2024 Residential Energy Consumption Survey: User's Guide to the Public Use Microdata File

May 2026

Independent Statistics and Analysis
U.S. Energy Information Administration

www.eia.gov
U.S. Department of Energy
Washington, DC 20585

Table of Contents

Table of Figures

Overview

EIA makes a public-use microdata file available for each cycle of the *Residential Energy Consumption Survey* (RECS). The 2024 file, available in both SAS and CSV formats, allows users to conduct detailed analysis of home energy characteristics, as well as energy consumption and expenditures. This document provides a brief overview of the RECS design and variance estimation. It also offers illustrative examples and guidance to assist users working with the RECS microdata.

The sample was not designed to estimate all survey variables at the state level. Please use discretion when interpreting results from the microdata as insufficient sample sizes may lead to unreliable estimates.

This initial version of the User Guide focuses on the estimation of household characteristics. An updated User Guide, with examples on estimating energy consumption and expenditure, will be published at a later date. The release will be accompanied by an updated microdata file, which will include more detailed characteristics such as the types of fuel used for space heating and water heating, along with energy consumption and expenditure variables.

RECS sample design

We designed the 2024 RECS sample to estimate energy characteristics, consumption, and expenditures for the national stock of occupied housing units and the people who live in them, at both the national level and state level. In addition, we added precision requirements for 10 geographically diverse Metropolitan Statistical Areas (MSAs) to allow EIA to explore small area estimation, contingent upon availability of resources. The sample allocation at the MSA level was not available in previous survey cycles. To produce estimates at different geographical levels in the 2024 RECS, we weighed the sampled housing units to represent the total in-scope population. In this setting, a housing unit's weight indicates the number of housing units that the particular household represents.

As part of the weighting process, EIA first calculated a base sampling weight for each sampled housing unit, which is the reciprocal of the probability of that unit being selected for the RECS sample. EIA then adjusted the base weights to account for survey nonresponse and eligibility. In addition, EIA used poststratification adjustments to ensure that the RECS weights add up to U.S. Census Bureau estimates of the stratum-level number of occupied housing units for 2024. The variable NWEIGHT in the data file represents the final sampling weight, accounting for different probabilities of selection, rates of response, and adjustment for the U.S. Census Bureau housing unit estimates. NWEIGHT is the number of households in the population that the observation represents. For example, if NWEIGHT for a household is 10,000, then that household represents itself and 9,999 other households in the population that either were not sampled or were sampled but did not respond to the survey. More details about the sample design and weighting adjustments are available in the 2024 RECS Household Characteristics Technical Documentation Summary.

Sampling error and relative standard error (RSE)

Estimates from the RECS are subject to sampling error, because they are based on a sample rather than a census of the entire population.

Standard errors are used with survey estimates to measure relative amounts of sampling error, construct confidence intervals, or perform hypothesis tests. Similar to previous RECS, the 2024 RECS data tables include weighted estimates and RSEs. An RSE is formulated as the standard error (square root of the sampling variance) of a survey estimator, divided by the survey estimate, and multiplied by 100. In other words, the RSE quantifies how much the estimator varies over all possible samples that could have been selected from the population using the same sample design, relative to the corresponding survey estimate, and expressed as a percentage. The smaller the RSE, the more precise a survey estimate is in terms of its sampling variability. Each estimate in the RECS tables is accompanied by an RSE, which is provided in both CSV and PDF formats. Estimates greater than zero but with a corresponding RSE denoted by P indicate that an estimate is equal to the control total that was used in poststratification in the weighting adjustment process. The control totals are based on the 2024 American Community Survey (ACS) one-year estimates[1].

Instructions for calculating RSEs for microdata analysis in SAS, R, and Python software are shown below. Note that an RSE can be calculated by multiplying the coefficient of variation (CV) by 100 in the SAS/STAT statistical software.

Jackknife method of estimating standard error

The 2024 RECS uses the Jackknife method to first produce replicate weights to calculate standard errors of an estimate of interest. This method then uses replicate weights to repeatedly estimate the statistic of interest from each of multiple replicate samples generated from the full sample and calculates the differences between these estimates and the full-sample estimate. We constructed 60 Jackknife replicates to produce variance estimates for univariate statistics with 59 nominal degrees of freedom. The mathematical formula for the variance estimation is expressed below (See Lohr, S.L. (2010) for more technical details).

If θ is a population parameter of interest, let $\widehat{\theta}$ be the estimate from the full sample for θ. Let $\widehat{\theta} r$ be an estimator used for the r-th replicate, and R is the number of the replicate weights, the variance of $\widehat{\theta}$ is estimated by:

$$\hat{V}(\hat{\theta}) = (\frac{R-1}{R}) \sum_{r=1}^{R} (\hat{\theta}\mathrm{r} - \hat{\theta})^2$$

[1] For more information on the ACS one-year data products, please refer to the U.S. Census Bureau website.

The formula for calculating the RSE is:

$$\left(\frac{\sqrt{\hat{V}(\hat{\theta})}}{\hat{\theta}}\right) X\ 100$$

Examples: Using Final Weights (NWEIGHT) and Replicate Weights to Calculate Estimates and RSEs

The following instructions are examples for calculating any RECS estimate using the final weights (NWEIGHT) and the associated RSE using the replicate weights (NWEIGHT1 – NWEIGHT60). Software packages such as SAS, R, Stata, SUDAAN, and Python can process replicate weights to calculate RSEs. We provided instructions for Excel users and users with access to SAS/STAT, R, and Python. Note that the version and components of SAS/STAT used could affect the analysis capability; examples used below were done in SAS/STAT 15.3., R version 4.4.1, and Python version 3.12.4. We show how to compute point estimates using Excel, but Excel does not have a built-in function that calculates RSEs directly using replicate weights. We recommend calculating standard errors or RSEs using the supplied replicate weights in conjunction with estimates to account for sampling errors.

