ***
*** among all observations)
***

graph drop _all

* Altura (Height)
histogram imp_ht_`yr' if ht_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr1)
histogram imp_ht_`yr' if ht_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases*") name(gr2) ytitle("")
histogram imp_ht_`yr', xtitle("") title("All Cases*") name(gr3) ///
    ytitle("")
graph combine gr1 gr2 gr3, xcommon ycommon title("Height (cm)", position(6)) ///
    rows(1) name(gr_ht) // Combina histogramas de altura en una figura
(Com-
                                     // bine histograms of height into one
figure)

* Peso (Weight)
histogram imp_wt_`yr' if wt_miss_`yr' == 0, xtitle("") ///

```

```

        title("Non-Imputed Cases") name(gr4)
    histogram imp_wt_`yr' if wt_miss_`yr' == 1, xtitle("") ///
        title("Imputed Cases*") name(gr5) ytitle("")
    histogram imp_wt_`yr', xtitle("") title("All Cases*") name(gr6) ytitle("")
    graph combine gr4 gr5 gr6, xcommon ycommon title("Weight (kg)", position(6)) ///
        rows(1) name(gr_wt) // Combina histogramas de peso en una figura (Com-
        // bine histograms of weight into one
    figure)

* Combinar histogramas de altura con histogramas de peso (Combine histograms of
* height with hitsograms of weigth)
graph combine gr_ht gr_wt, cols(1) title(`"20`yr'") ///
    note("*Imputed values in each observation are averaged across 10
    imputations")

* Guardar histogramas (Save histograms)
graph save ``gph_out'\ht_wt_hist`yr'.gph'', replace

*** Construir diagramas de caja de ÍMC por tamaño de localidad entre ***
*** observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para ***
*** observaciones donde ÍMC fue imputado (Construct box plots of BMI ***
*** by locality size among observations where BMI was observed and ***
*** among all observations, using medians across imputations for ob- ***
*** servations where BMI was imputed) ***
graph drop _all

* Casos no imputados (Non-imputed cases)
graph hbox bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0, ///
    over(tam_loc_`yr') title("Non-Imputed Cases") ///
    ytitle("BMI (kg/m2)") name(gr1)

* Todos los casos (All cases)
graph hbox bmi_med_`yr', over(tam_loc_`yr') title("All Cases*") ///
    ytitle("Median BMI (kg/m2)") name(gr2)

```

```

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'`) ycommon ///
    note("*BMI's for imputed cases are medians among 10 imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_tam_loc_box`yr'.gph", replace

*** Construir diagramas de caja de ÍMC por presencia o ausencia de diabetes ***
*** entre observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para observaciones ***
*** donde ÍMC fue imputado y excluyendo observaciones donde faltan datos ***
*** sobre diabetes (Construct box plots of BMI by presence or absence of ***
*** diabetes among observations where BMI was observed and among all obser- ***
*** vations, using medians across imutations for observations wher BMI was ***
*** imputed and excluding observations where data on diabetes are missing) ***
graph drop _all

* Casos no imputados (Non-imputed cases)
graph box bmi_med`yr' if ht_miss`yr' == 0 & wt_miss`yr' == 0 & ///
    inlist(c6`yr',1,2), over(c6`yr', relabel(1 "Diabetic" 2 ///
    "Not Diabetic")) title("Non-Imputed Cases{superscript:1}") ///
    ytitle("BMI (kg/m{superscript:2})") name(gr1)

* Todos los casos (All cases)
graph box bmi_med`yr' if inlist(c6`yr',1,2), over(c6`yr', relabel(1 ///
    "Diabetic" 2 "Not Diabetic")) title("All Cases{superscript:1,2}") ///
    ytitle("Median BMI (kg/m{superscript:2})") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'`) ycommon ///
    note("{superscript:1}Excludes observations where diabetic status is
missing" ///

```

```
"2BMI's for imputed cases are medians among 10
imputations")
```

```
* Guardar diagramas de caja (Save box plots)
```

```
graph save "`gph_out'\bmi_diab_box`yr'.gph"', replace
```

```
*** Construir diagramas de dispersión "agrupados" de logaritmos de ÍMC por ***
*** años de educación entre observaciones donde ÍMC fue observado y entre ***
*** todas las observaciones (Construct "binned" scatter plots of log-BMI ***
*** vs. years of education among observations where BMI was observed and ***
*** among all observations)
***
```

```
graph drop _all
```

```
* Casos no imputados (Non-imputed cases)
```

```
binscatter log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & ///
           wt_miss_`yr' == 0, xtitle("Years of education") ytitle("ln(BMI)") ///
           title("Non-Imputed Cases") name(gr1)
```

```
* Todos los casos (All cases)
```

```
binscatter log_bmi_mean_`yr' yrschool_mean, xtitle("Years of education") ///
           ytitle("ln(BMI)") title("All Cases*") name(gr2)
```

```
* Combinar diagramas de dispersión agrupados entre casos no imputados y entre
* todos los casos (Combine binned scatter plots among non-imputed cases and
* among all cases)
```

```
graph combine gr1 gr2, xcommon ycommon rows(1) title("`20`yr'") ///
```

```
           note("*Imputed BMI's and imputed years of education are averaged
across 10 imputations")
```

```
* Guardar diagramas de dispersión (Save scatter plots)
```

```
graph save "`gph_out'\log_bmi_edu_bin`yr'.gph"', replace
```

```
*** PASO 7: GUARDAR BASE DE DATOS FINAL ***
```

```
*** STEP 7: SAVE FINAL DATASET ***
```

```

*** Borrar variables no necesarias (Delete unnecessary variables) ***
keep cunicalh np subhog_`yr' tam_loc_`yr' imp_ht_`yr' imp_ht_sd_`yr' ///
    imp_wt_`yr' imp_wt_sd_`yr' imp_bmi_`yr' imp_bmi_sd_`yr' ///
    ht_miss_`yr' wt_miss_`yr'

*** Cargar otras variables importantes, por ejemplo tipo de entrevista y ***
*** peso de muestreo (Load other important variables, e.g. type of ***
*** interview and sampling weight)
***
merge 1:m cunicalh np using ``maestro'', nogenerate assert(match using) ///
    keep(match) keepusing(UNHHID CUNICALH NP UNHHIDNP tipent_`yr' ///
    tam_loc_`yr' eam_`yr' factori_`yr' age_`yr' sex_`yr')

*** Guardar datos finales (Save final data) ***
save ``output'\MHAS_20`yr'_Weight_and_Height_IMPUTED.dta'', replace

```

2015

```
*** ENASEM 2015 — IMPUTACIÓN DE ALTURA Y PESO AUTO-REPORTADOS Y DE ÍNDICE DE ***
*** MASA CORPORAL (IMC)
***
*** MHAS 2015 — IMPUTATION OF SELF-REPORTED HEIGHT AND WEIGHT AND OF BODY ***
*** MASS INDEX (BMI)
***
*** Versión octubre 2020 (October 2020 Version) ***

version 15
local yr "15"

*** PASO 1: DEFINIR UBICACIONES DE CARPETAS ***
*** STEP 1: DEFINE LOCATIONS OF FILES ***

*** Convertir los siguientes macros en los directorios donde se guardan las ***
*** petas descritas abajo (Convert the following macros to the directories ***
*** where the files described below are stored)
***

local maestro "||maestro||" // Archivo maestro de seguimiento (Most recent mas-
                           // ter follow-up file)
local indiv  "||individual||" // Bases de datos crudos al nivel individual
                           // (Individual-level databases)
local indiv01 "||individual_2001||"
               // Datos al nivel individual de 2001 (Individual-level data from 2001)
local indiv03 "||individual_2003||"
               // Datos al nivel individual de 2003 (Individual-level data from 2003)
local indiv12 "||individual_2012||"
               // Datos al nivel individual de 2012 (Individual-level data from 2012)
local gph_out "||graph_output||"
               // Figuras construidas en este programa (Figures constructed in this
               // program)
local output "||output||" // Base de datos final con medias y desviaciones
                           // estándares entre imputaciones (Final dataset
with
                           // means and standard deviations across
imputations)

*** Cargar datos brutos (Load raw data) ***
local rmr "18" // Año de la ronda más reciente (Year of most recent wave)
```

```

local maestro "`maestro'\master_follow_up_file_20`rmr'.dta'"
use cunichah np subhog`yr' tipent`yr' tam_loc`yr' age`yr' sex`yr' if ///
    inlist(tipent`yr',1,2,3,4) using "`maestro'", clear
    // Se recomienda que se usa el archivo maestro de seguimiento
disponible
    // más reciente. (It is recommended that the most recent available
mas-
    // ter follow-up file is used.)

** Cargar otras variables para imputación (Load other variables for imputation)
local indiv`yr' "`indiv'\sect_a_c_d_e_pc_f_h_i_20`yr'.dta'"
    // Directorio completo de base de datos al nivel individual de la
ronda
    // actual (Complete directory of individual-level database from
current
    // wave)
merge 1:1 cunichah np using "`indiv`yr'", assert(match) nogenerate ///
    keepusing(c6`yr' c66`yr' c67_1`yr' c67_2`yr' yrschool)
    // Educación para nuevas personas, altura y peso auto-reportados y
    // diabetes (Education for new persons, self-reported height and
weight,
    // and diabetes)
local antro`yr' "`indiv'\Anthropometrics_and_Biomarkers_Mex_Cog_2016.dta'"
    // Base de datos con medidas antropométricas de la ronda actual (Data-
    // base with anthropometric measurements from current wave)
merge 1:1 cunichah np using "`antro`yr'", assert(match master) ///
    generate(anth_meas_16) keepusing(ma_q7_16 ma_q7_1_16 ma_q8_16
ma_q8_1_16)
    // Altura y peso medidos (Measured height and weight)

*** PASO 2: PREPARAR DATOS CRUDOS PARA IMPUTACIÓN ***
*** STEP 2: PREPARE RAW DATA FOR IMPUTATION ***

*** Crear una sola variable de altura auto-reportada (Create single self- ***
*** reported height variable)
***
generate c67`yr' = c67_1`yr' * 100 + c67_2`yr'
    // Altura auto-reportada sin zapatos en centímetros (Self-reported
    // height without shoes in centimeters)

```

```

*** Crear variable de educación para todos los sujetos (Create education ***
*** variable for all subjects)
***

* Cargar datos sobre educación de olas previas (Load data on education from pre-
* vious waves
tempfile a01 aa03
merge 1:1 cunichah np using ``maestro'', assert(match using) keep(match) ///
        nogenerate keepusing(unhhid codent01 acthog codent03)

preserve
use ``indiv01'\sect_a.dta'', clear
rename (ps3 a3_3) (codent01 a3_3_01)
save `a01'
use ``indiv03'\sect_aa.dta''
rename(ent2 aa4a_3) (codent03 aa4a_3_03)
destring acthog, replace
save `aa03'
restore

merge m:1 unhhid codent01 using `a01', keep(match master) keepusing(a3_3_01) ///
        generate(merge01)
merge m:1 cunichah acthog codent03 using `aa03', keep(match master) ///
        keepusing(aa4a_3_03) generate(merge03)
local indiv12 ``indiv12'\sect_a_c_d_e_pc_f_h_i_em_2012.dta''
merge 1:1 cunichah np using ``indiv12'', generate(merge12) keep(match ///
        master) keepusing(yrschool_12 c67_1_12 c67_2_12)

* Construir variable yrschool (Construct variable yrschool)
replace yrschool = . if inlist(yrschool,88,99,.m)
replace yrschool = yrschool_12 if merge12 == 3 & missing(yrschool) & ///
        !(missing(yrschool_12) | inlist(yrschool_12,88,99))
replace yrschool = aa4a_3_03 if merge03 == 3 & missing(yrschool) & ///
        !(missing(aa4a_3_03) | aa4a_3_03 == 99)
replace yrschool = a3_3_01 if merge01 == 3 & missing(yrschool) & !(a3_3_01 ///
        == 99 | missing(a3_3_01))

```

```

*** Computar medias de alturas y pesos medidos (Compute averages of measured ***
*** heights and weights)
***
generate estatu_16 = (ma_q7_16 + ma_q7_1_16) / 2 if anth_meas_16 == 3
generate peso_16 = (ma_q8_16 + ma_q8_1_16) / 2 if anth_meas_16 == 3

*** Sustituir valores que indican datos faltantes y valores faltantes ***
*** extendidos (i.e., .a, .b,..., .z) por . ***
*** Replace values that indicate missing data and "extended" missing ***
*** values (i.e., .a, .b,..., .z) with . ***
replace c66_`yr' = . if inlist(c66_`yr',888,999,.i)
replace c67_`yr' = . if inlist(c67_1_`yr',9,888,999,.i)
replace estatu_16 = . if inrange(ma_q7_16,995,999) | inlist(ma_q7_16,.r,.s) ///
                    | inrange(ma_q7_1_16,995,999) | inlist(ma_q7_1_16,.r,.s)
replace peso_16 = . if inrange(ma_q8_16,995,999) | inlist(ma_q8_16,.r,.s) ///
                    | inrange(ma_q8_1_16,995,999) | inlist(ma_q8_1_16,.r,.s)
replace age_`yr' = . if age_`yr' == 999

*** Sustituir alturas y pesos auto-reportados que no emparejan con valores ***
*** medidos por . (Replace self-reported heights and weights that do not ***
*** match measured values with .)
***
replace c66_`yr' = . if anth_meas_16 == 3 & !(missing(c66_`yr') | ///
                    missing(peso_16)) & abs((c66_`yr' - peso_16) / peso_16) > 0.1
replace c67_`yr' = . if anth_meas_16 & !(missing(c67_`yr') | ///
                    missing(estatu_16)) & abs((c67_`yr' - estatu_16) / estatu_16) ///
                    > 0.1

*** Sustituir alturas reportados en 2012 que no emparejan con la altura ***
*** reportada en cualquier ola previa por . (Replace heights reported ***
*** in 2012 that do not match the height reported in any previous wave ***
*** with .)
***
drop merge01 merge03
tempfile c01 c03
preserve
use ``indiv01'\sect_c.dta"', clear
rename (ps3 c72) (codent01 c72_01)
save `c01'

```

```

use ``indiv03'\sect_c.dta''
rename (ent2 c67) (codent03 c67_03)
destring acthog, replace
save `c03'
restore

merge m:1 unhhid codent01 using `c01', keep(match master) keepusing(c72_01) ///
    generate(merge01)
merge m:1 cunicah acthog codent03 using `c03', keep(match master) ///
    keepusing(c67_03) generate(merge03)
generate c67_12 = c67_1_12 * 100 + c67_2_12
replace c67_12 = . if inlist(c67_1_12,8,9)
replace c67_`yr' = . if (merge01 == 3 & !(missing(c72_01) | ///
    missing(c67_`yr') | inrange(c72_01,888,999)) & abs((c67_`yr' - ///
    c72_01) / c72_01) > 0.1) | (merge03 == 3 & !(missing(c67_03) | ///
    missing(c67_`yr') | inlist(c67_03,888,999)) & abs((c67_`yr' - ///
    c67_03) / c67_03) > 0.1) | (merge12 == 3 & !(missing(c67_12) | ///
    missing(c67_`yr')) & abs((c67_`yr' - c67_12) / c67_12) > 0.1)

*** Crear indicadores de alturas y pesos faltantes (Create indicator of mis- ***
*** sing heights and weights)
***
generate byte ht_miss_`yr' = missing(c67_`yr')
generate byte wt_miss_`yr' = missing(c66_`yr')

*** Sustituir alturas y pesos medidos que faltan porque los sujetos ***
*** correspondientes no fueron seleccionados para medidas por 0 ***
*** antropometricas (Substitute 0's for height and weight measure- ***
*** ments that are missing because subject was not selected for an- ***
*** thropometric measurements)
***
replace peso_16 = 0 if anth_meas_16 != 3
replace estatu_16 = 0 if anth_meas_16 != 3

