Analysis Plan

Project title: Characterizing Nonresponse Bias in the American Housing Survey (AHS)
Project code: 1901

1 Project Objectives

The American Housing Survey (AHS) is a biannual, longitudinal survey of housing units designed by the U.S. Department of Housing and Urban Development and administered by the U.S. Census Bureau. The sample of housing units is drawn from residential units in the United States and is designed to provide nationally representative statistics and statistics representative of the nation's largest metropolitan areas. The AHS provides important information on key features of the U.S. housing stock: how many people rent versus own their homes? How many are evicted? What proportion of units have adequate conditions, and what are the demographics of those who live in inadequate units?

Yet as with many sources of official statistics, the AHS has experienced declining response rates, requiring increasing amounts of time and effort to maintain an acceptable response rate. In the 2015 survey, 69,511 out of 82,011 units responded, or a rate of 85 percent.[1] In the 2017 survey, 66,752 out of 82,933 units responded, or a rate of 80 percent.[2] The 2019 response rates could be even lower.

The declining AHS response *rates* also could contribute to nonresponse *bias*, or divergence between the attributes of the sample and the attributes of the target population. To account for this bias, the AHS calculates a noninterview adjustment factor (NAF) that reweights for nonresponse within cells defined by metropolitan area, type of housing unit, block group median income, and area-level rural/urban status.

The goal of the present plan is to pre-specify an analysis to more fully characterize the degree of nonresponse bias in the AHS by:

1. Defining different mechanisms for nonresponse;
2. Defining and illustrating the use of different measures of unit-level nonresponse bias; and
3. Defining and illustrating the use of different measures of item-level nonresponse bias.

For the actual analysis, we will (1) focus on the 2015, 2017, and 2019 waves of the AHS, and (2) use the Internal Use Files (IUF) that contain information on both respondents and nonresponders. We use the 2015 and 2017 integrated national AHS and the Public Use File (PUF) for illustrating each measure in this pre-analysis plan.

The analysis of various forms of nonresponse bias in recent AHS waves will help us characterize the form that bias takes.[3] This understanding will support plans to field an incentives experiment that will (1) model different units' propensity of contributing to nonresponse bias, and (2) compare the effect of randomized incentives on reducing bias to the effect of propensity-targeted incentives.[4]

[1] Source: Section 2.2 of 2015 AHS documentation.

[2] Source: Section 2.2 of 2017 AHS documentation.

[3] For instance, how much of the missingness among interviewees and items is explainable via observed characteristics?

[4] Office of Evaluation Sciences. 2020. *Project design: using incentives to reduce nonresponse bias in the American Housing Survey*.

2 Types of and mechanisms for nonresponse

The different types and mechanisms of nonresponse have meaningul implications for potential nonresponse bias. We describe two types and three mechanisms of nonresponse.

2.1 Types of nonresponse: unit-level versus item-level

There are two types of nonresponse by units in the AHS.

The first type of nonresponse is unit-level nonresponse: there is an attempt to include the unit in the survey, but either the interviewer was unable to find the unit (occupied or vacant units), the interviewer found the unit but no one was at home after repeated visits (occupied units), or the interviewer found the unit and made contact with a resident, but the resident refused to be interviewed (occupied units).[5]

The second type is item-level nonresponse: someone was interviewed, but they refused to respond to a specific question or questions.[6] Item-level nonresponse matters because different users draw on the AHS to investigate different questions. For instance, one user may be interested in the demographics of residents who had their utilities shut off; another may be interested in the demographics of residents who live in mobile homes. A different degree of nonresponse bias in the items used to answer these questions means that the answers the researchers find will generalize more or less well.

2.2 Mechanisms for nonresponse

There are three mechanisms that generate a nonresponse, either at the unit or at the item level.

First, a respondent's entire interview or value for an item may be Missing Completely at Random (MCAR): the missingness is neither related to observed attributes of the respondent nor to the value(s) of the item(s) themselves. If this is the mechanism generating nonresponse, nonresponse may decrease the *precision* of estimates—the smaller sample size contributes to more variance in the estimates—but it does not contribute to nonresponse bias because, in expectation, the sample still resembles the broader population from which it is drawn.

Second, a respondent's values may be Missing at Random (MAR). MAR is when the observed characteristics of a unit are correlated with its likelihood of response. For instance, if units located in more difficult-to-access rural areas have a higher likelihood of nonresponse, we know that we need to correct for this bias and which attributes to use in this correction. In particular, in theory, we can adjust for MAR-generated nonresponse by reweighting using attributes that are predictive of (1) nonresponse and (2) the values of the items themselves. In practice, even if MAR is the *mechanism* behind missingness, there are practical drawbacks behind reweighting. For unit-level nonresponders, we only have a few sampling frame variables to predict nonresponse. If the attributes we have available are poor predictors, reweighting increases the sampling variance of resulting parameters.

Finally, a respondent's values may be Missing Not at Random (MNAR), or where missingness is a function not only of observed attributes of units but also unobserved attributes like the value of the missing item itself. For instance, missing not at random would occur if those who feel unsafe in their neighborhood are the least likely to answer the item about neighborhood safety. In the case of missing not at random, methods like reweighting can increase rather than decrease bias, since we are adjusting estimates based on *observed* attributes of respondents and nonresponders, but unobserved attributes are generating the missing interviews or values for items.

The methods we review in this plan are focused on nonresponse generated by the mechanism of Missing at Random,

[5] The public use file (PUF) that we are using for the present analysis plan does not have information on nonresponders, so we cannot describe the breakdown of these three sources of nonresponse. The internal use file (IUF) may have information on the relative magnitude of each source of nonresponse.