For Excel users (estimates only, no RSEs)

Excel Example 1: Calculate the total of households that have 50% or more LED indoor bulbs in the U.S. (Table HC5.2)

LGTINLED represents the portion of indoor light bulbs that are LED, and it has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

A simple count of households can be estimated using the sum of NWEIGHTs for a specified subset of cases within the RECS data file. For this example:

Step 1. Filter the file for all cases where households have at least 50% or more LED indoor bulbs. There are 12,679 cases where LGTINLED equals to 1, 2, and 3.

Step 2. Sum the NWEIGHT column for these 12,679 cases.

Answer: The weighted total of households having at least 50% or more LED indoor bulbs is 99,127,702 (48,705,711+34,266,932+16,155,058). This represents approximately 75% of all homes (99.18 million out of 132.54 million total weighted homes).

Figure 1. 2024 RECS Table HC5.2 Snapshot for LGTINLED Estimates

Preliminary data release date: March 2026

Table HC5.2 Lighting in U.S. homes, by owner or renter status, 2024

	Number of housing units (million)			Housing unit type			
				Single-family and mobile home		Apartment	
	Total U.S.[a]	Own	Rent[b]	Own	Rent[b]	Own	Rent[b]
All homes	132.54	89.54	43.00	84.76	13.03	4.78	29.97
LED bulbs used indoors							
Yes, at least one LED bulb	118.89	83.24	35.65	79.01	11.37	4.24	24.28
All bulbs are LED	48.71	32.47	16.24	30.82	4.55	1.65	11.69
Most bulbs are LED	34.27	26.24	8.03	25.03	2.82	1.20	5.21
About half of bulbs are LED	16.16	11.58	4.57	10.99	1.63	0.59	2.95
Some bulbs are LED	19.76	12.96	6.80	12.16	2.37	0.80	4.43
No LED bulbs	13.65	6.29	7.35	5.76	1.67	0.54	5.69

Data source: *2024 Residential Energy Consumption Survey* (RECS) Table HC5.2 Lighting in U.S. homes, by owner and renter status

For SAS users

SAS Example 1 (National Household Characteristics Estimation)

Objective: Calculate the total of households (with the estimated percentage) and RSE that have 50% or more LED indoor bulbs in the U.S. (Table HC5.2)

LGTINLED represents the portion of indoor light bulbs that are LED, and it has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

Step 1. After reading in the microdata file to SAS, derive a new variable LEDGE50PCT by combining categories 1, 2, and 3 from LGTINLED to indicate a household with at least 50% of indoor LED bulbs.

```
/*Read in the microdata file in SAS format*/
libname recs24 '< location where file is stored >';

   DATA RECS2024_PUBLIC;
      SET recs24.RECS2024_PUBLIC_V1;
      IF LGTINLED in (1,2,3) THEN LEDGE50PCT="Yes";
      ELSE LEDGE50PCT="No";
   RUN;
```

Step 2. Use the `PROC SURVEYFREQ` procedure with the `VARMETHOD`, `WEIGHT`, and `REPWEIGHTS` statements to obtain sampling errors associated with the estimates. `JACKKNIFE` or `JK` can be used as the `VARMETHOD` option for the Jackknife variance estimation method. The Jackknife coefficient can be customized using the `JKCOEFS` option, the coefficient for the 2024 RECS Jackknife is 59/60, which is also the default value in the procedure; therefore, it does not need to be specified. The `TABLES` statement can output statistics for multiple variables at the same time. In addition, for the population total estimate, the `CLWT` and `CVWT` options, respectively, provide the 95% confidence limits and the coefficient of variation (`CV`). Similarly, in obtaining the confidence limits and coefficient of variation for the percentages (proportions) associated with each category, use the `CL` and `CV` options.

```
PROC SURVEYFREQ DATA=RECS2024_PUBLIC VARMETHOD=JK;
      REPWEIGHTS NWEIGHT1-NWEIGHT60;
      WEIGHT NWEIGHT;
      TABLES LGTINLED LEDGE50PCT/CLWT CVWT CL CV;
RUN;
```

Answer. The estimated number of households with 50% or more indoor LED bulbs is 99,127,702, as shown in the table output from either the LGTINLED or the LEDGE50PCT variables in Figure 2. The corresponding RSE for the estimate is 0.0043 (CV)*100=0.43%; or you can also calculate the RSE using the standard error of the weighted frequency, which is (426,121/99,127,702)*100=0.43%. In other words, the RSE is less than 1% of the estimated total population, a relatively small amount, indicating that the estimate is very precise. In addition, using the Percent column, it is estimated that about 75% of households have at least 50% indoor LED bulbs.

Figure 2. PROC SURVEYFREQ Output of LGTINLED and LEDGE50PCT in SAS at the U.S. Level

Portion of inside light bulbs that are LED

LGTINLED	Frequency	Weighted Frequency	Std Err of Wgt Freq	95% Confidence Limits for Wgt Freq		CV for Wgt Freq	Percent	Std Err of Percent	95% Confidence Limits for Percent		CV for Percent
0	1550	13646346	327661	12990927	14301764	0.0240	10.2963	0.2472	9.8017	10.7908	0.0240
1	6084	48705711	484074	47737418	49674004	0.0099	36.7488	0.3652	36.0182	37.4794	0.0099
2	4516	34266933	511602	33243576	35290289	0.0149	25.8546	0.3860	25.0825	26.6268	0.0149
3	2079	16155058	380447	15394051	16916066	0.0235	12.1891	0.2871	11.6149	12.7633	0.0235
4	2447	19762787	380191	19002291	20523283	0.0192	14.9112	0.2869	14.3374	15.4850	0.0192
Total	16676	132536835	0.07170	132536835	132536835	0.0000	100.0000				

Table of LEDGE50PCT

LEDGE50PCT	Frequency	Weighted Frequency	Std Err of Wgt Freq	95% Confidence Limits for Wgt Freq		CV for Wgt Freq	Percent	Std Err of Percent	95% Confidence Limits for Percent		CV for Percent
No	3997	33409133	426121	32556764	34261501	0.0128	25.2074	0.3215	24.5643	25.8506	0.0128
Yes	12679	99127702	426121	98275334	99980071	0.0043	74.7926	0.3215	74.1494	75.4357	0.0043
Total	16676	132536835	0.07170	132536835	132536835	0.0000	100.0000				

SAS Example 2 (State Level Household Characteristics Estimation)

Objective: Calculate the totals of households (with the estimated percentage) and RSEs that have 50% or more LED indoor bulbs in Alabama and Alaska (Table STelectronicsAndLighting_2024).