*** PASO 3: IMPUTAR DATOS FALTANTES ***
*** STEP 3: IMPUTE MISSING DATA ***

*** Definir parámetros de imputación (cambiar los siguientes macros para ***

```

```

*** modificar las imputaciones)
***

*** Define parameters of imputation (change the following macros to modify ***
*** imputations)
***

local n_imp = 10 // Número de imputaciones (Number of imputations)
local n_iter = 300 // Número de iteraciones en imputación multivariada usanda
                    // ecuaciones ligadas
                    // Number of iterations in multivariate imputation using
                    // chained equations (MICE)

local k = 5 // Número de "vecinos más cercanos" para emparejar medias predictivas
            // Number of "nearest neighbors" for predictive mean matching
(PMM)

local rng_seed = 101 // "Semilla" para generar números aleatorios ("Seed" to
                    // generate random numbers)

*** Declarar base de datos como "datos de imputación" y variables como ***
*** "imputadas" o "regulares" (ningún valor se imputará) ***
*** Declare database as "imputation data" and variables as "imputed or ***
*** "regular" (no values will be imputed)
***

mi set wide

mi register imputed c66_`yr' c67_`yr' peso_16 estatu_16 age_`yr' yrschool
mi register regular tam_loc_`yr' sex_`yr' subhog_`yr' cunich np ///
            ht_miss_`yr' wt_miss_`yr' anth_meas_16 c6_`yr'

*** Imputar valores faltantes (Impute missing values) ***
mi impute chained (pmm, knn(`k')) c66_`yr' c67_`yr' age_`yr' yrschool (pmm, ///
            knn(`k') conditional(if anth_meas_16 == 3)) peso_16 estatu_16 = ///
            i.sex_`yr' i.tam_loc_`yr', burnin(`n_iter') rseed(`rng_seed') ///
            add(`n_imp')

*** Crear variables pasivas (definidas como funciones de variables ***
*** imputadas)
***

*** Create passive variables (defined as functions of imputed ***
*** variables) ***

mi passive: generate bmi_`yr' = c66_`yr' / ((c67_`yr' / 100)^2)
mi passive: generate log_bmi_`yr' = log(bmi_`yr')

```

```

*** PASO 4: ESTIMAR CONEXIONES MARGINALES ENTRE EL LOGARITMO DE ÍMC Y ***
*** EDUCACIÓN, TAMAÑO DE LOCALIDAD Y DIABETES USANDO DATOS IMPUTADOS ***
*** STEP 4: ESTIMATE MARGINAL ASSOCIATIONS BETWEEN LOG-BMI AND EDUCATION, ***
*** LOCALITY SIZE, AND DIABETES USING IMPUTED DATA ***

mi estimate: regress log_bmi_`yr' yrschool // Educación
mi estimate: regress log_bmi_`yr' ib4.tam_loc_`yr'
              // Tamaño de localidad (Locality size)
mi estimate: regress log_bmi_`yr' ib2.c6_`yr' if inlist(c6_`yr',1,2) // Diabetes

*** PASO 5: PLEGAR BASE DE DATOS EN ESTADÍSTICAS SUMARIAS NECESARIAS PARA ***
*** BASE DE DATOS FINAL Y PARA TABLAS Y FIGURAS ***
*** STEP 5: COLLAPSE DATASET INTO SUMMARY STATISTICS NECESSARY FOR FINAL ***
*** DATASET AND FOR TABLES AND FIGURES ***

mi convert mlong, clear
collapse (mean) imp_ht_`yr'=c67_`yr' imp_wt_`yr'=c66_`yr' ///
          imp_bmi_`yr'=bmi_`yr' log_bmi_mean_`yr'=log_bmi_`yr' ///
          yrschool_mean=yrschool (sd) imp_ht_sd_`yr'=c67_`yr' ///
          imp_wt_sd_`yr'=c66_`yr' imp_bmi_sd_`yr'=bmi_`yr' ///
          yrschool_sd=yrschool (median) bmi_med_`yr'=bmi_`yr' (firstnm) ///
          subhog_`yr' _mi_m _mi_id _mi_miss ht_miss_`yr' wt_miss_`yr' c6_`yr'
///
          tam_loc_`yr', by(cunicah np)
mi extract 0, clear // Stata ya no reconoce estos datos como datos de
                    // imputación. (Stata no longer recognizes these data
as im-
                    // putation data.)

*** Sustituir desviaciones estándares por 0 en observaciones completas (Re- ***
*** place standard deviations with 0 in complete observations) ***
***
replace imp_ht_sd_`yr' = 0 if ht_miss_`yr' == 0
replace imp_wt_sd_`yr' = 0 if wt_miss_`yr' == 0
replace imp_bmi_sd_`yr' = 0 if ht_miss_`yr' == 0 & wt_miss_`yr' == 0

*** Reetiquetar variables que se les enviarán a usuarios (Relabel variables ***

```

```

*** that will be sent to users)
***

label variable imp_ht_`yr' "Imputed/observed height without shoes (cm) 20`yr'"
label variable imp_wt_`yr' "Imputed/observed weight (kg) 20`yr'"
label variable imp_bmi_`yr' "Imputed/observed BMI (kg/m^2) 20`yr'"
label variable imp_ht_sd_`yr' "Between-imputation standard deviation of height
20`yr'"
label variable imp_wt_sd_`yr' "Between-imputation standard deviation of weight
20`yr'"
label variable imp_bmi_sd_`yr' "Between-imputation standard deviation of BMI
20`yr'"
label variable subhog_`yr' "20`yr' sub-household identifier"
label variable ht_miss_`yr' "Height was imputed 20`yr'"
label variable wt_miss_`yr' "Weight was imputed 20`yr'"

*** Volver a aplicar etiquetas de valores (Reapply value labels) ***
label values subhog_`yr' clavesubhog

*** Agregarle notas sobre valores imputados a base de datos (Add note about ***
*** imputed values to dataset)
***
notes: Values of imp_ht_`yr' and imp_wt_`yr' where height and weight were imputed
(ht_miss_`yr' = 1 and wt_miss_`yr' = 1, respectively) were averaged across 10
imputations.
notes: Values of imp_bmi_`yr' where BMI was imputed (ht_miss_`yr' = 1 and/or
wt_miss_`yr' = 1) were averaged across 10 imputations.

*** Agregarle nota sobre desviaciones estándares entre imputaciones a base ***
*** de datos (Add note about between-imputation standard deviations to ***
*** dataset)
***
notes: imp_ht_sd_`yr', imp_wt_sd_`yr', and imp_bmi_sd_`yr' are 0 where height,
weight, and BMI, respectively, were observed (i.e., ht_miss_`yr' = 0,
wt_miss_`yr' = 0, and both ht_miss_`yr' and wt_miss_`yr' are 0, respectively)
because observed values are constant across imputations.

*** PASO 6: COMPARAR RESULTADOS DE ANALIZAR DATOS IMPUTADOS CON LOS DE ***
*** ANALIZAR DATOS BRUTOS
***
*** STEP 6: COMPARE RESULTS OF ANALYZING IMPUTED DATA WITH THOSE FROM ANALY- ***
*** ZING RAW DATA
***

```

```

*** Estimar conexiones marginales entre el logaritmo de ÍMC y educación, ***
*** tamaño de localidad y diabetes usando datos brutos (Estimate margi- ***
*** nal associations between log-BMI and education, locality size, and ***
*** diabetes using raw data
***
regress log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & wt_miss_`yr' ///
      == 0 & (yrschool_sd == 0 | missing(yrschool_sd))
regress log_bmi_mean_`yr' ib4.tam_loc_`yr' if ht_miss_`yr' == 0 & ///
      wt_miss_`yr' == 0
regress log_bmi_mean_`yr' ib2.c6_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == ///
      0 & inlist(c6_`yr',1,2)

*** Computar estadísticas sumarias y percentiles de alturas y peso auto- ***
*** reportados entre observaciones donde fueron observados, entre las ***
*** donde fueron imputados y entre todas las observaciones (Compute sum- ***
*** mary statistics and percentiles of self-reported heights and weights ***
*** among observations where they were observed, among those where they ***
*** were imputed, and among all observations)
***
foreach i in ht wt {
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 0, detail
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 1, detail
    summarize imp_`i'_`yr', detail
}

*** Construir histogramas de alturas y pesos auto-reportados entre ***
*** observaciones donde fueron observados, entre las donde fueron ***
*** imputados y entre todas las observaciones (Construct histograms ***
*** of self-reported heights and weights among observations where ***
*** they were observed, among those where they were imputed, and ***
*** among all observations)
***
graph drop _all

* Altura (Height)
histogram imp_ht_`yr' if ht_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr1)
histogram imp_ht_`yr' if ht_miss_`yr' == 1, xtitle("") ///

```

```

        title("Imputed Cases*") name(gr2) ytitle("")
histogram imp_ht_`yr', xtitle("") title("All Cases*") name(gr3) ///
        ytitle("")
graph combine gr1 gr2 gr3, xcommon ycommon title("Height (cm)", position(6)) ///
        rows(1) name(gr_ht) // Combina histogramas de altura en una figura
(Com-
                                // bine histograms of height into one
figure)

* Peso (Weight)
histogram imp_wt_`yr' if wt_miss_`yr' == 0, xtitle("") ///
        title("Non-Imputed Cases") name(gr4)
histogram imp_wt_`yr' if wt_miss_`yr' == 1, xtitle("") ///
        title("Imputed Cases*") name(gr5) ytitle("")
histogram imp_wt_`yr', xtitle("") title("All Cases*") name(gr6) ytitle("")
graph combine gr4 gr5 gr6, xcommon ycommon title("Weight (kg)", position(6)) ///
        rows(1) name(gr_wt) // Combina histogramas de peso en una figura (Com-
                                // bine histograms of weight into one
figure)

* Combinar histogramas de altura con histogramas de peso (Combine histograms of
* height with hitsograms of weigth)
graph combine gr_ht gr_wt, cols(1) title(`"20`yr'") ///
        note("**Imputed values in each observation are averaged across 10
imputations")

* Guardar histogramas (Save histograms)
graph save ``gph_out'\ht_wt_hist`yr'.gph'', replace

*** Construir diagramas de caja de ÍMC por tamaño de localidad entre ***
*** observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para ***
*** observaciones donde ÍMC fue imputado (Construct box plots of BMI ***
*** by locality size among observations where BMI was observed and ***
*** among all observations, using medians across imputations for ob- ***
*** servations where BMI was imputed) ***
graph drop _all

* Casos no imputados (Non-imputed cases)

```

```
graph hbox bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0, ///
    over(tam_loc_`yr') title("Non-Imputed Cases") ///
    ytitle("BMI (kg/m2)") name(gr1)
```

* Todos los casos (All cases)

```
graph hbox bmi_med_`yr', over(tam_loc_`yr') title("All Cases*") ///
    ytitle("Median BMI (kg/m2)") name(gr2)
```

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)

```
graph combine gr1 gr2, rows(1) title(`"20`yr"'') ycommon ///
    note("*BMI's for imputed cases are medians among 10 imputations")
```

* Guardar diagramas de caja (Save box plots)

```
graph save `"gph_out'\bmi_tam_loc_box`yr'.gph"
```

```
*** Construir diagramas de caja de ÍMC por presencia o ausencia de diabetes ***
*** entre observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para observaciones ***
*** donde ÍMC fue imputado y excluyendo observaciones donde faltan datos ***
*** sobre diabetes (Construct box plots of BMI by presence or absence of ***
*** diabetes among observations where BMI was observed and among all obser- ***
*** vations, using medians across imutations for observations wher BMI was ***
*** imputed and excluding observations where data on diabetes are missing) ***
graph drop _all
```

* Casos no imputados (Non-imputed cases)

```
graph box bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0 & ///
    inlist(c6_`yr',1,2), over(c6_`yr', relabel(1 "Diabetic" 2 ///
    "Not Diabetic")) title("Non-Imputed Cases1") ///
    ytitle("BMI (kg/m2)") name(gr1)
```

* Todos los casos (All cases)

```
graph box bmi_med_`yr' if inlist(c6_`yr',1,2), over(c6_`yr', relabel(1 ///
    "Diabetic" 2 "Not Diabetic")) title("All Cases1,2") ///
    ytitle("Median BMI (kg/m2)") name(gr2)
```

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine

```

* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'") ycommon ///
    note("{superscript:1}Excludes observations where diabetic status is
missing" ///
    "{superscript:2}BMI's for imputed cases are medians among 10
imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_diab_box`yr'.gph'', replace

*** Construir diagramas de dispersión "agrupados" de logaritmos de ÍMC por ***
*** años de educación entre observaciones donde ÍMC fue observado y entre ***
*** todas las observaciones (Construct "binned" scatter plots of log-BMI ***
*** vs. years of education among observations where BMI was observed and ***
*** among all observations)
***
graph drop _all

* Casos no imputados (Non-imputed cases)
binscatter log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & ///
    wt_miss_`yr' == 0, xtitle("Years of education") ytitle("ln(BMI)") ///
    title("Non-Imputed Cases") name(gr1)

* Todos los casos (All cases)
binscatter log_bmi_mean_`yr' yrschool_mean, xtitle("Years of education") ///
    ytitle("ln(BMI)") title("All Cases*") name(gr2)

* Combinar diagramas de dispersión agrupados entre casos no imputados y entre
* todos los casos (Combine binned scatter plots among non-imputed cases and
* among all cases)
graph combine gr1 gr2, xcommon ycommon rows(1) title(`"20`yr'") ///
    note("*Imputed BMI's and imputed years of education are averaged
across 10 imputations")

* Guardar diagramas de dispersión (Save scatter plots)
graph save ``gph_out'\log_bmi_edu_bin`yr'.gph'', replace

*** PASO 7: GUARDAR BASE DE DATOS FINAL ***
*** STEP 7: SAVE FINAL DATASET ***

```

```

*** Borrar escalares (Delete scalars) ***
scalar drop _all

*** Borrar variables no necesarias (Delete unnecessary variables) ***
keep cunicalh np subhog_`yr' tam_loc_`yr' imp_ht_`yr' imp_ht_sd_`yr' ///
    imp_wt_`yr' imp_wt_sd_`yr' imp_bmi_`yr' imp_bmi_sd_`yr' ///
    ht_miss_`yr' wt_miss_`yr'

*** Cargar otras variables importantes, por ejemplo tipo de entrevista y ***
*** peso de muestreo (Load other important variables, e.g. type of ***
*** interview and sampling weight)
***
merge 1:m cunicalh np using "`maestro'", nogenerate assert(match using) ///
    keep(match) keepusing(UNHHID CUNICALH NP UNHHIDNP tipent_`yr' ///
    tam_loc_`yr' eam_`yr' factori_`yr' age_`yr' sex_`yr')

*** Guardar datos finales (Save final data) ***
save "`output'\MHAS_20`yr'_Weight_and_Height_IMPUTED.dta'", replace

2018

*** ENASEM 2018 — IMPUTACIÓN DE ALTURA Y PESO AUTO-REPORTADOS Y DE ÍNDICE DE ***
*** MASA CORPORAL (IMC)
***
*** MHAS 2018 — IMPUTATION OF SELF-REPORTED HEIGHT AND WEIGHT AND OF BODY ***
*** MASS INDEX (BMI)
***

*** Versión octubre 2020 (October 2020 Version) ***

version 15
local yr "18"

*** PASO 1: DEFINIR UBICACIONES DE CARPETAS ***
*** STEP 1: DEFINE LOCATIONS OF FILES ***

*** Convertir los siguientes macros en los directorios donde se guardan las ***
*** petas descritas abajo (Convert the following macros to the directories ***
*** where the files described below are stored)
***

local maestro "||maestro||" // El archivo maestro de seguimiento más reciente

```

```

// (Most recent master follow-up file)
local indiv "||individual||" // Bases de datos crudos al nivel individual
// (Individual-level databases)
local indiv01 "|individual_2001|"
// Datos al nivel individual de 2001 (Individual-level data from 2001)
local indiv03 "||individual_2003||"
// Datos al nivel individual de 2003 (Individual-level data from 2003)
local indiv12 "||individual_2012||"
// Datos al nivel individual de 2012 (Individual-level data from 2012)
local indiv15 "||individual_2015||"
// Datos al nivel individual de 2012 (Individual-level data from 2015)
local gph_out "||graph_output||"
// Figuras construidas en este programa (Figures constructed in this
// program)
local output "||output||" // Base de datos final con medias y desviaciones
// estándares entre imputaciones (Final dataset
with
// means and standard deviations across
imputations)

*** Cargar datos brutos (Load raw data) ***
local rmr "18" // Año de la ronda más reciente (Year of most recent wave)
local maestro "`maestro'\master_follow_up_file_20`rmr'.dta'"
use cunichah np subhog`yr' tipent`yr' tam_loc`yr' age`yr' sex`yr' if ///
inlist(tipent`yr',1,2,3,4) using "`maestro'", clear
// Se recomienda que se usa el archivo maestro de seguimiento
disponible
// más reciente. (It is recommended that the most recent availabel
mas-
// ter follow-up file is used.)