LGTINLED represents the portion of indoor light bulbs that are LED, and it has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

Step 1. After reading in the microdata file to SAS, derive a new variable LEDGE50PCT by combining categories 1, 2, and 3 from LGTINLED to indicate a household with at least 50% of indoor LED bulbs.

```
/*Read in the microdata file in SAS format*/
libname recs24 '< location where file is stored >';

   DATA RECS2024_PUBLIC;
      SET RECS2024_PUBLIC_V1;
      IF LGTINLED in (1,2,3) THEN LEDGE50PCT="Yes";
      ELSE LEDGE50PCT="No";
   RUN;
```

Step 2. Use the `PROC SURVEYFREQ` procedure with the `VARMETHOD`, `WEIGHT`, and `REPWEIGHTS` statements to obtain sampling errors associated with the estimates. `JACKKNIFE` or `JK` can be used in the `VARMETHOD` option for the Jackknife variance estimation method. The Jackknife coefficient can be customized using the `JKCOEFS` option, the coefficient for the 2024 RECS Jackknife is 59/60, which is also the default value in the procedure; therefore, it does not need to be specified. The `TABLES` statement can output statistics for multiple variables at the same time. In addition, for the population total estimate, the `CLWT` and `CVWT` options, respectively, provide the 95% confidence limits and the coefficient of variation (`CV`). Similarly, in obtaining the confidence limits and coefficient of variation for the percentages (proportions) associated with each category, use the `CL` and `CV` options. The `ROW` option requests that percentages of each category of LEDGE50PCT be calculated for each level of state_name. The `NOCELLPCT` option suppresses the overall cell percentages and their standard errors.

```
   PROC SURVEYFREQ DATA=RECS2024_PUBLIC VARMETHOD=JK;
         REPWEIGHTS NWEIGHT1-NWEIGHT60;
         WEIGHT NWEIGHT;
         TABLES state_name*LEDGE50PCT/CLWT CVWT CL CV ROW NOCELLPCT;
         WHERE state_name in ("Alabama", "Alaska");
   RUN;
```

Answer. The estimated number of households with 50% or more indoor LED bulbs in Alabama is 1,390,258, and 221,941 in Alaska, as shown in Figure 3. The corresponding RSEs for the estimates are 0.0415 (CV)*100=4.15% in Alabama, and 3.36% in Alaska. You can also calculate the RSE using the standard error of the weighted frequency; for example, in Alabama, it is (57,664/1,390,258)*100=4.15%. In addition, about 68% of households in Alabama and 81% of households in Alaska have at least 50% indoor LED bulbs.

Figure 3. PROC SURVEYFREQ Output of LEDGE50PCT in SAS for Alabama and Alaska

Table of state_name by LEDGE50PCT															
state_name	LEDGE50PCT	Frequency	Weighted Frequency	Std Err of Wgt Freq	95% Confidence Limits for Wgt Freq		CV for Wgt Freq	95% Confidence Limits for Percent		CV for Percent	Row Percent	Std Err of Row Percent	95% Confidence Limits for Row Percent		CV for Row Percent
Alabama	No	92	661737	57664	546392	777082	0.0871	23.4968	33.4173	0.0871	32.2485	2.8101	26.6273	37.8696	0.0871
	Yes	209	1390258	57664	1274913	1505603	0.0415	54.8258	64.7463	0.0415	67.7515	2.8101	62.1304	73.3727	0.0415
	Total	301	2051995	1.96874E-8	2051995	2051995	0.0000	88.2431	88.2431	0.0000	100.0000				
Alaska	No	50	51452	7446	36557	66347	0.1447	1.5721	2.8532	0.1447	18.8198	2.7237	13.3717	24.2680	0.1447
	Yes	202	221941	7446	207046	236836	0.0336	8.9037	10.1848	0.0336	81.1802	2.7237	75.7320	86.6283	0.0336
	Total	252	273393	0.04551	273393	273393	0.0000	11.7569	11.7569	0.0000	100.0000				
Total	No	142	713189	58731	595709	830669	0.0824	25.6176	35.7217	0.0824					
	Yes	411	1612199	58731	1494719	1729679	0.0364	64.2783	74.3824	0.0364					
	Total	553	2325388	0.04551	2325388	2325388	0.0000								

For R users

First, install the *haven*, *survey,* and *dplyr* packages (Lumley 2017) as needed. The *haven* package is used to import and export data created from other statistical software including SAS; the *survey* package is used to analyze complex survey data; and the *dplyr* package is for data manipulation. Note that the text starting with # are comment lines. Below shows some examples of installing the packages and loading the R libraries:

```
install.packages("survey","dplyr", "haven")
library(haven)
library(survey)
library(dplyr)
```

R Example 1 (National Household Characteristics Estimation)

Objective: Calculate the total of households (with the estimated percentage) and RSE that have 50% or more LED indoor bulbs in the U.S. (Table HC5.2)

LGTINLED represents the portion of indoor light bulbs that are LED, and it has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

Step 1. After reading in the microdata file to R, derive a new variable LEDGE50PCT by combining categories 1, 2, and 3 from LGTINLED to indicate a household with at least 50% of indoor LED bulbs.

```
#Read in the microdata file in csv format
RECS2024_PUBLIC_v1<-read.csv("<location where file is stored>/recs2024_public_v1.csv",
header=TRUE, sep=",")

# Create LEDGE50PCT variable in the dataset
RECS2024_PUBLIC<- RECS2024_PUBLIC_v1 %>% mutate(LEDGE50PCT = ifelse(LGTINLED %in% c(1,
2, 3), "Yes", "No"))
```