** Cargar otras variables para imputación (Load other variables for imputation)
local indiv`yr' "`indiv'\sect_a_c_d_f_e_pc_h_i_20`yr'.dta'"
// Directorio completo de base de datos al nivel individual de la
ronda
// actual (Complete directory of individual-level database from
current
// wave)
merge 1:1 cunichah np using "`indiv`yr'", assert(match) nogenerate ///
keepusing(c6`yr' c66`yr' c67_1`yr' c67_2`yr' yrschool)

```

```

        // Educación para nuevas personas, altura y peso auto-reportados y
        // diabetes (Education for new persons, self-reported height and
weight,
        // and diabetes)

*** PASO 2: PREPARAR DATOS CRUDOS PARA IMPUTACIÓN ***
*** STEP 2: PREPARE RAW DATA FOR IMPUTATION ***

*** Crear una sola variable de altura auto-reportada (Create single self- ***
*** reported height variable)
***
generate c67_`yr' = c67_1_`yr' * 100 + c67_2_`yr'
        // Altura auto-reportada sin zapatos en centímetros (Self-reported
        // height without shoes in centimeters)

*** Crear variable de educación para todos los sujetos (Create education ***
*** variable for all subjects)
***

* Cargar datos sobre educación de olas previas (Load data on education from pre-
* vious waves)
tempfile a01 aa03
merge 1:1 cunich np using ``maestro'', assert(match using) keep(match) ///
        nogenerate keepusing(unhhid codent01 acthog codent03)

preserve
use ``indiv01'\sect_a.dta'', clear
rename (ps3 a3_3) (codent01 a3_3_01)
save `a01'
use ``indiv03'\sect_aa.dta''
rename(ent2 aa4a_3) (codent03 aa4a_3_03)
destring acthog, replace
save `aa03'
restore

merge m:1 unhhid codent01 using `a01', keep(match master) keepusing(a3_3_01) ///
        generate(merge01)
merge m:1 cunich acthog codent03 using `aa03', keep(match master) ///

```

```

        keepusing(aa4a_3_03) generate(merge03)
merge 1:1 cunicah np using ``indiv12'\sect_a_c_d_e_pc_f_h_i_em_2012.dta'', ///
        generate(merge12) keep(match master) keepusing(yrschool_12 c67_1_12
///
        c67_2_12)
merge 1:1 cunicah np using ``indiv15'\sect_a_c_d_e_pc_f_h_i_2015.dta'', ///
        generate(merge15) keep(match master) keepusing(yrschool_15 c67_1_15
///
        c67_2_15)

* Construir variable yrschool (Construct variable yrschool)
replace yrschool = . if inlist(yrschool,.r,.d,.m)
replace yrschool = yrschool_15 if merge15 == 3 & missing(yrschool) & ///
        !(missing(yrschool_15) | yrschool_15 == 99)
replace yrschool = yrschool_12 if merge12 == 3 & missing(yrschool) & ///
        !(missing(yrschool_12) | inlist(yrschool_12,88,99))
replace yrschool = aa4a_3_03 if merge03 == 3 & missing(yrschool) & ///
        !(missing(aa4a_3_03) | aa4a_3_03 == 99)
replace yrschool = a3_3_01 if merge01 == 3 & missing(yrschool) & !(a3_3_01 ///
        == 99 | missing(a3_3_01))

*** Sustituir valores que indican datos faltantes y valores faltantes ***
*** extendidos (i.e., .a, .b,..., .z) por . ***
*** Replace values that indicate missing data and "extended" missing ***
*** values (i.e., .a, .b,..., .z) with . ***
replace c66_`yr' = . if inlist(c66_`yr',888,999)
replace c67_`yr' = . if inlist(c67_1_`yr',888,999) | inlist(c67_2_`yr',888,999)
replace age_`yr' = . if inlist(age_`yr',888,999)

*** Sustituir alturas reportadas en 2018 que no emparejan con la altura ***
*** reportada en cualquier ola previa por . (Replace heights reported ***
*** in 2012 that do not match the height reported in any previous wave ***
*** with .)
***
drop merge01 merge03
tempfile c01 c03
preserve
use ``indiv01'\sect_c.dta'', clear
rename (ps3 c72) (codent01 c72_01)

```

```

save `c01'
use ``indiv03'\sect_c.dta''
rename (ent2 c67) (codent03 c67_03)
destring acthog, replace
save `c03'
restore

merge m:1 unhhid codent01 using `c01', keep(match master) keepusing(c72_01) ///
    generate(merge01)
merge m:1 cunicah acthog codent03 using `c03', keep(match master) ///
    keepusing(c67_03) generate(merge03)
generate c67_12 = c67_1_12 * 100 + c67_2_12
generate c67_15 = c67_1_15 * 100 + c67_2_15
replace c67_12 = . if inlist(c67_1_12,8,9)
replace c67_`yr' = . if (merge01 == 3 & !(missing(c72_01) | ///
    missing(c67_`yr') | inrange(c72_01,888,999)) & abs((c67_`yr' - ///
    c72_01) / c72_01) > 0.1) | (merge03 == 3 & !(missing(c67_03) | ///
    missing(c67_`yr') | inlist(c67_03,888,999)) & abs((c67_`yr' - ///
    c67_03) / c67_03) > 0.1) | (merge12 == 3 & !(missing(c67_12) | ///
    missing(c67_`yr')) & abs((c67_`yr' - c67_12) / c67_12) > 0.1) | ///
    (merge15 == 3 & !(missing(c67_15) | missing(c67_`yr') | ///
    inlist(c67_1_15,9,888,999)) & abs((c67_`yr' - c67_15) / c67_15) > 0.1)

*** Crear indicadores de alturas y pesos faltantes (Create indicator of mis- ***
*** sing heights and weights)
***
generate byte ht_miss_`yr' = missing(c67_`yr')
generate byte wt_miss_`yr' = missing(c66_`yr')

*** PASO 3: IMPUTAR DATOS FALTANTES ***
*** STEP 3: IMPUTE MISSING DATA ***

*** Definir parámetros de imputación (cambiar los siguientes macros para ***
*** modificar las imputaciones)
***
*** Define parameters of imputation (change the following macros to modify ***
*** imputations)
***

```

```

local n_imp = 10 // Número de imputaciones (Number of imputations)
local n_iter = 300 // Número de iteraciones en imputación multivariada usando
                    // ecuaciones ligadas
                    // Number of iterations in multivariate imputation using
                    // chained equations (MICE)
local k = 5 // Número de "vecinos más cercanos" para emparejar medias predictivas
            // Number of "nearest neighbors" for predictive mean matching
(PMM)
local rng_seed = 101 // "Semilla" para generar números aleatorios ("Seed" to
                    // generate random numbers)

*** Declarar base de datos como "datos de imputación" y variables como ***
*** "imputadas" o "regulares" (ningún valor se imputará) ***
*** Declare database as "imputation data" and variables as "imputed or ***
*** "regular" (no values will be imputed)
***

mi set wide
mi register imputed c66_`yr' c67_`yr' age_`yr' yrschool
mi register regular tam_loc_`yr' sex_`yr' subhog_`yr' cunich np ///
                    ht_miss_`yr' wt_miss_`yr' c6_`yr'

*** Imputar valores faltantes (Impute missing values) ***
mi impute chained (pmm, knn(`k')) c66_`yr' c67_`yr' age_`yr' yrschool = ///
                    i.sex_`yr' i.tam_loc_`yr', burnin(`n_iter') rseed(`rng_seed') ///
                    add(`n_imp')

*** Crear variables pasivas (definidas como funciones de variables ***
*** imputadas)
***
*** Create passive variables (defined as functions of imputed ***
*** variables) ***

mi passive: generate bmi_`yr' = c66_`yr' / ((c67_`yr' / 100)^2)
mi passive: generate log_bmi_`yr' = log(bmi_`yr')

*** PASO 4: ESTIMAR CONEXIONES MARGINALES ENTRE EL LOGARITMO DE ÍMC Y ***
*** EDUCACIÓN, TAMAÑO DE LOCALIDAD Y DIABETES USANDO DATOS IMPUTADOS ***
*** STEP 4: ESTIMATE MARGINAL ASSOCIATIONS BETWEEN LOG-BMI AND EDUCATION, ***

```

```

*** LOCALITY SIZE, AND DIABETES USING IMPUTED DATA
***

mi estimate: regress log_bmi_`yr' yrschool // Educación
mi estimate: regress log_bmi_`yr' ib4.tam_loc_`yr'
              // Tamaño de localidad (Locality size)
mi estimate: regress log_bmi_`yr' ib2.c6_`yr' if inlist(c6_`yr',1,2) // Diabetes

*** PASO 5: PLEGAR BASE DE DATOS EN ESTADÍSTICAS SUMARIAS NECESARIAS PARA ***
*** BASE DE DATOS FINAL Y PARA TABLAS Y FIGURAS
***

*** STEP 5: COLLAPSE DATASET INTO SUMMARY STATISTICS NECESSARY FOR FINAL ***
*** DATASET AND FOR TABLES AND FIGURES
***

mi convert mlong, clear
collapse (mean) imp_ht_`yr'=c67_`yr' imp_wt_`yr'=c66_`yr' ///
          imp_bmi_`yr'=bmi_`yr' log_bmi_mean_`yr'=log_bmi_`yr' ///
          yrschool_mean=yrschool (sd) imp_ht_sd_`yr'=c67_`yr' ///
          imp_wt_sd_`yr'=c66_`yr' imp_bmi_sd_`yr'=bmi_`yr' ///
          yrschool_sd=yrschool (median) bmi_med_`yr'=bmi_`yr' (firstnm) ///
          subhog_`yr' _mi_m _mi_id _mi_miss ht_miss_`yr' wt_miss_`yr' c6_`yr'
///
          tam_loc_`yr', by(cunicah np)
mi extract 0, clear // Stata ya no reconoce estos datos como datos de
                    // imputación. (Stata no longer recognizes these data
as im-
                    // putation data.)

*** Sustituir desviaciones estándares por 0 en observaciones completas (Re- ***
*** place standard deviations with 0 in complete observations)
***
replace imp_ht_sd_`yr' = 0 if ht_miss_`yr' == 0
replace imp_wt_sd_`yr' = 0 if wt_miss_`yr' == 0
replace imp_bmi_sd_`yr' = 0 if ht_miss_`yr' == 0 & wt_miss_`yr' == 0

*** Reetiquetar variables que se les enviarán a usuarios (Relabel variables ***
*** that will be sent to users)
***
label variable imp_ht_`yr' "Imputed/observed height without shoes (cm) 20`yr'"
label variable imp_wt_`yr' "Imputed/observed weight (kg) 20`yr'"
label variable imp_bmi_`yr' "Imputed/observed BMI (kg/m^2) 20`yr'"

```

```

label variable imp_ht_sd_`yr' "Between-imputation standard deviation of height
20`yr'"

label variable imp_wt_sd_`yr' "Between-imputation standard deviation of weight
20`yr'"

label variable imp_bmi_sd_`yr' "Between-imputation standard deviation of BMI
20`yr'"

label variable subhog_`yr' "20`yr' sub-household identifier"

label variable ht_miss_`yr' "Height was imputed 20`yr'"

label variable wt_miss_`yr' "Weight was imputed 20`yr'"

*** Volver a aplicar etiquetas de valores (Reapply value labels) ***
label values subhog_`yr' clavesubhog

*** Agregarle notas sobre valores imputados a base de datos (Add note about ***
*** imputed values to dataset)
***
notes: Values of imp_ht_`yr' and imp_wt_`yr' where height and weight were imputed
(ht_miss_`yr' = 1 and wt_miss_`yr' = 1, respectively) were averaged across 10
imputations.

notes: Values of imp_bmi_`yr' where BMI was imputed (ht_miss_`yr' = 1 and/or
wt_miss_`yr' = 1) were averaged across 10 imputations.

*** Agregarle nota sobre desviaciones estándares entre imputaciones a base ***
*** de datos (Add note about between-imputation standard deviations to ***
*** dataset)
***
notes: imp_ht_sd_`yr', imp_wt_sd_`yr', and imp_bmi_sd_`yr' are 0 where height,
weight, and BMI, respectively, were observed (i.e., ht_miss_`yr' = 0,
wt_miss_`yr' = 0, and both ht_miss_`yr' and wt_miss_`yr' are 0, respectively)
because observed values are constant across imputations.

*** PASO 6: COMPARAR RESULTADOS DE ANALIZAR DATOS IMPUTADOS CON LOS DE ***
*** ANALIZAR DATOS BRUTOS
***

*** STEP 6: COMPARE RESULTS OF ANALYZING IMPUTED DATA WITH THOSE FROM ANALY- ***
*** ZING RAW DATA
***

*** Estimar conexiones marginales entre el logaritmo de ÍMC y educación, ***
*** tamaño de localidad y diabetes usando datos brutos (Estimate margi- ***
*** nal associations between log-BMI and education, locality size, and ***

```

```

*** diabetes using raw data
***

regress log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & wt_miss_`yr' ///
    == 0 & (yrschool_sd == 0 | missing(yrschool_sd))
regress log_bmi_mean_`yr' ib4.tam_loc_`yr' if ht_miss_`yr' == 0 & ///
    wt_miss_`yr' == 0
regress log_bmi_mean_`yr' ib2.c6_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == ///
    0 & inlist(c6_`yr',1,2)

*** Computar estadísticas sumarias y percentiles de alturas y peso auto- ***
*** reportados entre observaciones donde fueron observados, entre las ***
*** donde fueron imputados y entre todas las observaciones (Compute sum- ***
*** mary statistics and percentiles of self-reported heights and weights ***
*** among observations where they were observed, among those where they ***
*** were imputed, and among all observations)
***
foreach i in ht wt {
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 0, detail
    summarize imp_`i'_`yr' if `i'_miss_`yr' == 1, detail
    summarize imp_`i'_`yr', detail
}

*** Construir histogramas de alturas y pesos auto-reportados entre ***
*** observaciones donde fueron observados, entre las donde fueron ***
*** imputados y entre todas las observaciones (Construct histograms ***
*** of self-reported heights and weights among observations where ***
*** they were observed, among those where they were imputed, and ***
*** among all observations)
***
graph drop _all

* Altura (Height)
histogram imp_ht_`yr' if ht_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr1)
histogram imp_ht_`yr' if ht_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases") name(gr2) ytitle("")
histogram imp_ht_`yr', xtitle("") title("All Cases") name(gr3) ///
    ytitle("")
graph combine gr1 gr2 gr3, xcommon ycommon title("Height (cm)", position(6)) ///

```

```

        rows(1) name(gr_ht) // Combina histogramas de altura en una figura
(Com-
                                // bine histograms of height into one
figure)

* Peso (Weight)
histogram imp_wt_`yr' if wt_miss_`yr' == 0, xtitle("") ///
        title("Non-Imputed Cases") name(gr4)
histogram imp_wt_`yr' if wt_miss_`yr' == 1, xtitle("") ///
        title("Imputed Cases*") name(gr5) ytitle("")
histogram imp_wt_`yr', xtitle("") title("All Cases*") name(gr6) ytitle("")
graph combine gr4 gr5 gr6, xcommon ycommon title("Weight (kg)", position(6)) ///
        rows(1) name(gr_wt) // Combina histogramas de peso en una figura (Com-
                                // bine histograms of weight into one
figure)

* Combinar histogramas de altura con histogramas de peso (Combine histograms of
* height with hitsograms of weigth)
graph combine gr_ht gr_wt, cols(1) title(`"20`yr'") ///
        note("**Imputed values in each observation are averaged across 10
imputations")

* Guardar histogramas (Save histograms)
graph save ``gph_out'\ht_wt_hist`yr'.gph'', replace

*** Construir diagramas de caja de ÍMC por tamaño de localidad entre ***
*** observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para ***
*** observaciones donde ÍMC fue imputado (Construct box plots of BMI ***
*** by locality size among observations where BMI was observed and ***
*** among all observations, using medians across imputations for ob- ***
*** servations where BMI was imputed) ***
graph drop _all