Step 2. Define the Jackknife replicate weights used for variance estimation:

```
repweights<-select(RECS2024_PUBLIC,NWEIGHT1:NWEIGHT60)
```

Step 3. Define the survey design with the Jackknife replicate weights to calculate appropriate standard errors using *svrepdesign*. The statement *type="JK1"* tells R to use the delete-1 Jackknife method for variance estimation. The statement *combined.weights=TRUE* means the sum of each replicate weight equals the sum of the full-sample weight. The *scale* argument defines the factor parameter for computing variance. The default is the total number of replicates minus one divided by the total number of replicates. For the 2024 RECS, setting it explicitly as below is the same as the default. The *mse=TRUE* means the Jackknife variance estimator is calculated based on deviations from the full-sample estimate, rather than from the mean of the replicates.

```
RECS24 <- svrepdesign(data = RECS2024_PUBLIC,
                      weight = ~NWEIGHT,
                      repweights = repweights,
                      type = "JK1",
                      combined.weights = TRUE,
                      scale = (ncol(repweights)-1)/ncol(repweights),
                      mse = TRUE)
```

Step 4. Use *svytotal* to sum the total weighted frequency of households for LEDGE50PCT, using the survey design defined above; use *svymean* to obtain estimated percentage of households for the two categories of LEDGE50PCT. The RSEs can be calculated using the SE of the estimates divided by the estimates and multiplied by 100. For example, the RSE for the total weighted frequency is *(LEDGE50PCT_Total$SE/LEDGE50PCT_Total$total)*100*. To calculate the 95% confidence interval, use the t-value with a degree of freedom of 59, which is approximately 2. Note that if the *confint* function is used in R, it defaults to a z-value of 1.96 for a 95% confidence interval.

```
#Calculate the weighted total, RSE, and the 95% confidence interval for the estimated
total of LEDGE50PCT, using a t-value of 2 wth 59 degree of freedom
LEDGE50PCT_Total<-as.data.frame(svytotal(~LEDGE50PCT,RECS24))
LEDGE50PCT_Total$RSE<-(LEDGE50PCT_Total$SE/LEDGE50PCT_Total$total)*100
LEDGE50PCT_Total$CI_Lower<-(LEDGE50PCT_Total$total-2*LEDGE50PCT_Total$SE)
LEDGE50PCT_Total$CI_Upper<-(LEDGE50PCT_Total$total+2*LEDGE50PCT_Total$SE)

#Calculate the estimated proportion, RSE, and the 95% confidence interval for the
estimated portion of LEDGE50PCT, using a t-value of 2 with 59 degree of freedom
LEDGE50PCT_Percent<-as.data.frame(svymean(~LEDGE50PCT,RECS24))
LEDGE50PCT_Percent$RSE<-(LEDGE50PCT_Percent$SE/LEDGE50PCT_Percent$mean)*100
LEDGE50PCT_Percent$CI_Lower<-(LEDGE50PCT_Percent$mean-2*LEDGE50PCT_Percent$SE)
LEDGE50PCT_Percent$CI_Upper<-(LEDGE50PCT_Percent$mean+2*LEDGE50PCT_Percent$SE)

#Print the results
print(LEDGE50PCT_Total)
print(LEDGE50PCT_Percent)
```

Answer. The estimated number of households that have at least 50% of indoor LED bulbs is 99,127,702, as shown in Figure 4. The corresponding RSE for the estimate is 0.43%. In other words, the RSE is less than 1% of the estimated total population, a relatively small amount, indicating that the estimate is very precise. In addition, about 75% of households have at least 50% of indoor LED bulbs.

Figure 4. Output of LEDGE50PCT in R at the U.S. Level

```
> print(LEDGE50PCT_Total)
                  total       SE       RSE CI_Lower CI_Upper
LEDGE50PCTNo   33409133 426120.9 1.2754624 32556891 34261375
LEDGE50PCTYes  99127702 426120.9 0.4298707 98275461 99979944
> print(LEDGE50PCT_Percent)
                    mean          SE       RSE  CI_Lower  CI_Upper
LEDGE50PCTNo   0.2520743 0.003215113 1.2754624 0.2456441 0.2585045
LEDGE50PCTYes  0.7479257 0.003215113 0.4298707 0.7414955 0.7543559
```

R Example 2 (State Level Household Characteristics Estimation)

Objective: Calculate the totals of households (with the estimated percentage) and RSEs that have 50% or more LED indoor bulbs in Alabama and Alaska (Table STelectronicsAndLighting_2024)

LGTINLED represents the portion of indoor light bulbs that are LED, and it has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

Step 1. After reading in the microdata file to R, derive a new variable LEDGE50PCT by combining the 1, 2, and 3 categories from LGTINLED to indicate a household with at least 50% of indoor LED bulbs.

```
#Read in the microdata file in csv format
RECS2024_PUBLIC_v1<-read.csv("<location where file is stored>/recs2024_public_v1.csv",
header=TRUE, sep=",")

# Create LEDGE50PCT variable in the dataset
RECS2024_PUBLIC<- RECS2024_PUBLIC_v1 %>% mutate(LEDGE50PCT = ifelse(LGTINLED %in% c(1,
2, 3), "Yes", "No"))
```

Step 2. Define the Jackknife replicate weights used for variance estimation:

```
repweights<-select(RECS2024_PUBLIC,NWEIGHT1:NWEIGHT60)
```

Step 3. Define the survey design with the Jackknife replicate weights to calculate appropriate standard errors using *svrepdesign*. The *type="JK1"* tells R to use the delete-1 Jackknife method for variance estimation. The *combined.weights=TRUE* means the sum of each replicate weight equals to the sum of the full-sample weight. The *scale* argument defines the factor parameter for computing variance. The default is the total number of replicates minus one divided by the total number of replicates, which does not need to be set explicitly as below. The *mse=TRUE* means the Jackknife variance estimator is calculated based on deviations from the full-sample estimate, rather than from the mean of the replicates.

```
RECS24 <- svrepdesign(data = RECS2024_PUBLIC,
               weight = ~NWEIGHT,
               repweights = repweights,
               type = "JK1",
               combined.weights = TRUE,
```

```
                              scale = (ncol(repweights)-1)/ncol(repweights),
                              mse = TRUE)
```