* Casos no imputados (Non-imputed cases)
graph hbox bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0, ///
        over(tam_loc_`yr') title("Non-Imputed Cases") ///
        ytitle("BMI (kg/m2)") name(gr1)

```

```

* Todos los casos (All cases)
graph hbox bmi_med_`yr', over(tam_loc_`yr') title("All Cases*") ///
    ytitle("Median BMI (kg/m2)") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title("`20`yr'") ycommon ///
    note("*BMI's for imputed cases are medians among 10 imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_tam_loc_box`yr'.gph"

*** Construir diagramas de caja de ÍMC por presencia o ausencia de diabetes ***
*** entre observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para observaciones ***
*** donde ÍMC fue imputado y excluyendo observaciones donde faltan datos ***
*** sobre diabetes (Construct box plots of BMI by presence or absence of ***
*** diabetes among observations where BMI was observed and among all obser- ***
*** vations, using medians across imutations for observations wher BMI was ***
*** imputed and excluding observations where data on diabetes are missing) ***
graph drop _all

* Casos no imputados (Non-imputed cases)
graph box bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0 & ///
    inlist(c6_`yr',1,2), over(c6_`yr', relabel(1 "Diabetic" 2 ///
    "Not Diabetic")) title("Non-Imputed Cases1") ///
    ytitle("BMI (kg/m2)") name(gr1)

* Todos los casos (All cases)
graph box bmi_med_`yr' if inlist(c6_`yr',1,2), over(c6_`yr', relabel(1 ///
    "Diabetic" 2 "Not Diabetic")) title("All Cases1,2") ///
    ytitle("Median BMI (kg/m2)") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title("`20`yr'") ycommon ///
    note("1Excludes observations where diabetic status is
missing" ///

```

```
"{superscript:2}BMI's for imputed cases are medians among 10
imputations")
```

```
* Guardar diagramas de caja (Save box plots)
```

```
graph save ``gph_out'\bmi_diab_box`yr'.gph"', replace
```

```
*** Construir diagramas de dispersión "agrupados" de logaritmos de ÍMC por ***
*** años de educación entre observaciones donde ÍMC fue observado y entre ***
*** todas las observaciones (Construct "binned" scatter plots of log-BMI ***
*** vs. years of education among observations where BMI was observed and ***
*** among all observations)
***
```

```
graph drop _all
```

```
* Casos no imputados (Non-imputed cases)
```

```
binscatter log_bmi_mean`yr' yrschool_mean if ht_miss`yr' == 0 & ///
           wt_miss`yr' == 0, xtitle("Years of education") ytitle("ln(BMI)") ///
           title("Non-Imputed Cases") name(gr1)
```

```
* Todos los casos (All cases)
```

```
binscatter log_bmi_mean`yr' yrschool_mean, xtitle("Years of education") ///
           ytitle("ln(BMI)") title("All Cases*") name(gr2)
```

```
* Combinar diagramas de dispersión agrupados entre casos no imputados y entre
* todos los casos (Combine binned scatter plots among non-imputed cases and
* among all cases)
```

```
graph combine gr1 gr2, xcommon ycommon rows(1) title(`"20`yr"'') ///
           note("*Imputed BMI's and imputed years of education are averaged
across 10 imputations")
```

```
* Guardar diagramas de dispersión (Save scatter plots)
```

```
graph save ``gph_out'\log_bmi_edu_bin`yr'.gph"', replace
```

```
*** PASO 7: GUARDAR BASE DE DATOS FINAL ***
```

```
*** STEP 7: SAVE FINAL DATASET ***
```

```
*** Borrar variables no necesarias (Delete unnecessary variables) ***
```

```
keep cunich np subhog`yr' tam_loc`yr' imp_ht`yr' imp_ht_sd`yr' ///
           imp_wt`yr' imp_wt_sd`yr' imp_bmi`yr' imp_bmi_sd`yr' ///
```

```
ht_miss_`yr' wt_miss_`yr'
```

```
*** Cargar otras variables importantes, por ejemplo tipo de entrevista y ***  
*** peso de muestreo (Load other important variables, e.g. type of ***  
*** interview and sampling weight)  
***
```

```
merge 1:m cunicah np using ``maestro'', nogenerate assert(match using) ///  
    keep(match) keepusing(UNHHID CUNICAH NP UNHHIDNP tipent_`yr' ///  
    tam_loc_`yr' eam_`yr' factori_`yr' age_`yr' sex_`yr')
```

```
*** Guardar datos finales (Save final data) ***
```

```
save ``output'\MHAS_20`yr'_Weight_and_Height_IMPUTED.dta'', replace
```

2018

```
*** ENASEM 2018 — IMPUTACIÓN DE ALTURA Y PESO AUTO-REPORTADOS Y DE ÍNDICE DE ***
*** MASA CORPORAL (IMC)
***
*** MHAS 2018 — IMPUTATION OF SELF-REPORTED HEIGHT AND WEIGHT AND OF BODY ***
*** MASS INDEX (BMI)
***
*** Versión octubre 2020 (October 2020 Version) ***

version 15
local yr "18"

*** PASO 1: DEFINIR UBICACIONES DE CARPETAS ***
*** STEP 1: DEFINE LOCATIONS OF FILES ***

*** Convertir los siguientes macros en los directorios donde se guardan las ***
*** petas descritas abajo (Convert the following macros to the directories ***
*** where the files described below are stored)
***

local maestro "||maestro||" // El archivo maestro de seguimiento más reciente
                        // (Most recent master follow-up file)
local indiv  "||individual||" // Bases de datos crudos al nivel individual
                        // (Individual-level databases)
local indiv01 "||individual_2001||"
                        // Datos al nivel individual de 2001 (Individual-level data from 2001)
local indiv03 "||individual_2003||"
                        // Datos al nivel individual de 2003 (Individual-level data from 2003)
local indiv12 "||individual_2012||"
                        // Datos al nivel individual de 2012 (Individual-level data from 2012)
local indiv15 "||individual_2015||"
                        // Datos al nivel individual de 2012 (Individual-level data from 2015)
local gph_out "||graph_output||"
                        // Figuras construidas en este programa (Figures constructed in this
                        // program)
local output "||output||" // Base de datos final con medias y desviaciones
                        // estándares entre imputaciones (Final dataset
with
                        // means and standard deviations across
imputations)
```

```

*** Cargar datos brutos (Load raw data) ***
local rmr "18" // Año de la ronda más reciente (Year of most recent wave)
local maestro "`maestro'\master_follow_up_file_20`rmr'.dta'"
use cunichah np subhog_`yr' tipent_`yr' tam_loc_`yr' age_`yr' sex_`yr' if ///
    inlist(tipent_`yr',1,2,3,4) using "`maestro'", clear
    // Se recomienda que se usa el archivo maestro de seguimiento
disponible
    // más reciente. (It is recommended that the most recent available
mas-
    // ter follow-up file is used.)

** Cargar otras variables para imputación (Load other variables for imputation)
local indiv_`yr' "`indiv'\sect_a_c_d_f_e_pc_h_i_20`yr'.dta'"
    // Directorio completo de base de datos al nivel individual de la
ronda
    // actual (Complete directory of individual-level database from
current
    // wave)
merge 1:1 cunichah np using "`indiv_`yr'", assert(match) nogenerate ///
    keepusing(c6_`yr' c66_`yr' c67_1_`yr' c67_2_`yr' yrschool)
    // Educación para nuevas personas, altura y peso auto-reportados y
    // diabetes (Education for new persons, self-reported height and
weight,
    // and diabetes)

*** PASO 2: PREPARAR DATOS CRUDOS PARA IMPUTACIÓN ***
*** STEP 2: PREPARE RAW DATA FOR IMPUTATION ***

*** Crear una sola variable de altura auto-reportada (Create single self- ***
*** reported height variable)
***
generate c67_`yr' = c67_1_`yr' * 100 + c67_2_`yr'
    // Altura auto-reportada sin zapatos en centímetros (Self-reported
    // height without shoes in centimeters)

*** Crear variable de educación para todos los sujetos (Create education ***
*** variable for all subjects)
***

```

```

* Cargar datos sobre educación de olas previas (Load data on education from pre-
* vious waves)
tempfile a01 aa03
merge 1:1 cunicah np using ``maestro'', assert(match using) keep(match) ///
        nogenerate keepusing(unhhid codent01 acthog codent03)

preserve
use ``indiv01'\sect_a.dta'', clear
rename (ps3 a3_3) (codent01 a3_3_01)
save `a01'
use ``indiv03'\sect_aa.dta''
rename(ent2 aa4a_3) (codent03 aa4a_3_03)
destring acthog, replace
save `aa03'
restore

merge m:1 unhhid codent01 using `a01', keep(match master) keepusing(a3_3_01) ///
        generate(merge01)
merge m:1 cunicah acthog codent03 using `aa03', keep(match master) ///
        keepusing(aa4a_3_03) generate(merge03)
merge 1:1 cunicah np using ``indiv12'\sect_a_c_d_e_pc_f_h_i_em_2012.dta'', ///
        generate(merge12) keep(match master) keepusing(yrschool_12 c67_1_12
///
        c67_2_12)
merge 1:1 cunicah np using ``indiv15'\sect_a_c_d_e_pc_f_h_i_2015.dta'', ///
        generate(merge15) keep(match master) keepusing(yrschool_15 c67_1_15
///
        c67_2_15)

* Construir variable yrschool (Construct variable yrschool)
replace yrschool = . if inlist(yrschool,.r,.d,.m)
replace yrschool = yrschool_15 if merge15 == 3 & missing(yrschool) & ///
        !(missing(yrschool_15) | yrschool_15 == 99)
replace yrschool = yrschool_12 if merge12 == 3 & missing(yrschool) & ///
        !(missing(yrschool_12) | inlist(yrschool_12,88,99))
replace yrschool = aa4a_3_03 if merge03 == 3 & missing(yrschool) & ///
        !(missing(aa4a_3_03) | aa4a_3_03 == 99)
replace yrschool = a3_3_01 if merge01 == 3 & missing(yrschool) & !(a3_3_01 ///
        == 99 | missing(a3_3_01))

```

```

*** Sustituir valores que indican datos faltantes y valores faltantes ***
*** extendidos (i.e., .a, .b,..., .z) por . ***
*** Replace values that indicate missing data and "extended" missing ***
*** values (i.e., .a, .b,..., .z) with . ***
replace c66_`yr' = . if inlist(c66_`yr',888,999)
replace c67_`yr' = . if inlist(c67_1_`yr',888,999) | inlist(c67_2_`yr',888,999)
replace age_`yr' = . if inlist(age_`yr',888,999)

*** Sustituir alturas reportadas en 2018 que no emparejan con la altura ***
*** reportada en cualquier ola previa por . (Replace heights reported ***
*** in 2012 that do not match the height reported in any previous wave ***
*** with .) ***
drop merge01 merge03
tempfile c01 c03
preserve
use ```indiv01'\sect_c.dta'', clear
rename (ps3 c72) (codent01 c72_01)
save `c01'
use ```indiv03'\sect_c.dta''
rename (ent2 c67) (codent03 c67_03)
destring acthog, replace
save `c03'
restore

merge m:1 unhhid codent01 using `c01', keep(match master) keepusing(c72_01) ///
    generate(merge01)
merge m:1 cunicah acthog codent03 using `c03', keep(match master) ///
    keepusing(c67_03) generate(merge03)
generate c67_12 = c67_1_12 * 100 + c67_2_12
generate c67_15 = c67_1_15 * 100 + c67_2_15
replace c67_12 = . if inlist(c67_1_12,8,9)
replace c67_`yr' = . if (merge01 == 3 & !(missing(c72_01) | ///
    missing(c67_`yr') | inrange(c72_01,888,999)) & abs((c67_`yr' - ///
    c72_01) / c72_01) > 0.1) | (merge03 == 3 & !(missing(c67_03) | ///
    missing(c67_`yr') | inlist(c67_03,888,999)) & abs((c67_`yr' - ///
    c67_03) / c67_03) > 0.1) | (merge12 == 3 & !(missing(c67_12) | ///

```

```

missing(c67_`yr')) & abs((c67_`yr' - c67_12) / c67_12) > 0.1) | ///
(merge15 == 3 & !(missing(c67_15) | missing(c67_`yr')) | ///
inlist(c67_1_15,9,888,999)) & abs((c67_`yr' - c67_15) / c67_15) > 0.1)

*** Crear indicadores de alturas y pesos faltantes (Create indicator of mis- ***
*** sing heights and weights)
***
generate byte ht_miss_`yr' = missing(c67_`yr')
generate byte wt_miss_`yr' = missing(c66_`yr')

*** PASO 3: IMPUTAR DATOS FALTANTES ***
*** STEP 3: IMPUTE MISSING DATA ***

*** Definir parámetros de imputación (cambiar los siguientes macros para ***
*** modificar las imputaciones)
***
*** Define parameters of imputation (change the following macros to modify ***
*** imputations)
***

local n_imp = 10 // Número de imputaciones (Number of imputations)
local n_iter = 300 // Número de iteraciones en imputación multivariada usanda
// ecuaciones ligadas
// Number of iterations in multivariate imputation using
// chained equations (MICE)

local k = 5 // Número de "vecinos más cercanos" para emparejar medias predictivas
// Number of "nearest neighbors" for predictive mean matching
(PMM)

local rng_seed = 101 // "Semilla" para generar números aleatorios ("Seed" to
// generate random numbers)

*** Declarar base de datos como "datos de imputación" y variables como ***
*** "imputadas" o "regulares" (ningún valor se imputará) ***
*** Declare database as "imputation data" and variables as "imputed or ***
*** "regular" (no values will be imputed)
***

mi set wide

mi register imputed c66_`yr' c67_`yr' age_`yr' yrschool
mi register regular tam_loc_`yr' sex_`yr' subhog_`yr' cunicah np ///
ht_miss_`yr' wt_miss_`yr' c6_`yr'

```

```

*** Imputar valores faltantes (Impute missing values) ***
mi impute chained (pmm, knn(`k')) c66_`yr' c67_`yr' age_`yr' yrschool = ///
    i.sex_`yr' i.tam_loc_`yr', burnin(`n_iter') rseed(`rng_seed') ///
    add(`n_imp')

*** Crear variables pasivas (definidas como funciones de variables ***
*** imputadas)
***

*** Create passive variables (defined as functions of imputed ***
*** variables) ***

mi passive: generate bmi_`yr' = c66_`yr' / ((c67_`yr' / 100)^2)
mi passive: generate log_bmi_`yr' = log(bmi_`yr')

*** PASO 4: ESTIMAR CONEXIONES MARGINALES ENTRE EL LOGARITMO DE ÍMC Y ***
*** EDUCACIÓN, TAMAÑO DE LOCALIDAD Y DIABETES USANDO DATOS IMPUTADOS ***
*** STEP 4: ESTIMATE MARGINAL ASSOCIATIONS BETWEEN LOG-BMI AND EDUCATION, ***
*** LOCALITY SIZE, AND DIABETES USING IMPUTED DATA
***

mi estimate: regress log_bmi_`yr' yrschool // Educación
mi estimate: regress log_bmi_`yr' ib4.tam_loc_`yr'
    // Tamaño de localidad (Locality size)
mi estimate: regress log_bmi_`yr' ib2.c6_`yr' if inlist(c6_`yr',1,2) // Diabetes

*** PASO 5: PLEGAR BASE DE DATOS EN ESTADÍSTICAS SUMARIAS NECESARIAS PARA ***
*** BASE DE DATOS FINAL Y PARA TABLAS Y FIGURAS
***

*** STEP 5: COLLAPSE DATASET INTO SUMMARY STATISTICS NECESSARY FOR FINAL ***
*** DATASET AND FOR TABLES AND FIGURES
***

mi convert mlong, clear

collapse (mean) imp_ht_`yr'=c67_`yr' imp_wt_`yr'=c66_`yr' ///
    imp_bmi_`yr'=bmi_`yr' log_bmi_mean_`yr'=log_bmi_`yr' ///
    yrschool_mean=yrschool (sd) imp_ht_sd_`yr'=c67_`yr' ///
    imp_wt_sd_`yr'=c66_`yr' imp_bmi_sd_`yr'=bmi_`yr' ///
    yrschool_sd=yrschool (median) bmi_med_`yr'=bmi_`yr' (firstnm) ///
    subhog_`yr' _mi_m _mi_id _mi_miss ht_miss_`yr' wt_miss_`yr' c6_`yr'
///

tam_loc_`yr', by(cunicah np)

```

```

mi extract 0, clear // Stata ya no reconoce estos datos como datos de
                        // imputación. (Stata no longer recognizes these data
as im-
                        // putation data.)