Step 4. Use *subset* on the survey design to get estimates for Alabama and Alaska only.

```
RECS24_SUBST<-subset(RECS24,state_name %in% c("Alabama", "Alaska"))
```

Step 5. Use *svytotal* to sum the total weighted frequency of households for LEDGE50PCT, using the survey design defined above; and use *svymean* to obtain estimated percentage of households for the two categories of LEDGE50PCT. The RSEs can be calculated using the SE of the estimates divided by the estimates, and multiply by 100. For example, the RSE for the total weighted frequency is *(LEDGE50PCT_Total$SE/LEDGE50PCT_Total$total)*100*. To calculate the 95% confidence interval, use the t-value with a degree of freedom of 59, which is approximately 2. Note that if the *confint* function is used in R, it defaults to a z-value of 1.96 for a 95% confidence interval.

```
#Calculate the weighted totals, RSEs, and the 95% confidence intervals for the
estimated total of LEDGE50PCT for Alabama and Alaska, using a t-value of 2 wth 59
degrees of freedom
LEDGE50PCT_Total_SubST<-as.data.frame(svytotal(~interaction(state_name,
LEDGE50PCT),RECS24_SUBST))
LEDGE50PCT_Total_SubST$RSE<-
(LEDGE50PCT_Total_SubST$SE/LEDGE50PCT_Total_SubST$total)*100
LEDGE50PCT_Total_SubST$CI_Lower<-(LEDGE50PCT_Total_SubST$total-
2*LEDGE50PCT_Total_SubST$SE)
LEDGE50PCT_Total_SubST$CI_Upper<-
(LEDGE50PCT_Total_SubST$total+2*LEDGE50PCT_Total_SubST$SE)

#Calculate the estimated proportion, RSEs, and the 95% confidence intervals for the
estimated portion of LEDGE50PCT for Alabama and Alaska, using a t-value of 2 wth 59
degrees of freedom
LEDGE50PCT_Percent_SubST<-as.data.frame(svyby(~factor(LEDGE50PCT),
~state_name,RECS24_SUBST,svymean))
LEDGE50PCT_Percent_SubST$RSE<-
(LEDGE50PCT_Percent_SubST$se2/LEDGE50PCT_Percent_SubST$'factor(LEDGE50PCT)Yes')*100
LEDGE50PCT_Percent_SubST$CI_Lower<-(LEDGE50PCT_Percent_SubST$'factor(LEDGE50PCT)Yes'-
2*LEDGE50PCT_Percent_SubST$se2)
LEDGE50PCT_Percent_SubST$CI_Upper<-
(LEDGE50PCT_Percent_SubST$'factor(LEDGE50PCT)Yes'+2*LEDGE50PCT_Percent_SubST$se2)

#Print the results
print(LEDGE50PCT_Total_SubST)
print(LEDGE50PCT_Percent_SubST)
```

Answer. The estimated number of households that have at least 50% indoor LED bulbs in Alabama is 1,390,258, and 221,941 in Alaska, as shown in Figure 5. The corresponding estimated percentages are approximately 68% of households in Alabama and 81% of households in Alaska. In addition, the corresponding RSEs for the estimates are 4.15% in Alabama and 3.36% in Alaska.

Figure 5. Output of LEDGE50PCT in R for Alabama and Alaska

```
> print(LEDGE50PCT_Total_SubST)
                                                  total        SE       RSE  CI_Lower   CI_Upper
interaction(state_name, LEDGE50PCT)Alabama.No  661736.79 57663.940  8.714030  546408.9  777064.67
interaction(state_name, LEDGE50PCT)Alaska.No    51452.14  7446.374 14.472427   36559.4   66344.89
interaction(state_name, LEDGE50PCT)Alabama.Yes 1390258.21 57663.940  4.147714 1274930.3 1505586.09
interaction(state_name, LEDGE50PCT)Alaska.Yes   221940.86  7446.384  3.355121  207048.1  236833.62
> print(LEDGE50PCT_Percent_SubST)
        state_name factor(LEDGE50PCT)No factor(LEDGE50PCT)Yes        se1        se2      RSE  CI_Lower  CI_Upper
Alabama    Alabama            0.3224846             0.6775154 0.02810140 0.02810140 4.147714 0.6213126 0.7337182
Alaska      Alaska            0.1881985             0.8118015 0.02723689 0.02723689 3.355117 0.7573277 0.8662753
```

For Python users

Since *svy*, the new Python package for complex survey analysis, is currently in a beta development phase, the following examples demonstrate the manual calculations.

First, install the *pandas and numpy* packages (Lumley 2017) as needed. The *pandas* package is a library used for data manipulation and analysis; and the *numpy* library is fundamental for numerical computing. After the packages are installed, import the pandas and numpy libraries and name them as pd and np. Note that the text starting with # are comment lines.

```
import pandas as pd
import numpy as np
```

Python Example 1 (National Household Characteristics Estimation)

Objective: Calculate the total of households (with the estimated percentage) and RSE that have 50% or more LED indoor bulbs in the U.S. (Table HC5.2)

The LGTINLED variable in the data file represents the portion of indoor light bulbs that are LED within a household, which has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

Step 1. After reading in the microdata file, the following data processing steps can be completed in any order:

- Extract the replicate weights from the dataset, store their column names, and count the number of replicate weights.
- Calculate the total number of households by summing the full-sample analysis weight (denoted as *total_households* below), and create a new variable *mask* based on LGTINLED, where categories 1, 2, and 3 are grouped together to indicate a household with at least 50% indoor LED bulbs.

```
#Read in the microdata file in csv format
RECS2024_PUBLIC = pd.read_csv("<location where file is stored>/recs2024_public_v1.csv")

#Extract the replicate weights from the dataset, store their column names, and count
the number of replicate weights
repweights = RECS2024_PUBLIC.loc[:, "NWEIGHT1":"NWEIGHT60"]
cols = repweights.columns
r = repweights.shape[1]
```

```
#Calculate the weighted total number of households and derive a new variable mask from LGTINLED, where categories 1, 2, and 3 are combined
total_households = RECS2024_PUBLIC.NWEIGHT.sum()
mask = RECS2024_PUBLIC['LGTINLED'].isin([1,2,3])
```

Step 2. Calculate the weighted count estimates, standard errors, relative standard errors, and the corresponding 95% confidence interval for households that have at least 50% or more indoor LED bulbs, then store the outputs in a data frame.

```
#Compute the weighted total of LEDGE50PCT by summing household weights
LEDGE50PCT_total_sampleEstimate = RECS2024_PUBLIC.NWEIGHT.loc[mask].sum()

#Compute replicate differences, variance, standard error (se), and relative standard error (RSE) for the weighted total estimate
LEDGE50PCT_total_diffs = [repweights.loc[mask, cols[i]].sum() for i in range(r)] - 
LEDGE50PCT_total_sampleEstimate
LEDGE50PCT_total_squares = [x**2 for x in LEDGE50PCT_total_diffs]
LEDGE50PCT_total_sumSquares = sum([x * (r-1)/r for x in LEDGE50PCT_total_squares])
LEDGE50PCT_total_se = np.sqrt(LEDGE50PCT_total_sumSquares)
LEDGE50PCT_total_rse = LEDGE50PCT_total_se / LEDGE50PCT_total_sampleEstimate * 100

#Compute the 95% confidence interval of the weighted total estimate, using a t-value of 2 with 59 degrees of freedom
LEDGE50PCT_total_CI_lower = LEDGE50PCT_total_sampleEstimate - 2 * LEDGE50PCT_total_se
LEDGE50PCT_total_CI_upper = LEDGE50PCT_total_sampleEstimate + 2 * LEDGE50PCT_total_se

#Store the weighted total estimate, se, rse, and confidence interval values in a data frame
LEDGE50PCT_total = pd.DataFrame(data = {
    "estimate": LEDGE50PCT_total_sampleEstimate,
    "SE": LEDGE50PCT_total_se,
    "RSE": LEDGE50PCT_total_rse,
    "CI_lower": LEDGE50PCT_total_CI_lower,
    "CI_upper": LEDGE50PCT_total_CI_upper
    },
    index = ["LEDGE50PCT_total"])
```

Step 3. Calculate the estimated percentage, standard errors, relative standard errors, and the corresponding 95% confidence interval for households that have at least 50% or more indoor LED bulbs.

```
#Compute the weighted percentage estimate of LEDGE50PCT
LEDGE50PCT_percent_sampleEstimate = LEDGE50PCT_total_sampleEstimate / 
total_households * 100

#Compute replicate differences, variance, standard error (se), and relative standard error (RSE) for the weighted percentage estimate
```

```
LEDGE50PCT_percent_diffs = [repweights.loc[mask, cols[i]].sum() /
total_households * 100 for i in range(r)] - LEDGE50PCT_percent_sampleEstimate
LEDGE50PCT_percent_squares = [x**2 for x in LEDGE50PCT_percent_diffs]
LEDGE50PCT_percent_sumSquares = sum([x * (r-1)/r for x in
LEDGE50PCT_percent_squares])
LEDGE50PCT_percent_se = np.sqrt(LEDGE50PCT_percent_sumSquares)
LEDGE50PCT_percent_rse = LEDGE50PCT_percent_se /
LEDGE50PCT_percent_sampleEstimate * 100

#Compute the 95% confidence interval of the weighted percentage estimate, using
a t-value of 2 with 59 degrees of freedom
LEDGE50PCT_percent_CI_lower = LEDGE50PCT_percent_sampleEstimate - 2 *
LEDGE50PCT_percent_se
LEDGE50PCT_percent_CI_upper = LEDGE50PCT_percent_sampleEstimate + 2 *
LEDGE50PCT_percent_se

#Store the weighted percentage estimate, se, rse, and confidence interval values
in a data frame
LEDGE50PCT_percent = pd.DataFrame(data = {
    "estimate": LEDGE50PCT_percent_sampleEstimate,
    "SE": LEDGE50PCT_percent_se,
    "RSE": LEDGE50PCT_percent_rse,
    "CI_lower": LEDGE50PCT_percent_CI_lower,
    "CI_upper": LEDGE50PCT_percent_CI_upper
    },
    index = ["LEDGE50PCT_percent"])

#Set a global rule to display numbers with two decimal places, print the results
pd.options.display.float_format = '{:.2f}'.format
print(LEDGE50PCT_total)
print(LEDGE50PCT_percent)
```

Answer. The estimated number of households that have at least 50% of indoor LED bulbs is 99,127,702, as shown in Figure 6. The corresponding RSE for the estimate is 0.43%. In other words, the RSE is less than 1% of the estimated total population, a relatively small amount, indicating that the estimate is very precise. In addition, about 75% of households have at least 50% of indoor LED bulbs.

Figure 6. Output of LEDGE50PCT in Python at the U.S. Level

```
                      estimate        SE  RSE    CI_lower    CI_upper
LEDGE50PCT_total   99127702.35 426120.92 0.43 98275460.51 99979944.18
                      estimate   SE  RSE  CI_lower  CI_upper
LEDGE50PCT_percent       74.79 0.32 0.43     74.15     75.44
```

Python Example 2 (State Level Household Characteristics Estimation)

Objective: Calculate the totals of households (with the estimated percentage) and RSEs that have 50% or more LED indoor bulbs in Alabama and Alaska (Table STelectronicsAndLighting_2024)

LGTINLED represents the portion of indoor light bulbs that are LED, and it has five coded values: 1=All; 2=Most; 3=About half; 4=Some; 0=None.