*** Sustituir desviaciones estándares por 0 en observaciones completas (Re- ***
*** place standard deviations with 0 in complete observations)
***
replace imp_ht_sd`yr' = 0 if ht_miss`yr' == 0
replace imp_wt_sd`yr' = 0 if wt_miss`yr' == 0
replace imp_bmi_sd`yr' = 0 if ht_miss`yr' == 0 & wt_miss`yr' == 0

*** Reetiquetar variables que se les enviarán a usuarios (Relabel variables ***
*** that will be sent to users)
***
label variable imp_ht`yr' "Imputed/observed height without shoes (cm) 20`yr'"
label variable imp_wt`yr' "Imputed/observed weight (kg) 20`yr'"
label variable imp_bmi`yr' "Imputed/observed BMI (kg/m^2) 20`yr'"
label variable imp_ht_sd`yr' "Between-imputation standard deviation of height
20`yr'"
label variable imp_wt_sd`yr' "Between-imputation standard deviation of weight
20`yr'"
label variable imp_bmi_sd`yr' "Between-imputation standard deviation of BMI
20`yr'"
label variable subhog`yr' "20`yr' sub-household identifier"
label variable ht_miss`yr' "Height was imputed 20`yr'"
label variable wt_miss`yr' "Weight was imputed 20`yr'"

*** Volver a aplicar etiquetas de valores (Reapply value labels) ***
label values subhog`yr' clavesubhog

*** Agregarle notas sobre valores imputados a base de datos (Add note about ***
*** imputed values to dataset)
***
notes: Values of imp_ht`yr' and imp_wt`yr' where height and weight were imputed
(ht_miss`yr' = 1 and wt_miss`yr' = 1, respectively) were averaged across 10
imputations.

notes: Values of imp_bmi`yr' where BMI was imputed (ht_miss`yr' = 1 and/or
wt_miss`yr' = 1) were averaged across 10 imputations.

*** Agregarle nota sobre desviaciones estándares entre imputaciones a base ***

```

```

*** de datos (Add note about between-imputation standard deviations to ***
*** dataset)
***
notes: imp_ht_sd`yr', imp_wt_sd`yr', and imp_bmi_sd`yr' are 0 where height,
weight, and BMI, respectively, were observed (i.e., ht_miss`yr' = 0,
wt_miss`yr' = 0, and both ht_miss`yr' and wt_miss`yr' are 0, respectively)
because observed values are constant across imputations.

*** PASO 6: COMPARAR RESULTADOS DE ANALIZAR DATOS IMPUTADOS CON LOS DE ***
*** ANALIZAR DATOS BRUTOS
***
*** STEP 6: COMPARE RESULTS OF ANALYZING IMPUTED DATA WITH THOSE FROM ANALY- ***
*** ZING RAW DATA
***

*** Estimar conexiones marginales entre el logaritmo de ÍMC y educación, ***
*** tamaño de localidad y diabetes usando datos brutos (Estimate margi- ***
*** nal associations between log-BMI and education, locality size, and ***
*** diabetes using raw data
***

regress log_bmi_mean`yr' yrschool_mean if ht_miss`yr' == 0 & wt_miss`yr' ///
      == 0 & (yrschool_sd == 0 | missing(yrschool_sd))
regress log_bmi_mean`yr' ib4.tam_loc`yr' if ht_miss`yr' == 0 & ///
      wt_miss`yr' == 0
regress log_bmi_mean`yr' ib2.c6`yr' if ht_miss`yr' == 0 & wt_miss`yr' == ///
      0 & inlist(c6`yr',1,2)

*** Computar estadísticas sumarias y percentiles de alturas y peso auto- ***
*** reportados entre observaciones donde fueron observados, entre las ***
*** donde fueron imputados y entre todas las observaciones (Compute sum- ***
*** mary statistics and percentiles of self-reported heights and weights ***
*** among observations where they were observed, among those where they ***
*** were imputed, and among all observations)
***

foreach i in ht wt {
    summarize imp_`i'_`yr' if `i'_miss`yr' == 0, detail
    summarize imp_`i'_`yr' if `i'_miss`yr' == 1, detail
    summarize imp_`i'_`yr', detail
}

```

```

*** Construir histogramas de alturas y pesos auto-reportados entre ***
*** observaciones donde fueron observados, entre las donde fueron ***
*** imputados y entre todas las observaciones (Construct histograms ***
*** of self-reported heights and weights among observations where ***
*** they were observed, among those where they were imputed, and ***
*** among all observations)
***

graph drop _all

* Altura (Height)
histogram imp_ht_`yr' if ht_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr1)
histogram imp_ht_`yr' if ht_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases") name(gr2) ytitle("")
histogram imp_ht_`yr', xtitle("") title("All Cases") name(gr3) ///
    ytitle("")
graph combine gr1 gr2 gr3, xcommon ycommon title("Height (cm)", position(6)) ///
    rows(1) name(gr_ht) // Combina histogramas de altura en una figura
(Com-
                                // bine histograms of height into one
figure)

* Peso (Weight)
histogram imp_wt_`yr' if wt_miss_`yr' == 0, xtitle("") ///
    title("Non-Imputed Cases") name(gr4)
histogram imp_wt_`yr' if wt_miss_`yr' == 1, xtitle("") ///
    title("Imputed Cases") name(gr5) ytitle("")
histogram imp_wt_`yr', xtitle("") title("All Cases") name(gr6) ytitle("")
graph combine gr4 gr5 gr6, xcommon ycommon title("Weight (kg)", position(6)) ///
    rows(1) name(gr_wt) // Combina histogramas de peso en una figura (Com-
                                // bine histograms of weight into one
figure)

* Combinar histogramas de altura con histogramas de peso (Combine histograms of
* height with hitsograms of weigth)
graph combine gr_ht gr_wt, cols(1) title("`20`yr'") ///
    note("**Imputed values in each observation are averaged across 10
imputations")

* Guardar histogramas (Save histograms)

```

```
graph save ``gph_out'\ht_wt_hist`yr'.gph'', replace
```

```
*** Construir diagramas de caja de ÍMC por tamaño de localidad entre ***
*** observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para ***
*** observaciones donde ÍMC fue imputado (Construct box plots of BMI ***
*** by locality size among observations where BMI was observed and ***
*** among all observations, using medians across imputations for ob- ***
*** servations where BMI was imputed) ***
graph drop _all
```

```
* Casos no imputados (Non-imputed cases)
```

```
graph hbox bmi_med`yr' if ht_miss`yr' == 0 & wt_miss`yr' == 0, ///
    over(tam_loc`yr') title("Non-Imputed Cases") ///
    ytitle("BMI (kg/m2)") name(gr1)
```

```
* Todos los casos (All cases)
```

```
graph hbox bmi_med`yr', over(tam_loc`yr') title("All Cases") ///
    ytitle("Median BMI (kg/m2)") name(gr2)
```

```
* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
```

```
graph combine gr1 gr2, rows(1) title(`"20`yr"') ycommon ///
    note("*BMI's for imputed cases are medians among 10 imputations")
```

```
* Guardar diagramas de caja (Save box plots)
```

```
graph save ``gph_out'\bmi_tam_loc_box`yr'.gph''
```

```
*** Construir diagramas de caja de ÍMC por presencia o ausencia de diabetes ***
*** entre observaciones donde ÍMC fue observado y entre todas las ***
*** observaciones, usando medianas entre imputaciones para observaciones ***
*** donde ÍMC fue imputado y excluyendo observaciones donde faltan datos ***
*** sobre diabetes (Construct box plots of BMI by presence or absence of ***
*** diabetes among observations where BMI was observed and among all obser- ***
*** vations, using medians across imutations for observations wher BMI was ***
*** imputed and excluding observations where data on diabetes are missing) ***
graph drop _all
```

```

* Casos no imputados (Non-imputed cases)
graph box bmi_med_`yr' if ht_miss_`yr' == 0 & wt_miss_`yr' == 0 & ///
    inlist(c6_`yr',1,2), over(c6_`yr', relabel(1 "Diabetic" 2 ///
    "Not Diabetic")) title("Non-Imputed Cases{superscript:1}") ///
    ytitle("BMI (kg/m{superscript:2})") name(gr1)

* Todos los casos (All cases)
graph box bmi_med_`yr' if inlist(c6_`yr',1,2), over(c6_`yr', relabel(1 ///
    "Diabetic" 2 "Not Diabetic")) title("All Cases{superscript:1,2}") ///
    ytitle("Median BMI (kg/m{superscript:2})") name(gr2)

* Combinar diagramas entre casos no imputados y entre todos los casos (Combine
* plots for non-imputed cases and for all cases)
graph combine gr1 gr2, rows(1) title(`"20`yr'") ycommon ///
    note("{superscript:1}Excludes observations where diabetic status is
missing" ///
    "{superscript:2}BMI's for imputed cases are medians among 10
imputations")

* Guardar diagramas de caja (Save box plots)
graph save ``gph_out'\bmi_diab_box`yr'.gph"', replace

*** Construir diagramas de dispersión "agrupados" de logaritmos de ÍMC por ***
*** años de educación entre observaciones donde ÍMC fue observado y entre ***
*** todas las observaciones (Construct "binned" scatter plots of log-BMI ***
*** vs. years of education among observations where BMI was observed and ***
*** among all observations)
***
graph drop _all

* Casos no imputados (Non-imputed cases)
binscatter log_bmi_mean_`yr' yrschool_mean if ht_miss_`yr' == 0 & ///
    wt_miss_`yr' == 0, xtitle("Years of education") ytitle("ln(BMI)") ///
    title("Non-Imputed Cases") name(gr1)

* Todos los casos (All cases)
binscatter log_bmi_mean_`yr' yrschool_mean, xtitle("Years of education") ///
    ytitle("ln(BMI)") title("All Cases*") name(gr2)

```

```

* Combinar diagramas de dispersión agrupados entre casos no imputados y entre
* todos los casos (Combine binned scatter plots among non-imputed cases and
* among all cases)
graph combine gr1 gr2, xcommon ycommon rows(1) title(`"20`yr'") ///
    note("Imputed BMI's and imputed years of education are averaged
across 10 imputations")

* Guardar diagramas de dispersión (Save scatter plots)
graph save `"gph_out'\log_bmi_edu_bin`yr'.gph"', replace

*** PASO 7: GUARDAR BASE DE DATOS FINAL ***
*** STEP 7: SAVE FINAL DATASET ***

*** Borrar variables no necesarias (Delete unnecessary variables) ***
keep cunicalh np subhog_`yr' imp_ht_`yr' imp_ht_sd_`yr'
imp_wt_`yr' ///
    imp_wt_sd_`yr' imp_bmi_`yr' imp_bmi_sd_`yr'
ht_miss_`yr' wt_miss_`yr'

*** Cargar otras variables importantes, por ejemplo tipo de entrevista y ***
*** peso de muestreo (Load other important variables, e.g. type of ***
*** interview and sampling weight)
***
merge 1:m cunicalh np using `"maestro"', nogenerate assert(match using) ///
    keep(match) keepusing(UNHHID CUNICALH NP UNHHIDNP tipent_`yr' ///
    tam_loc_`yr' eam_`yr' factori_`yr' age_`yr' sex_`yr')

*** Guardar datos finales (Save final data) ***
save `"output'\MHAS_20`yr'_Weight_and_Height_IMPUTED.dta"', replace

```

2021

```
***** BMI_Imputation2021.do *****
** Purpose: To impute missing self-reported heights and weights in MHAS 2021 **
** and compute BMI's from both observed and imputed heights and weights *****
***** Version: January 2024 *****
```

```
local yr "21"
```

```
***** STEP 1: ESTABLISH DIRECTORIES WHERE NECESSARY RAW DATA FILES ARE *****
***** STORED *****
```

```
local wv 21 // Last two digits of year of current wave
local raw01 "||Health data from MHAS 2001||"
local raw03 "||Health data from MHAS 2003||"
local raw12 "||Individual-level 2012 MHAS data ||"
local raw15 "||Individual-level 2015 MHAS data||"
local raw18 "||Individual-level 2018 MHAS data||"
local raw`wv' `||Individual-level 20`wv' MHAS data||"
```

```
***** STEP 2: CLEAN DATA *****
```

```
use ``raw`wv'\sect_a_c_d_e_pc_f_h_i_2021.dta"
```

```
*** Remove missing value codes in height and weight ***
```

```
generate height_`wv' = c67_1_`wv' + c67_2_`wv' / 100 if ///
    !(inlist(c67_1_`wv',888,999) | inlist(c67_2_`wv',888,999))
```

```
summarize height_`wv'
```

```
recode age_`wv' (999 = .), generate(age2_`wv')
```

```
recode yrschool (missing = .), generate(education)
```

```
recode c66_`wv' (888 999 = .), generate(weight_`wv')
```

```
save BMI_Imputation`wv'
```

```
*** Substitute system missing values for observations where height in 2021 ***
```

```
*** differs from height in any previous wave by more than 10%
```

```
***
```

```
* 2018
```

```
merge 1:1 cunicah np using ``raw18'/sect_a_c_d_f_e_pc_h_i_2018.dta"',
keep(master match) keepusing(c67_1_18 c67_2_18) nogenerate
```

```

generate height_18 = c67_1_18 + c67_2_18 / 100 if !(inlist(c67_1_18,888,999,.) |
inlist(c67_2_18,888,999,.))

* 2015

merge 1:1 cunicah np using ``raw15'sect_a_c_d_e_pc_f_h_i_2015.dta'', keep(master
match) keepusing(c67_1_15 c67_2_15) nogenerate

generate height_15 = c67_1_15 + c67_2_15 if !(inlist(c67_1_15,888,999,.i) |
c67_2_15 == .i)

replace height_15 = c67_1_15 + c67_2_15 / 100 if !(inlist(c67_1_15,888,999,.i) |
c67_2_15 == .i)

merge 1:1 cunicah np using
"C:\Users\User\Documents\MHAS\InputFiles\2012\sect_a_c_d_e_pc_f_h_i_em_2012.dta",
keep(master match) keepusing(c67_1_12 c67_2_12) nogenerate

generate height_12 = c67_1_12 + c67_2_12 / 100 if inlist(c67_1_12,1,2)

* Load identifiers needed to merge data from 2001 & 2003

save BMI_Imputation`wv', replace

merge 1:1 cunicah np using
"C:\Users\User\Downloads\master_follow_up_file_2021(1).dta", assert(using match)
keep(match) keepusing(acthog codent01 codent03) nogenerate

save BMI_Imputation`wv', replace

browse cunicah acthog codent03

generate height_03 = c67_03 / 100 if c67_03 < 888
generate height_01 = c72_01 / 100 if c72_01 < 888
save BMI_Imputation`wv', replace

* ! Identify where heights reported in 2021 differ from those reported in any
prior wave by more than 10%

save BMI_Imputation`wv', replace

* !!! Prepare data for imputation

mi set wide

mi register regular unhhid cunicah np subhog`wv' tipent`wv' factor?`wv' acthog
codent01 codent03

mi register regular sex`wv' tam_loc`wv'

mi register imputed age2`wv' education weight`wv' height`wv'

mi register passive age`wv' yrschool

mi unregister age`wv' yrschool

```

```

* ! Construct flags indicating where each imputed variable is missing
mi xeq: generate byte age_flag_`wv' = missing(age2_`wv'); label variable
age_flag_`wv' "Age was imputed 2021"
mi xeq: label values age_flag_`wv' yesno
mi describe

mi xeq: generate byte edu_flag_`wv' = missing(education); label variable
edu_flag_`wv' "Years of education were imputed 2021"; generate byte wt_flag_`wv'
= missing(weight_`wv'); label variable wt_flag_`wv' "Weight (kg) was imputed
2021"; generate byte ht_flag_`wv' =
> missing(height_`wv'); label variable ht_flag_`wv' "Height w/o shoes (m) was
imputed 2021"; label values edu_flag_`wv' wt_flag_`wv' ht_flag_`wv' yesno
mi register regular age_flag_`wv' edu_flag_`wv' wt_flag_`wv' ht_flag_`wv'