Step 1. After reading in the microdata file, the following data processing steps can be completed in any order:

- Extract the replicate weights from the dataset, store their column names, and count the number of replicate weights.
- First filter the desired states, then calculate the total number of households by summing the full-sample analysis weight (denoted as *total_households* below), and create a new variable *mask* based on LGTINLED, where categories 1,2, and 3 are grouped together to indicate a household with at least 50% indoor LED bulbs.

```
#Read in the microdata file in SAS format
RECS2024_PUBLIC = pd.read_sas("<location where file is
stored>/recs2024_public_v1.sas7bdat", encoding = 'iso-8859-1')

#Extract the replicate weights from the dataset, store their column names, and count
the number of replicate weights
repweights = RECS2024_PUBLIC.loc[:, "NWEIGHT1":"NWEIGHT60"]
cols = repweights.columns
r = repweights.shape[1]

#Loop through the selected geographic areas ALL, Alabama, and Alaska. Calculate the
weighted total number of households and derive a new variable mask from LGTINLED, where
categories 1, 2, and 3 are combined
for state in ["ALL", "Alabama", "Alaska"]:
    if state == "ALL":
        mask = RECS2024_PUBLIC['LGTINLED'].isin([1,2,3])
        total_households = RECS2024_PUBLIC.NWEIGHT.sum()
    else:
        mask = (RECS2024_PUBLIC['LGTINLED'].isin([1,2,3])) &
(RECS2024_PUBLIC['state_name'] == state)
        total_households = RECS2024_PUBLIC.NWEIGHT.loc[(RECS2024_PUBLIC['state_name']
== state)].sum()
```

Step 2. Calculate the weighted count estimates, standard errors, relative standard errors, and the corresponding 95% confidence interval for households that have at least 50% or more indoor LED bulbs, then store the outputs in a data frame.

```
#Compute the weighted total of LEDGE50PCT by summing household weights
 LEDGE50PCT_total_sampleEstimate = RECS2024_PUBLIC.NWEIGHT.loc[mask].sum()
```

```
#Compute replicate differences, variance, standard error (se), and relative standard
error (RSE) for the weighted total estimate
    LEDGE50PCT_total_diffs = [repweights.loc[mask, cols[i]].sum() for i in range(r)] -
LEDGE50PCT_total_sampleEstimate
    LEDGE50PCT_total_squares = [x**2 for x in LEDGE50PCT_total_diffs]
    LEDGE50PCT_total_sumSquares = sum([x * (r-1)/r for x in LEDGE50PCT_total_squares])
    LEDGE50PCT_total_se = np.sqrt(LEDGE50PCT_total_sumSquares)
    LEDGE50PCT_total_rse = LEDGE50PCT_total_se / LEDGE50PCT_total_sampleEstimate * 100

#Compute the 95% confidence interval of the weighted total estimates, using a t-value
of 2 with 59 degrees of freedom
  LEDGE50PCT_total_CI_lower = LEDGE50PCT_total_sampleEstimate - 2 * LEDGE50PCT_total_se
  LEDGE50PCT_total_CI_upper = LEDGE50PCT_total_sampleEstimate + 2 * LEDGE50PCT_total_se

#Store the weighted total estimates, se, rse, and confidence interval values in a data
frame
    LEDGE50PCT_total = pd.DataFrame(data = {
        "estimate": LEDGE50PCT_total_sampleEstimate,
        "SE": LEDGE50PCT_total_se,
        "RSE": LEDGE50PCT_total_rse,
        "CI_lower": LEDGE50PCT_total_CI_lower,
        "CI_upper": LEDGE50PCT_total_CI_upper
        },
        index = [" ".join([state, "LEDGE50PCT_total"])]
    )
```

Step 3. Calculate the estimated percentage, standard errors, relative standard errors, and the corresponding 95% confidence interval for households that have at least 50% or more indoor LED bulbs.

```
#Compute the weighted percentage estimate of LEDGE50PCT
    LEDGE50PCT_percent_sampleEstimate = LEDGE50PCT_total_sampleEstimate / total_households

   #Compute replicate differences, variance, standard error (se), and relative standard
   error (RSE) for the weighted percentage estimate
       LEDGE50PCT_percent_diffs = [repweights.loc[mask, cols[i]].sum() / total_households
   *100 for i in range(r)] - LEDGE50PCT_percent_sampleEstimate
       LEDGE50PCT_percent_squares = [x**2 for x in LEDGE50PCT_percent_diffs]
       LEDGE50PCT_percent_sumSquares = sum([x * (r-1)/r for x in
   LEDGE50PCT_percent_squares])
       LEDGE50PCT_percent_se = np.sqrt(LEDGE50PCT_percent_sumSquares)
       LEDGE50PCT_percent_rse = LEDGE50PCT_percent_se / LEDGE50PCT_percent_sampleEstimate
   * 100

   #Compute the 95% confidence interval of the weighted percentage estimates, using a t-
   value of 2 with 59 degrees of freedom
    LEDGE50PCT_percent_CI_lower = LEDGE50PCT_percent_sampleEstimate - 2 *
LEDGE50PCT_percent_se
```

```
    LEDGE50PCT_percent_CI_upper = LEDGE50PCT_percent_sampleEstimate + 2 *
LEDGE50PCT_percent_se

    #Store the weighted percentage estimate, se, rse, and confidence interval values in a
   data frame
    LEDGE50PCT_percent = pd.DataFrame(data = {
        "estimate": LEDGE50PCT_percent_sampleEstimate,
        "SE": LEDGE50PCT_percent_se,
        "RSE": LEDGE50PCT_percent_rse,
        "CI_lower": LEDGE50PCT_percent_CI_lower,
        "CI_upper": LEDGE50PCT_percent_CI_upper
        },
        index = [" ".join([state, "LEDGE50PCT_percent"])]
    )

    #Set a global rule to display numbers with two decimal places, print the results
    pd.options.display.float_format = '{:.2f}'.format
    print(LEDGE50PCT_total)
    print(LEDGE50PCT_percent)
```

Answer. The estimated number of households that have at least 50% indoor LED bulbs in Alabama is 1,390,258, and 221,941 in Alaska, as shown in Figure 7. The corresponding estimated percentages are approximately 68% of households in Alabama and 81% of households in Alaska. In addition, the corresponding RSEs for the estimates are 4.15% in Alabama and 3.36% in Alaska.

Figure 7. Output of LEDGE50PCT in Python for Alabama and Alaska

```
                        estimate        SE  RSE    CI_lower    CI_upper
ALL LEDGE50PCT_total 99127702.35 426120.92 0.43 98275460.51 99979944.18
                        estimate   SE  RSE  CI_lower  CI_upper
ALL LEDGE50PCT_percent     74.79 0.32 0.43     74.15     75.44
                            estimate       SE  RSE   CI_lower   CI_upper
Alabama LEDGE50PCT_total  1390258.21 57663.94 4.15 1274930.33 1505586.09
                             estimate   SE  RSE  CI_lower  CI_upper
Alabama LEDGE50PCT_percent      67.75 2.81 4.15     62.13     73.37
                          estimate      SE  RSE   CI_lower   CI_upper
Alaska LEDGE50PCT_total  221940.86 7446.38 3.36  207048.09  236833.62
                            estimate   SE  RSE  CI_lower  CI_upper
Alaska LEDGE50PCT_percent      81.18 2.72 3.36     75.73     86.63
```

Notes to Consider When Using the Microdata File and Replicate Weights

1. *Publication standards:* We do not publish RECS estimates where the RSE is higher than 50 or the number of households used for the calculation is less than 10 (indicated by a *Q* in the data tables). We recommend following these standards for custom analysis using the public use

microdata file.

2. *Imputation variables:* We imputed most variables for *Don't Know* and *Refuse* responses. The *Z variables*, also referred to as *imputation flags*, are in the public use microdata file. The imputation flag indicates whether we based the corresponding non-Z variable on reported data (Z variable = 0) or if we imputed it (Z variable = 1). Variables from the RECS questionnaire that we did not impute, contained no missing data, or were derived from the questionnaire (and not used as core variables in imputation) have no corresponding *Z* variables. We recommend using the imputed data, where available, to avoid biased estimation.

3. *Standardized coding:* Variables for questions that we did not ask all respondents use the response code -2 for *Not Applicable*. For example, respondents who answered that they did not use any televisions at home (TVCOLOR = 0) were not asked what size television they most use at home, so TVSIZE1 = -2. Use caution when performing calculations on variables that have -2 responses.

4. *Modified variables and revised variable names:* A small number of survey variables in the microdata file have been modified for disclosure protection or analytical purposes. These modifications primarily involved top-coding. The list below provides the original and revised variable names:

    - BEDROOMS (number of bedrooms) was renamed to BEDROOMS_PUB.
    - OTHROOMS (number of other rooms) was renamed to OTHROOMS_PUB.
    - NCOMBATH (number of full bathrooms) was renamed to NCOMBATH_PUB.
    - NHAFBATH (number of half bathrooms) was renamed to NHAFBATH_PUB.
    - HHAGE (age of the householder) was renamed to HHAGE_PUB.
    - NUMCHILD (number of children under 18) was renamed to NUMCHILD_PUB.
    - NUMADULT1 (number of household members age 18 to 64) was renamed to NUMADULT1_PUB.
    - NUMADULT2 (number of household members age 65 or older) was renamed to NUMADULT2_PUB.
    - RACE_WHITE (Respondent is White), RACE_BLACK (Respondent is Black or African-American), RACE_AIAN (Respondent is American Indian or Alaska Native), RACE_ASIAN (Respondent is Asian), and RACE_NHPI (Respondent is Native Hawaiian or Other Pacific Islander) were reprocessed into one variable - HOUSEHOLDER_RACE.

5. *Confidentiality:* We collected the 2024 RECS under the authority of the Confidential Information Protection and Statistical Efficiency Act (CIPSEA). The agency, project staff, and our contractors

and agents are personally accountable for protecting the identity of individual respondents. We took the following steps to avoid disclosing personally identifiable information in the public-use microdata file.

- We removed local geographic identifiers of sampled housing units, such as addresses.
- The following variables were top-coded and are identifiable in the microdata file by the suffix '_PUB' appended to their new variable names
  - BEDROOMS_PUB (number of bedrooms) to 6
  - OTHROOMS_PUB (number of other rooms) to 9
  - NCOMBATH_PUB (number of full bathrooms) to 4
  - NHAFBATH_PUB (number of half bathrooms) to 2
  - HHAGE_PUB (age of the householder) to 90
  - NUMCHILD_PUB (number of children under 18) to 4
  - NUMADULT1_PUB (number of household members age 18 to 64) to 4
  - NUMADULT2_PUB (number of household members age 65 or older) to 2
- Note that TOTROOMS_PUB (total number of rooms) was derived from the sum of BEDROOMS_PUB and OTHROOMS_PUB; and NHSLDMEM_PUB (total number of household members) was derived from the sum of NUMCHILD_PUB, NUMADULT1_PUB, and NUMADULT2_PUB.
- We rounded SQFTEST (respondent-reported square footage) to the nearest 10 and adjusted SQFTRANGE (Respondent-reported square footage range) where necessary. The modified variables are now denoted as SQFTEST_PUB and SQFTRANGE_PUB in the microdata file.

References

Lohr, S.L. (2010). Sampling: Design and Analysis. 2nd ed. Boston: Brooks/Cole. Page 380–383.

Lumley, T. (2017) "Survey: analysis of complex survey samples". R package version 4.1-1.

The SAS code and output presented in this document were developed and tested using SAS/STAT software, Version 9.4_M8 of the SAS Enterprise Guide 8.4 for UNIX.

The R code presented in this document was developed and tested in version 4.4.1.

The Python code presented in this document was developed and tested in version 3.12.4.