* !!! Impute missing data
* ! Check if imputed variables have monotone missing pattern
mi misstable nested age2_`wv' education weight_`wv' height_`wv'

* !! Check how many iterations needed for MICE to converge
mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) chainonly savetrace(impstats) burnin(10)

* 10 iterations
tsset iter
graph drop gr
graph drop _all
describe
graph drop _all

* 50 iterations
mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) chainonly savetrace(impstats, replace)
burnin(50)
graph drop _all
use "..\2021\impstats.dta", clear
tsset iter
do "C:\Users\User\AppData\Local\Temp\STD6054_000000.tmp"
* 75 iterations

```

```

mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) chainonly savetrace(impstats, replace)
burnin(75)

do "C:\Users\User\AppData\Local\Temp\STD6054_000000.tmp"

* 75 iterations

* 90 iterations

mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) chainonly savetrace(impstats, replace)
burnin(90)

do "C:\Users\User\AppData\Local\Temp\STD6054_000000.tmp"

mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) chainonly savetrace(impstats, replace)
burnin(100)

* 100 iterations

do "C:\Users\User\AppData\Local\Temp\STD6054_000000.tmp"

* 125 iterations

mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) chainonly savetrace(impstats, replace)
burnin(125)

do "C:\Users\User\AppData\Local\Temp\STD6054_000000.tmp"

graph combine gr1 gr2 gr3 gr4 gr5 gr6 gr7 gr8, rows(2) title("Trace plots of
summaries of imputed values")

exit, clear

log close

* !! Perform imputations

use BMI_Imputation_`wv'

mi impute chained (pmm, knn(5)) age2_`wv' education weight_`wv' height_`wv' =
i.sex_`wv' i.tam_loc_`wv', rseed(101) add(10) burnin(125) dots nomonotonechk
save BMI_Imputation_`wv', replace

log close

use BMI_Imputation_`wv'

* !!! Construct BMI as passively imputed variable

mi passive: generate bmi_`wv' = weight_`wv' / height_`wv'^2

* !!! Compute means and standard deviations of imputed variables, and medians
* and standard deviations of BMI across 10 imputations

drop imp_age*21 imp_edu* imp_wt_`wv'

mi update

do "C:\Users\User\AppData\Local\Temp\STD5b38_000000.tmp"

```

```

mi xeq: summarize imp_age_`wv'-imp_bmi_sd_`wv'
do "C:\Users\User\AppData\Local\Temp\STD5b38_000000.tmp"
browse imp_age_`wv'-imp_bmi_sd_`wv'
browse cunichah np imp_age_`wv'-imp_bmi_sd_`wv'
mi describe
save BMI_Imputation`wv'_5dec2023, replace
* !!! Collapse dataset to summarizes (means & SD's of imputed & passive
variables)
mi extract 0
keep cunichah np subhog_`wv' tipent_`wv' factor?_`wv' sex_`wv' tam_loc_`wv'
*flag_`wv' imp*
* ! Relabel variables
label variable imp_age_`wv' "Age 2021 (mean across 10 imputations)"
label variable imp_age_sd_`wv' "Age 2021 (standard deviation across 10
imputations)"
label variable imp_edu "Years of education (mean across 10 imputations)"
label variable imp_edu_sd "Years of education (standard deviation across 10
imputations)"
label variable imp_wt_`wv' "Weight (kg) 2021 (mean across 10 imputations)"
label variable imp_wt_sd_`wv' "Weight (kg) 2021 (standard deviation across 10
imputations)"
label variable imp_ht_`wv' "Height without shoes (cm) 2021 (mean across 10
imputations)"
label variable imp_ht_sd_`wv' "Height without shoes (m) 2021 (standard deviation
across 10 imputations)"
label variable imp_ht_`wv' "Height without shoes (m) 2021 (mean across 10
imputations)"
label variable imp_bmi_`wv' "BMI 2021 (mean across 10 imputations)"
label variable imp_bmi_sd_`wv' "BMI 2021 (standard deviation across 10
imputations)"
save BMI_Imputation`wv'_5dec2023, replace
* !!! Compare distributions of height, weight, and BMI among cases where such
values were observed, among those where such values were imputed, and among all
cases.
* !! Histograms
* ! Height
summarize imp_ht_`wv' if ht_flag_`wv' == 0
summarize imp_ht_`wv' if ht_flag_`wv' == 1
summarize imp_ht_`wv'

```

```

histogram imp_ht_`wv' if ht_flag_`wv' == 0, xtitle("") title("Not Imputed")
name(gr1, replace)

histogram imp_ht_`wv' if ht_flag_`wv' == 1, xtitle("") ytitle("")
title("Imputed") name(gr2)

histogram imp_ht_`wv', xtitle("") ytitle("") title("All Cases") name(gr3)

histogram imp_ht_`wv' if ht_flag_`wv' == 1, xtitle("") title("Imputed") name(gr2,
replace) nodraw

histogram imp_ht_`wv', xtitle("") title("All Cases") name(gr3, replace) nodraw

graph combine gr1 gr2 gr3, xcommon ycommon title("Height without shoes (m)",
position(6)) name(grht)

graph combine gr1 gr2 gr3, xcommon ycommon title("Height without shoes (m)",
position(6)) name(grht, replace) rows(1)

* ! Weight

histogram imp_wt_`wv' if wt_flag == 0, xtitle("") title("Not Imputed") name(gr4)
histogram imp_wt_`wv' if wt_flag == 1, xtitle("") title("Imputed") name(gr5)
histogram imp_wt_`wv', xtitle("") title("All Cases") name(gr6)

graph combine gr4 gr5 gr6, rows(1) title("Weight (kg)", position(6)) name(grwt)
graph combine gr4 gr5 gr6, rows(1) title("Weight (kg)", position(6)) name(grwt,
replace) xcommon ycommon

* ! BMI

histogram imp_bmi_`wv' if ht_flag_`wv' == 0 & wt_flag_`wv' == 0, xtitle("")
title("Not Imputed") name(gr7)

histogram imp_bmi_`wv' if ht_flag_`wv' == 1 | wt_flag_`wv' == 1, xtitle("")
title("Imputed") name(gr8)

histogram imp_bmi_`wv', xtitle("") title("All Cases") name(gr9)

graph combine gr7 gr8 gr9, xcommon ycommon rows(1) title("BMI (kg/m^2)",
position(6)) name(grbmi)

* ! Show histograms of height, weight, and BMI in one image

graph combine grht grwt grbmi, cols(1) title("2021") saving(BMI_Hist21)

graph save BMI_Hist21.png

graph save BMI_Hist21.jpg

graph drop _all

graph use "C:\Users\User\MHAS Dropbox\Matthew
Miller\ImputedBMI\2021\BMI_Hist21.gph"

* !!! Compute summary statistics of height, weight, and BMI among cases where
such values were observed, where they were imputed, and all cases.

summarize imp_ht_`wv' if ht_flag_`wv' == 0, detail
tabulate ht_flag_`wv', missing
tabulate wt_flag_`wv', missing
summarize imp_wt_`wv' if wt_flag_`wv' == 0, detail

```

```

summarize imp_wt`wv' if wt_flag`wv' == 1, detail
summarize imp_wt`wv', detail
* ! Convert heights to cm for comparability with prior waves
replace imp_ht`wv' = imp_ht`wv' * 100
replace imp_ht_sd`wv' = imp_ht_sd`wv' * 100
save BMI_Imputation`wv'_18dec2023
* !!! Compute summary statistics of height, weight, and BMI among cases where
such values were observed, where they were imputed, and all cases.
summarize imp_ht`wv' if ht_flag`wv' == 0, detail
summarize imp_ht`wv' if ht_flag`wv' == 1, detail
summarize imp_ht`wv', detail
tab1 ht_flag`wv' wt_flag`wv', missing
* ! Compare distributions of BMI's among cases where BMI was imputed and where it
wasn't, controlling for diabetic status.
* ! Load diabetes
merge 1:1 cunicah np using
"C:\Users\User\Downloads\sect_a_c_d_e_pc_f_h_i_2021(2).dta", assert(match)
keepusing(c6`wv') nogenerate
* ! Construct box plots
graph drop _all
tabulate c6`wv', missing
graph box imp_bmi`wv' if ht_flag`wv' == 0 & wt_flag`wv' == 0 &
inlist(c6`wv',1,2), over(c6`wv', relabel(1 "Diabetic" 2 "Not Diabetic"))
ytitle("BMI (kg/m^2)") title("Non-Imputed Cases") name(gr1)
graph box imp_bmi`wv' if inlist(c6`wv',1,2), over(c6`wv', relabel(1 "Diabetic"
2 "Not Diabetic")) ytitle("Mean BMI (kg/m^2)") title("All Cases") name(gr2)
* ! Construct histograms of height and weight (no BMI)
use BMI_Imputation`wv'_18dec2023
* Relabel height variables to reflect conversion from meters to centimeters
describe imp_ht*21
label variable imp_ht`wv' "Height without shoes (cm) 2021 (mean across 10
imputations)"
codebook imp_ht*21
label variable imp_ht_sd`wv' "Height without shoes (cm) 2021 (standard deviation
across 10 imputations)"
* Relabel values of flag variables
codebook *flag`wv'
label define yesno0 0 "0.No" 1 "1.Yes"
label values *flag`wv' yesno0

```

```

save BMI_Imputation`wv'_22dec2023, replace
* ! Construct histograms for methodological document
histogram imp_ht`wv', xtitle("Height (cm)") title("All Cases") name(gr3)
do "C:\Users\mrmiller\AppData\Local\Temp\STD2f6c_000000.tmp"
graph combine gr1 gr2 gr3, rows(1) xcommon ycommon name(grht)
graph combine gr4 gr5 gr6, rows(1) xcommon ycommon name(grwt)
graph combine grht grwt, cols(1) title("2021")
graph save HtWtHist21

graph save Graph "C:\Users\mrmiller\OneDrive - University of Texas Medical
Branch\Documents\MHAS\Height & Weight Imputation\2021\HtWtHist21.gph", replace
graph export "C:\Users\mrmiller\OneDrive - University of Texas Medical
Branch\Documents\MHAS\Height & Weight Imputation\2021\HtWtHist21.tif", as(tif)
replace

* !! Convert box plots of BMI by diabetic status to TIFF format (necessary for
insertion into imputation document)

* !!! Replace mean BMI's across 10 imputations with median BMI's across 10
imputations

use BMI_Imputation`wv'.dta
mi describe

mi passive: generate bmi`wv' = weight`wv' / height`wv'^2
use BMI_Imputation`wv'_5dec2023.dta, clear
use BMI_Imputation`wv'.dta

mi impute chained (pmm, knn(5)) age2`wv' education weight`wv' height`wv' =
i.sex`wv' i.tam_loc`wv', rseed(101) add(10) burnin(125) dots nomonotonechk
mi passive: generate bmi`wv' = weight`wv' / height`wv'^2
save BMI_Imputed21

* ! Calculate medians of BMI across 10 imputations
egen imp_bmi`wv' = rowmedian(*_bmi`wv')
mi register regular imp_bmi`wv'

* ! Save median BMI's
mi extract 0, clear
keep cunichah np subhog`wv' imp_bmi`wv'
label variable imp_bmi`wv' "BMI (kg/m^2) 2021 (median among 10 imputations)"
save BMI_Med`wv'

use BMI_Imputation`wv'_22dec2023
merge 1:1 cunichah np using BMI_Med`wv', assert(match match_update match_conflict)
keepusing(imp_bmi`wv') nogenerate update replace
save BMI_Imputation`wv'_22dec2023, replace

```

```

* !!! Construct box plots of BMI by diabetic status among observations where BMI
was observed & among observations where BMI was passively imputed (medians across
10 imputations in the latter case)

* ! Load diabetic status

merge 1:1 cunicah np using sect_a_c_d_e_pc_f_h_i_2021, assert(match) nogenerate
keepusing(c6_`wv`)

graph drop _all

graph box imp_bmi_`wv' if ht_flag_`wv' == 0 & wt_flag_`wv' == 0 &
inlist(c6_`wv',1,2), over(c6_`wv', relabel(1 "Diabetic" 2 "Not Diabetic"))
ytitle("BMI (kg/m2)") title("Non-Imputed Cases1")
name(gr1)

graph box imp_bmi_`wv' if inlist(c6_`wv',1,2), over(c6_`wv', relabel(1 "Diabetic"
2 "Not Diabetic")) ytitle("Median BMI (kg/m2)") title("All
Cases1") name(gr2)

graph box imp_bmi_`wv' if inlist(c6_`wv',1,2), over(c6_`wv', relabel(1 "Diabetic"
2 "Not Diabetic")) ytitle("Median BMI (kg/m2)") title("All
Cases1,2") name(gr2, replace)

graph combine gr1 gr2, ycommon title("2021") note("{superscript:1}Excludes
observations where diabetic status is missing" "{superscript:2}BMI's for imputed
cases are medians among 10 imputations")

graph save Graph BMI_DiabBoxPlots`wv'

graph use BMI_DiabBoxPlots`wv'

graph export BMI_DiabBoxPlots`wv'.tif

log close

use BMI_Imputation`wv'_22dec2023

* !!! Construct box plots of BMI by locality size among observations where BMI
was imputed and among all cases

tabulate tam_loc_`wv', missing

describe imp_bmi_`wv'

label variable imp_bmi_`wv' "BMI 2021 (median across 10 imputations)"

save BMI_Imputation`wv'_9jan2024, replace

log off

graph hbox imp_bmi_`wv' if wt_flag_`wv' == 0 & ht_flag_`wv' == 0,
over(tam_loc_`wv') ytitle("BMI (kg/m2)") title("Non-Imputed Cases")
name(gr1)

graph hbox imp_bmi_`wv', over(tam_loc_`wv') ytitle("Median BMI
(kg/m2)") title("All Cases*") name(gr2)

graph combine gr1 gr2, rows(1) ycommon title("2021") note("*BMI's for imputed
cases are medians among 10 imputations")

graph save bmi_tam_loc_box_`wv'

graph export bmi_tam_loc_box_`wv'.tif

```

```

* !!! Regress log-BMI on years of education
tab1 edu_flag_`wv' ht_flag_`wv' wt_flag_`wv', missing
* !! Cases where neither BMI nor education was imputed
generate log_bmi21 = log(imp_bmi_`wv')
regress log_bmi21 imp_edu if edu_flag_`wv' == 0 & wt_flag_`wv' == 0 &
wt_flag_`wv' == 0
* !!! Regress log-BMI on locality size among cases where BMI was not imputed
regress log_bmi21 i.tam_loc_`wv' if wt_flag_`wv' == 0 & ht_flag_`wv' == 0
regress log_bmi21 ib4.tam_loc_`wv' if wt_flag_`wv' == 0 & ht_flag_`wv' == 0
* !!! Regress log-BMI on years of education among cases where neither BMI nor
education was imputed
regress log_bmi21 imp_edu if edu_flag_`wv' == 0 & wt_flag_`wv' == 0 &
ht_flag_`wv' == 0
* !!! Regress log-BMI on diabetic status among cases where BMI was not imputed
and diabetic status isn't missing
merge 1:1 cunicah np using
"C:\Users\User\Downloads\sect_a_c_d_e_pc_f_h_i_2021(2).dta", assert(match)
keepusing(c6_`wv') nogenerate
tabulate c6_`wv', missing
regress log_bmi21 ib2.c6_`wv' if !inlist(c6_`wv',8,9) & wt_flag_`wv' == 0 &
ht_flag_`wv' == 0
* !!! Construct pooled logistic regression models of log-BMI on each of
education, locality size, and diabetic status (where diabetic status is not
missing)
use BMI_Imputed21, clear
* ! Construct log-BMI passive variable
mi passive: generate log_bmi_`wv' = log(bmi_`wv')
* ! Load diabetic status
merge 1:1 cunicah np using
"C:\Users\User\Downloads\sect_a_c_d_e_pc_f_h_i_2021(2).dta", assert(match)
keepusing(c6_`wv') nogenerate
mi register regular c6_`wv'
* !! log-BMI vs. years of education
mi estimate: regress log_bmi_`wv' education
mi estimate: regress log_bmi_`wv' ib4.tam_loc_`wv'
tab tam_loc_`wv' ht_flag_`wv', m
tab tam_loc_`wv' wt_flag_`wv', m
mi xeq 0: tab tam_loc_`wv' wt_flag_`wv', m col
mi xeq 0: tab tam_loc_`wv' wt_flag_`wv', m row
* !! log-BMI vs. diabetic status (where diabetic status is observed)

```

```

mi xeq 0: tabulate c6_`wv', missing
mi estimate: regress log_bmi_`wv' ib2.c6_`wv' if !inlist(c6_`wv',8,9)
log off

* !!! Construct binned scatter plots of log-BMI vs. years of education among
cases where both were imputed and among cases where neither was imputed
* !! Compute averages of log-BMI across 10 imputations
use BMI_Imputed21
mi passive: generate log_bmi_`wv' = log(bmi_`wv')
mi passive: replace log_bmi_`wv' = bmi_`wv' if bmi_`wv' > .
egen imp_log_bmi_`wv' = rowmean(_*log_bmi_`wv')
egen imp_log_bmi_sd_`wv' = rowstd(_*log_bmi_`wv')
mi register regular imp_log_bmi_`wv' imp_log_bmi_sd_`wv'
save BMI_Imputed21, replace
use BMI_Imputation`wv'_16jan2024
merge 1:1 cunicah np using BMI_Imputed21, nogenerate assert(match)
keepusing(imp_log_bmi_`wv' imp_log_bmi_sd_`wv')
save BMI_Imputation`wv'_16jan2024, replace
describe imp_bmi_`wv' imp_bmi_sd_`wv'
label variable imp_log_bmi_`wv' "log-transformed BMI 2021 (mean across 10
imputations)"
label variable imp_log_bmi_sd_`wv' "log-transformed BMI 2021 (standard deviation
across 10 imputations)"
save BMI_Imputation`wv'_16jan2024, replace

* !! Construct scatter plots
* ! NB: Non-imputed cases are cases where BMI was not imputed; years of education
may have been imputed in these cases.
binscatter imp_log_bmi_`wv' imp_edu if ht_flag_`wv' == 0 & wt_flag_`wv' == 0,
title("Non-Imputed Cases1") ytitle("ln(BMI)") xtitle("Years of
education") name(gr1)
binscatter imp_log_bmi_`wv' imp_edu, title("All Cases1,2")
ytitle("ln(BMI)") xtitle("Years of education") name(gr2)
graph combine gr1 gr2, xcommon ycommon title("2021") note("{1Imputed
years of education were averaged across 10 imputations" "2Imputed
log-BMI's were averaged across 10 imputations")
graph save Graph log_bmi_edu_bin_`wv'
graph export log_bmi_edu_bin_`wv'.tif

```

2024

```
log close _all

log using BMI_Imputation2024, replace

***** BMI_Imputation2024.do *****

** Purpose: To impute missing self-reported heights and weights in MHAS 2024 **
** and compute BMI's from both observed and imputed heights and weights *****
***** Version: June 2026 *****

***** STEP 1: ESTABLISH DIRECTORIES WHERE NECESSARY RAW DATA FILES ARE *****

***** STORED
*****

local wv 24 // Last two digits of year of current wave

* !!! Replace strings stored in following macros with folders where these files
* are stored in your device or where you plan on storing them

local raw01 "||Health data from MHAS 2001||"
local raw03 "||Health data from MHAS 2003||"
local raw12 "||Individual-level 2012 MHAS data ||"
local raw15 "||Individual-level 2015 MHAS data||"
local raw18 "||Individual-level 2018 MHAS data||"
local raw21 "||Individual-level 2021 MHAS data||"
local raw`wv' `"||Individual-level 20`wv' MHAS data||"'
local maestro `"||20`wv' Master Follow-Up File||"'
local graphs "||Graphs||"
local output "||Imputed datasets and logs||"

***** STEP 2: CLEAN DATA *****

*** Load raw MHAS 2024 data ***

use "`raw`wv'\sect_a_c_d_e_pc_f_h_i_20`wv'.dta"', clear

    // Individual-level raw data from current MHAS wave

*** Remove missing value codes in variables to impute ***
```

* Height

generate /// Creates new variable

height_`wv' = c67\_1\_`wv' * 100 + c67_2_`wv' /// Constructs new
variable defined by given expression

if /// Executes operation only on observations that meet given condition(s)

!(/// Negates given condition(s)

inlist(/// Checks if each observation has any of given values
for given variable

c67_1_`wv', /// Meters portion of self-reported height,
i.e. subject's height is c67_1 m & c67_2 cm

888, /// Subject refused to answer

999 /// Subject didn't know answer

) | ///

inlist(///

c67_2_`wv', /// Centimeters portion of self-reported height

888, ///

999 ///

) ///

)

label variable height_`wv' `Height without shoes (cm) 20`wv'""

* Age

recode age_`wv' (888 999 = .), generate(age2\_`wv')

// Generates new variable age2_XX identical to age_XX, except with 888

// ("Refused") and 999 ("Don't know") replaced with system missing value (.)

label variable age2_`wv' `Age in full years 20`wv'""

* Education

recode yrschool (missing = .), generate(education)

// Generates new variable education identical to yrschool, except with all

// missing values replaced with system missing

label variable education "Years of education at baseline"

```

* Weight
recode c66_`wv' (888 999 = .), generate(weight_`wv')
label variable weight_`wv' `"'Weight (kg) 20`wv'"'"'

save `"'output'/BMI_Imputation`wv'.dta"'', replace

*** Substitute system missing values for observations where height in 2024 ***
*** differs from height in any previous wave by more than 10%
***

*** Load heights from previous waves ***

* 2021
merge /// Combines dataset currently open ("master") with given dataset in
storage ("using"); merged variables attached to observations with matching values
of key variables

    1:1 /// Key variables uniquely identify observations in both master & using
datasets

        cunicah /// key variable
        np /// key variable
using ///

    `"'raw21'/sect_a_c_d_e_pc_f_h_i_2021.dta"'', ///
keep( /// Specifies which observations to retain in merged dataset
    master /// Observations appearing in master only
    match /// Observations appearing in both
) ///
keepusing( /// Specifies which variables to merge from using dataset
    c67_1_21 ///
    c67_2_21 ///
) ///

nogenerate // Does not generate variable indicating merge status (master
only, using only, or matched)
generate ///

    height_21 = c67_1_21 * 100 + c67_2_21 ///
if ///

```

```

        !( ///
            inlist( ///
                c67_1_21, ///
                888, ///
                999, ///
                . /// Missing not otherwise specified
            ) | /// Evaluates to TRUE if either or both conditions on either
side is true
            inlist( ///
                c67_2_21, ///
                888, ///
                999, ///
                . ///
            ) ///
        )

* 2018
merge ///
    1:1 ///
        cunicah ///
        np ///
using ///
    "`raw18'/sect_a_c_d_f_e_pc_h_i_2018.dta'", ///
keep( ///
    master ///
    match ///
) ///
keepusing( ///
    c67_1_18 ///
    c67_2_18 ///
) ///
nogenerate
generate ///

```

```

        height_18 = c67_1_18 * 100 + c67_2_18 ///
if ///
    !( ///
        inlist( ///
            c67_1_18, ///
            888, ///
            999, ///
            . ///
        ) | ///
        inlist( ///
            c67_2_18, ///
            888, ///
            999, ///
            . ///
        ) ///
    )

* 2015
merge ///
1:1 ///
    cunicah ///
    np ///
using ``raw15'/sect_a_c_d_e_pc_f_h_i_2015.dta'', ///
keep( ///
    master ///
    match ///
) ///
keepusing( ///
    c67_1_15 ///
    c67_2_15 ///
) ///
nogenerate
generate ///

```

```

        height_15 = c67_1_15 * 100 + c67_2_15 ///
if ///
    !( ///
        inlist( ///
            c67_1_15, ///
            888, ///
            999, ///
            .i /// Partially completed interview
        ) | ///
        c67_2_15 == .i ///
    )

* 2012
merge ///
    1:1 ///
        cunicah ///
        np ///
using "`raw12'\sect_a_c_d_e_pc_f_h_i_em_2012.dta", ///
keep( ///
    master ///
    match ///
) ///
keepusing( ///
    c67_1_12 ///
    c67_2_12 ///
) ///
nogenerate
generate ///

        height_12 = c67_1_12 * 100 + c67_2_12 ///
if ///
    inlist( ///
        c67_1_12, ///
        1, ///

```

2 ///

)

save ``output'/BMI_Imputation`wv'.dta'', replace

* ! Load identifiers needed to merge data from 2001 & 2003

merge ///

1:1 ///

cunicah ///

np ///

using ``maestro'\master_follow_up_file_20`wv'.dta", ///

assert(/// Returns error and halts program if not all observations in
master and/or using dataset meet given condition(s)

using /// Observation appears in using dataset only

match /// Observations appears in both datasets

) ///

keep(match) ///

keepusing(///

acthog ///

codent01 ///

codent03 ///

tipent_?? ///

) ///

nogenerate

save ``output'/BMI_Imputation`wv'.dta'', replace

* 2001 and 2003

tempfile c01 c03 // Creates temporary files to contain health data from MHAS 2001
& 2003 (files deleted after program halts)

preserve // Temporarily saves dataset in memory

use ``raw01'/sect_c.dta'', clear

rename (ps3 c72) (codent01 c72_01)

save `c01'

```

use ``raw03'/sect_c.dta'', clear

destring acthog, replace // Converts acthog (2003 updated household code) from
string to numeric

rename (ent2 c67) (codent03 c67_03)

save `c03'

restore // Restores data saved by above preserve command to memory


merge ///
    m:1 ///
        unhhid ///
        codent01 ///
using `c01', ///
keep( ///
    master ///
    match ///
) ///
keepusing(c72_01) ///
nogenerate

merge ///
    m:1 ///
        cunicah ///
        acthog ///
        codent03 ///
using `c03', ///
nogenerate ///
keep( ///
    master ///
    match ///
) ///
keepusing(c67_03)


generate height_03 = c67_03 if c67_03 < 888
generate height_01 = c72_01 if c72_01 < 888

```

```

save `"output'/BMI_Imputation`wv'.dta"', replace

*** Identify where heights reported in 2024 differ from those reported in ***
*** any prior wave by more than 10%
***

foreach w in 01 03 12 15 18 21 {
    // Add if tipent shows participation in both waves
    tempvar inboth // Creates temporary variable indicating whether subject par-
                    // ticipated in both 2024 and in prior wave currently
being
                    // checked for discrepancies
    if (`"w'" == "01" || `"w'" == "03") {
        // Wave currently being checked is 2001 or 2003
        generate ///
            byte ///
                `inboth' = inrange(tipent_`w',11,22) &
inrange(tipent_`wv',1,4)
    }
    else {
        generate ///
            byte ///
                `inboth' = inrange(tipent_`w',1,4) &
inrange(tipent_`wv',1,4)
    }
    quietly ///
        count ///
            if ///
                `inboth' & ///
                !( ///
                    missing(height_`wv') | ///
                    missing(height_`w') ///
                ) & ///
                abs(height_`wv' - height_`w') / height_`w' > 0.1
    local outs = `=r(N)' // Number of 2024 heights differing from height in

```

```

                                // prior wave by more than 10%

quietly count if ///
    `inboth' & ///
    !( ///
        missing(height_`wv') | ///
        missing(height_`w') ///
    )

local hts = `=r(N)' // Total number of observations with reported heights in
                    // both 2024 and prior wave

local pct : display %3.1f `='outs'/'`hts'*100'
            // Percentage of observations with self-reported heights in both
            // 2024 and prior wave that differ by more than 10% of height in
            // prior wave

display `"Heights in 20`wv' differing from reported height in 20`w' by more
than 10%: N = `outs' (`pct'% of observations not missing height in either wave)'"
replace ///
    height_`wv' = . ///
if ///
    `inboth' & ///
    !( ///
        missing(height_`wv') | ///
        missing(height_`w') ///
    ) & ///
    abs(height_`wv' - height_`w') / height_`w' > 0.1
drop `inboth'
}

*** Construct BMI's from observed heights and weights ***
generate ///
    bmi_`wv' = weight_`wv' / (height_`wv' / 100)^2 ///
if ///
    !( ///
        missing(height_`wv') | ///

```

```

        missing(weight_`wv') ///
    )

label variable bmi_`wv' `"'Body mass index (kg/m^2) 20`wv'"'

*** Construct flags indicating where each imputed variable is missing ***
// capture label define yesno 0 "0.No" 1 "1.Yes"

* Age
generate byte age_flag_`wv' = missing(age2_`wv')
label variable age_flag_`wv' "Age was imputed 20`wv'"
label values age_flag_`wv' yesno2

* Education
generate byte edu_flag = missing(education)
label variable edu_flag "Years of education were imputed 20`wv'"
label values edu_flag yesno2

* Weight
generate byte wt_flag_`wv' = missing(weight_`wv')
label variable wt_flag_`wv' "Weight (kg) was imputed 20`wv'"
label values wt_flag_`wv' yesno2

* Height
generate byte ht_flag_`wv' = missing(height_`wv')
label variable ht_flag_`wv' "Height w/o shoes (cm) was imputed 20`wv'"
label values ht_flag_`wv' yesno2

* BMI
generate byte bmi_flag_`wv' = missing(bmi_`wv')
label variable bmi_flag_`wv' `"'BMI (kg/m^2) was imputed 20`wv'"'
label values bmi_flag_`wv' yesno2

save `"'output'/BMI_Imputation`wv'.dta"', replace

```

```

*** Show distributions of variables to impute prior to imputation ***

summarize /// Displays summary statistics of given variables, assumed to be
continuous

    age2_`wv' ///
    education ///
    height_`wv' ///
    weight_`wv' ///
    bmi_`wv', ///

    detail // Displays various percentiles in addition to means, medians, SD's
and extrema

* Construct bar graph showing proportion of missing values of height and weight
* in 2024

graph ///

    bar ///

        (mean) /// Heights of bars are means of given variables

            ht_flag_`wv' /// Binary (0/1) indicator that height was missing
(mean is proportion of observations where height was missing)

            wt_flag_`wv', /// Binary (0/1) indicator that weight was missing

ytitle("Proportion Missing") /// y-axis title

title("2024") /// Main title of graph

legend(label(1 "Height") label(2 "Weight")) ///

blabel(bar, format("%05.3f")) // Displays bar height above bars

graph save ``graphs'/ht_wt_miss_bar_20`wv'.gph'', replace

graph export ``graphs'/ht_wt_miss_bar_20`wv'.jpg'', replace

    // "graph export" command needed to save graphs in formats other than
Stata's native .gph format

```

***** STEP 3: IMPUTE MISSING DATA *****

*** Prepare data for imputation ***

```

mi set wide // Declares dataset in memory as data to be imputed where missing,
using mutiple imputation using chained equations (MICE); "wide" tells Stata to
store imputed data in separate variables

mi register regular /// Variables without missing values, including missingness
flags

    unhhid ///
    cunicah ///
    np ///
    subhog_`wv' ///
    tipent_`wv' ///
    factor?_`wv' ///
    acthog ///
    codent01 ///
    codent03 ///
    sex_`wv' ///
    tam_loc_`wv' ///
    age_flag_`wv' ///
    edu_flag ///
    ht_flag_`wv' ///
    wt_flag_`wv' ///
    bmi_flag_`wv'

mi register imputed /// Variables with missing data to be imputed
    age2_`wv' ///
    education ///
    weight_`wv' ///
    height_`wv'

mi register passive bmi_`wv' // Variables to be computed from imputed variables
//    age_`wv' ///
//    yrschool
// mi unregister age_`wv' yrschool

*** Perform imputations ***

* ! Check if imputed variables have monotone missing pattern

```

```

mi ///
    misstable ///
        nested ///
            age2_`wv' ///
            education weight_`wv' ///
            height_`wv'

* !! Perform imputations

mi impute chained /// Imputes missing data with MICE
    (pmm, knn(5)) /// Given variables imputed using predictive mean matching
    (PMM) with five nearest neighbors (see descriptive document)
        age2_`wv' ///
        education ///
        weight_`wv' ///
        height_`wv' ///
    = /// Following variables are complete variables used as imputation
    covariates
        i.sex_`wv' ///
        i.tam_loc_`wv', ///
    rseed(101) /// Seed for pseudorandom number generation (specified for
    reproducibility)
    add(10) /// Creates 10 imputed datasets
    burnin(75) /// Indicates maximum number of times to execute chained
    equations per imputation
    dots /// Displays progress of "mi imputed chained" using dots (1 dot = 1
    imputation)
    nomonotonechk // Does not check whether raw data have monotone missing
    pattern (saves time; checked earlier in do-file)
    save "`output'/BMI_Imputation`wv'.dta"', replace

*** Construct BMI as passively imputed variable ***

mi passive: /// Performs command on all imputed datasets, as well as unimputed
data; values of passive variables automatically updated whenever values in any
given dataset(s), imputed or otherwise, change

    replace bmi_`wv' = weight_`wv' / (height_`wv' / 100)^2

```

```

*** Compute means and standard deviations of imputed variables, and medians ***
*** and standard deviations of BMI across 10 imputations
***

* Weight, height, and age
tokenize "wt ht age" // Stores "wt", "ht", & "age" in local macros `1', `2', &
                        // `3', respectively

local i = 1
foreach y of varlist weight_`wv' height_`wv' age2_`wv' {
    egen imp_``i'_'_wv' = rowmean(_*_`y')
        // Constructs variables containing between-imputation means of weight,
height, & age
    egen imp_``i'_'_sd_`wv' = rowsd(_*_`y')
        // Constructs variables containing between-imputation SD's of weight,
height, & age
    mi ///
        register ///
        regular ///
            imp_``i'_'_wv' ///
            imp_``i'_'_sd_`wv'

    local i = `i' + 1
}

* Education
egen imp_edu = rowmean(_*_education)
egen imp_edu_sd = rowsd(_*_education)

* BMI
egen imp_bmi_`wv' = rowmedian(_*_bmi_`wv')
    // Constructs variable containing between-imputation medians of BMI
egen imp_bmi_sd_`wv' = rowsd(_*_bmi_`wv')
mi register regular imp_bmi*`wv'
save "`output'/BMI_Imputation`wv'.dta"', replace

```

* Reduce dataset to summary statistics (means & SD's of height & weight, medians
* & SD's of BMI)

mi extract 0, clear // Drops imputed datasets, leaving only un-imputed data, in-
// cluding passive and regular variables

keep /// Drops all variables except those specified by command

cunichah ///

np ///

codent0? ///

subhog_`wv' ///

tipent_`wv' ///

factor?_`wv' ///

sex_`wv' ///

age_`wv' ///

imp_edu /// Retained for graphs; not in final dataset

tam_loc_`wv' *flag* ///

imp_ht*`wv' ///

imp_wt*`wv' ///

imp_bmi*`wv'

order /// Rearranges variables in memory in given order

cunichah ///

np ///

codent0? ///

subhog_`wv' ///

tipent_`wv' ///

factor?_`wv' ///

sex_`wv' ///

age_`wv' ///

imp_edu ///

tam_loc_`wv' ///

imp_ht*`wv' ///

ht_flag_`wv' ///

imp_wt*`wv' ///

wt_flag_`wv' ///

```

    imp_bmi*`wv' ///
    bmi_flag_`wv'

* Label summary variables=
label variable imp_edu "Years of education (mean across 10 imputations)"=
label variable imp_wt_`wv' "Weight (kg) 20`wv' (mean across 10 imputations)"
label variable imp_wt_sd_`wv' "Weight (kg) 20`wv' (standard deviation across 10
imputations)"
label variable imp_ht_`wv' "Height without shoes (cm) 20`wv' (mean across 10
imputations)"
label variable imp_ht_sd_`wv' "Height without shoes (cm) 20`wv' (standard
deviation across 10 imputations)"
label variable imp_bmi_`wv' "BMI 20`wv' (median across 10 imputations)"
label variable imp_bmi_sd_`wv' "BMI 20`wv' (standard deviation across 10
imputations)"
save "`output'/BMI_Imputation`wv'_Final.dta"', replace

***** STEP 4: ANALYZE IMPUTED DATA *****

*** Compare distributions of height, weight, and BMI among cases where such ***
*** values were observed, where such values were imputed, and all cases
***

* !! Histograms

* ! Height
summarize imp_ht_`wv' if ht_flag_`wv' == 0, detail // Non-imputed cases
summarize imp_ht_`wv' if ht_flag_`wv' == 1, detail // Imputed cases
summarize imp_ht_`wv', detail // All cases

* Non-imputed cases
quietly count if ht_flag_`wv' == 0 // Stores number of observations where self-
// reported height was observed in
r(N)
local nimp = string(`r(N)',"%6.0fc") // Cases where height was not imputed

```

```

histogram ///
    imp_ht_`wv' ///
if ///
    ht_flag_`wv' == 0, ///
xtitle("") /// x-axis label (unlabeled in this case)
title(`"Not Imputed ({it:N} = `nimp')"'') ///
name(gr1, replace) /// Stores graph in gr1, replacing graph already stored
in gr1 if applicable
nodraw

* Imputed cases
quietly count if ht_flag_`wv' == 1
local imp = string(`r(N)', "%6.0fc") // Cases where height was imputed
histogram ///
    imp_ht_`wv' ///
if ///
    ht_flag_`wv' == 1, ///
xtitle("") /// ytitle("") ///
title(`"Imputed ({it:N} = `imp')"'') ///
name(gr2, replace) ///
nodraw

* All cases
local all`wv' = string(`c(N)', "%6.0fc") // Total number of cases
histogram ///
    imp_ht_`wv', ///
xtitle("") /// ytitle("") ///
title(`"All Cases ({it:N} = `all`wv')"'') ///
name(gr3, replace) ///
nodraw

* Combine histograms of height among non-imputed, imputed, and all cases
graph ///

```

```

combine /// Combines given graphs into one image
    gr1 ///
    gr2 ///
    gr3, ///
xcommon /// Rescales graphs to have same x-axis scale
ycommon /// Rescales graphs to have same y-axis scale
title( ///
    "Height without shoes (cm)", ///
    position(6) /// Places title on bottom of combined graph
) ///
name(grht, replace) ///
rows(1)

* ! Weight
summarize imp_wt_`wv' if wt_flag_`wv' == 0, detail // Non-imputed cases
summarize imp_wt_`wv' if wt_flag_`wv' == 1, detail // Imputed cases
summarize imp_wt_`wv', detail // All cases

* Non-imputed cases
quietly count if wt_flag_`wv' == 0
local nimp = string(`r(N)', "%6.0fc") // Cases where weight was not imputed
histogram ///
    imp_wt_`wv' ///
if ///
    wt_flag_`wv' == 0, ///
xtitle("") ///
title(`"Not Imputed ({it:N} = `nimp')"'') ///
name(gr4, replace) ///
nodraw

* Imputed cases
quietly count if wt_flag_`wv' == 1
local imp = string(`r(N)', "%6.0fc") // Cases where weight was imputed

```

```

histogram ///
    imp_wt_`wv' ///
if ///
    wt_flag_`wv' == 1, ///
xtitle("") /// ytitle("") ///
title(`"Imputed ({it:N} = `imp')"'') ///
name(gr5, replace) ///
nodraw

```

* All cases

```

histogram ///
    imp_wt_`wv', ///
xtitle("") /// ytitle("") ///
title(`"All Cases ({it:N} = `all`wv')"'') ///
name(gr6, replace) ///
nodraw

```

* Combine histograms of weight among non-imputed, imputed, and all cases

```

graph ///
    combine ///
    gr4 ///
    gr5 ///
    gr6, ///
xcommon ///
ycommon ///
title( ///
    "Weight (kg)", ///
    position(6) ///
) ///
name(grwt, replace) ///
rows(1)

```

* ! BMI

```

summarize imp_bmi_`wv' if bmi_flag_`wv' == 0, detail // Not imputed
summarize imp_bmi_`wv' if bmi_flag_`wv' == 1, detail // Imputed
summarize imp_bmi_`wv', detail // All cases

* ! Combine histograms of height and weight in one image
graph ///
    combine ///
        grht ///
        grwt, /// grbmi, ///
    cols(1) ///
    title("20`wv'") ///
    saving("`graphs'/BMI_Hist`wv'.gph'", replace) ///
    note("Imputed heights and weights are means across 10 imputations")
graph export "`graphs'/BMI_Hist`wv'.jpg'", replace
graph drop _all

*** Compare distributions of BMI by imputation and diabetic status ***

* ! Load diabetes indicator
merge ///
    1:1 ///
        cunicah ///
        np ///
    using "`raw`wv'"/sect_a_c_d_e_pc_f_h_i_2024.dta", ///
    assert(match) ///
    keepusing(c6_`wv') ///
    nogenerate

* ! Construct box plots

* Non-imputed cases
quietly ///

```

```

count ///
    if ///
        ht_flag_`wv' == 0 & ///
        wt_flag_`wv' == 0 & ///
        inlist(c6_`wv',1,2)
local nimp = string(`r(N)',"%6.0fc")
    // Cases where BMI was not imputed (i.e., neither height nor weight
was
    // imputed) and diabetic status was not missing
quietly ///
count ///
    if ///
        ht_flag_`wv' == 0 & ///
        wt_flag_`wv' == 0 & ///
        c6_`wv' == 1
local nimpdiab = string(`r(N)',"%6.0fc") // Non-imputed cases with diabetes
quietly ///
count ///
    if ///
        ht_flag_`wv' == 0 & ///
        wt_flag_`wv' == 0 & ///
        c6_`wv' == 2
local nimpndiab = string(`r(N)',"%6.0fc") // Non-imputed cases without diabetes
graph ///
    box /// Constructs vertical box plots (categories on horizontal axis,
continuous values on vertical axis)
        imp_bmi_`wv' ///
    if ///
        ht_flag_`wv' == 0 & ///
        wt_flag_`wv' == 0 & ///
        inlist(c6_`wv',1,2), ///
over( /// Constructs separate box plot for each category of given variable
        c6_`wv', ///
relabel( /// Sets label for each box plot

```

```

        1 "Diabetic ({it:N} = `nimpdiab')" ///
        2 "Not Diabetic ({it:N} = `nimpndiab')" ///
    ) ///
) ///
ytitle("BMI (kg/m2)") ///
title(`"Non-Imputed Cases ({it:N} = `nimp')2"') ///
name(gr1, replace) ///
nodraw

* All cases
quietly count if inlist(c6_`wv',1,2)
local alldiab = string(`r(N)',"%6.0fc") // Total cases with non-missing diabetic
                                         // status

quietly count if c6_`wv' == 1
local diab : display %6.0fc `r(N)' // Total cases with diabetes
quietly count if c6_`wv' == 2
local ndiab : display %6.0fc `r(N)' // Total cases without diabetes
graph box imp_bmi_`wv' if inlist(c6_`wv',1,2), ///
    over(c6_`wv', ///
        relabel( ///
            1 "Diabetic ({it:N} = `diab')" ///
            2 "Not Diabetic ({it:N} = `ndiab')" ///
        ) ///
    ) ///
ytitle("Median BMI (kg/m2)") ///
title(`"All Cases ({it:N} = `alldiab')2,3"') ///
name(gr2, replace) ///
nodraw

* Combine box plots of BMI among non-imputed and all cases
graph combine gr1 gr2, ///
    ycommon ///
    title("BMI (20`wv') by Imputation and Diabetic Status1") ///

```

```

saving(```graphs'/BMI_DiabBox`wv'.gph'', replace) ///

note( /// Each separate string will appear in separate line in text box

      "{superscript:1}Lower and upper edges of boxes are 25th and 75th
percentiles of BMI, respectively, within group; line inside" ///

      "{tab}box is median BMI within group; lower and upper lines
outside boxes are least and greatest BMI's," ///

      "{tab}respectively, not considered outliers" ///

      "{superscript:2}Excludes observations where diabetic status is
missing" ///

      "{superscript:3}BMI's for imputed cases are medians across 10
imputations" ///

)

graph export ````graphs'/BMI_DiabBox`wv'.jpg'', replace
graph drop _all
drop c6_`wv'

*** Compare distributions of BMI by imputation status and locality size ***

* Non-imputed cases
graph ///

  hbox /// Constructs horizontal box plots (continuous values on horizontal
axis, categories on vertical axis)

    imp_bmi_`wv' ///

  if ///

    bmi_flag_`wv' == 0, ///

  over(tam_loc_`wv') ///

  ytitle("BMI (kg/m{superscript:2})") ///

  title("Non-Imputed Cases") ///

  name(gr1)

* All cases
graph ///

  hbox ///

    imp_bmi_`wv', ///

  over(tam_loc_`wv') ///

```

```

    ytitle("Median BMI (kg/m2)") ///
    title("All Cases*") ///
    name(gr2)

* Combine box plots of non-imputed and all cases
graph ///
    combine ///
        gr1 ///
        gr2, ///
    rows(1) ///
    ycommon ///
    title("`20`wv'") ///
    note("*BMI's for imputed cases are medians among 10 imputations")
graph save "`graphs'/bmi_tam_loc_box_`wv'.gph"
graph export "`graphs'/bmi_tam_loc_box_`wv'.jpg"

*** Construct binned scatter plots of log-BMI vs. years of education among ***
*** cases where both were imputed and among cases where neither was imputed ***

* !! Compute averages of log-BMI across 10 imputations
tempfile bmi_imp0 // Temporary file to store pre-imputation data + summaries of
                    // imputed values
preserve
use "`output'/BMI_Imputation`wv'.dta"', clear

* ! Compute log-BMI in each imputation
mi passive: generate log_bmi_`wv' = log(bmi_`wv')
mi passive: replace log_bmi_`wv' = bmi_`wv' if bmi_`wv' > .
    // Hard missing values (.a,...,.z in BMI replicated in log-BMI)

* ! Compute means across imputations
egen imp_log_bmi_`wv' = rowmean(_*_log_bmi_`wv')
label ///

```

```

variable ///
    imp_log_bmi_`wv' ///
        "log-transformed BMI 20`wv' (mean across 10 imputations)"
mi register regular imp_log_bmi_`wv'
mi extract 0, clear
save `bmi_imp0'

* Combine log-BMI with other averaged imputed values
restore
merge ///
    1:1 ///
        cunicah ///
        np ///
using `bmi_imp0', ///
nogenerate ///
assert(match) ///
keepusing(imp_log_bmi_`wv')

* !! Construct scatter plots

* ! NB: Non-imputed cases are cases where BMI was not imputed; years of education
may have been imputed in these cases.

* Non-imputed cases
quietly count if bmi_flag_`wv' == 0
local nimp = string(`r(N)', "%6.0fc") // Cases where BMI was not imputed
binscatter ///
    imp_log_bmi_`wv' ///
    imp_edu ///
if bmi_flag_`wv' == 0, ///
title(`"Non-Imputed Cases ({it:N} = `nimp'){superscript:1}"') ///
yttitle("ln(BMI)") ///
xttitle("Years of education") ///
name(gr1) ///

```

```

nodraw

* All cases
binscatter ///
    imp_log_bmi_`wv' ///
    imp_edu, ///
    title(`"All Cases ({it:N} = `all`wv'){superscript:1,2}"') ///
    ytitle("ln(BMI)") ///
    xtitle("Years of education") ///
    nodraw ///
    name(gr2)

* Combine graphs of non-imputed and all cases
graph ///
    combine ///
    gr1 ///
    gr2, ///
    xcommon ///
    ycommon ///
    title(`"20`wv"'') ///
    note( ///
        "{superscript:1}Imputed years of education were averaged across 10
imputations" ///
        "{superscript:2}Imputed log-BMI's were averaged across 10 imputations"
    ///
    )
graph save Graph ``graphs'/log_bmi_edu_bin_`wv'.gph''
graph export ``graphs'/log_bmi_edu_bin_`wv'.jpg''

* ! Delete means of imputed education from dataset
drop imp_edu
save ``output'/BMI_Imputation`wv'_Final.dta'', replace
log close

```

```
* ! Convert Stata log to plain text file

translate ///
    ``output'/BMI_Imputation20`wv'.smcl"' ///
    ``output'/BMI_Imputation20`wv'.log"', ///
    replace
// EOF
```