[6] We can think of unit-level nonresponse as an extreme form of item-level nonresponse: the respondent has refused to answer each of the items, rather than refused to answer a subset of items from the entire interview. HUD employs internal definitions for what items a respondent needs to answer to be counted as a complete response at the survey level. We will use those definitions for the final analysis, but do not have access to them for the present plan.

meaning that we expect some observable characteristics—i.e., the sampling frame variables for unit-level nonresponse; other items for item-level nonresponse—to predict nonresponse. Methods for addressing missing not at random, or non-ignorable missingness, require either (1) randomizing potential respondents to treatments that affect response propensities independent of unobserved characteristics, or (2) augmenting the survey to include items that directly measure general response propensities (Bailey 2019). We plan on using the first method in our incentives experiment. The goal of this plan is thus to diagnose nonresponse bias before intervening to try to decrease the bias.

3 Plan to assess unit-level nonresponse bias

Unit-level nonresponse is an extreme version of item-level nonresponse: we are missing values for all items within the interview. As a result, we have limited information about the units that *do not* respond to investigate how those units differ from those who do respond.

In the case of the AHS, we have three sources of information on units that do not respond to a particular wave:

1. **Sampling frame variables (all units):** the sampling frame is based on addresses. Table 1 lists the attributes we observe in both units that respond and nonresponding units.[7] We use S to refer to these attributes available for both responding units and units that are not interviewed.

2. **Variables from other waves of the AHS (some units):** some units that are nonresponders for a particular wave are responders for other waves. If the residents of a unit stay the same between waves, demographic attributes of those residents may predict whether they want to be re-interviewed. If the unit's residents change,[8] correlation in attributes between different households that reside in the same unit over time may still mean attributes from other waves are predictive of unit-level nonresponse for a focal wave.

3. **Geographic aggregate data from external surveys (all units):** while we have limited individual-level characteristics on units, especially units that are "never responders," the Internal Use File (IUF) does have the target unit's block group. We can link the block group to aggregate geographic information from the American Community Survey and/or Decenniel Census, such as racial demographics and educational attainment. While this is aggregated geographically, so potentially presents ecological fallacy issues,[9] it provides an additional source of information on unit-level nonresponders.

Table 1: Variables in the sampling frame

Variable	Survey
Census Division	PUF
Building type	PUF
CBSA	PUF
Quartiles of block group median income	IUF
Area rural/urban status	IUF

3.1 Descriptive analysis of unit-level nonresponse

The measures we outline below examine *bias* caused by nonresponse. Before calculating these measures of potential bias, we will conduct a descriptive analysis that:

[7] Source (Section 3.1) of AHS 2015 documentation. In this pre-analysis plan, we only have access to the three attributes available in the Public Use File (PUF).

[8] While this information is only available in the PUF for respondents to both waves, we will investigate whether data from sources like a national address database can be used to obtain this information for both respondents and nonresponders.

[9] For instance, if we observe a negative relationship between the percent college educated within a block group and whether a unit responds, but the true pattern is that households with less education living in high education areas respond at higher rates, the aggregate patterns provide misleading information on the relationship between individual attributes and response likelihood.

- Adds more attributes to predict nonresponse than the limited set of sampling frame variables. In particular, we will add the following characteristics:
    1. Uses the full set of sampling frame attributes;
    2. Merges in block group-level demographic characteristics that are not currently part of the sampling frame to add additional information; and
    3. For nonresponders for a focal wave that have responses from other waves, uses demographic, income, housing costs, and housing problems attributes from other waves to predict nonresponse for the focal wave. These characteristics may correspond to a household that moved into the unit before or after the focal wave. If households who reside in a unit at different times are similar, these attributes will still improve our prediction of nonresponse. If the households do not have similar characteristics, then these attributes will not improve our prediction of nonresponse and the best-performing model should downweight.
- Then, in addition to the logistic regression models for nonresponse we outline below, we will (1) test a range of binary classifiers, (2) use 5-fold cross-validation and the F1 score to choose a top-performing classifier, and (3) examine attributes that are most highly predictive of nonresponse in these more flexible models

The code in the Appendix outlines how we will implement the binary classifiers in Python.

3.2 First measure of unit-level nonresponse bias: nonresponse rates

The first measure we will use, because it is reported in other studies of nonresponse bias, is the response rate:

$$\frac{\text{Units interviewed (with any degree of missingness})}{\text{Units eligible}}$$

We will report the following response rates for each of the three waves:[10]

1. Full sample response rate
2. Response rate separated by each of the sampling frame variables in Table 1
    - For these, we will do a chi-square test for independence, separate for each sampling variable, where the categories are the levels of those sampling frame variables and there is a count of ”yes” or ”no” respondents.

3.3 Second measure of unit-level nonresponse bias: representivity index (R-indicator)

While response rate is commonly reported as a measure of nonresponse bias, the two are distinct. Lower rates do not necessarily indicate that those who respond differ in either observed or unobserved ways from those who do not (Groves and Peytcheva 2008).

Therefore, for our second measure of unit-level nonresponse bias, we use the R-indicator proposed by Schouten et al. (2009). The R-indicator measures, at the unit level, how “representative” the sampled units are of the target population. It investigates whether, based on observed characteristics, the units that respond have similar response propensities as the units that do not respond. If there is less dispersion in these propensities, we infer less bias.

To illustrate the value of the R-indicator as a complement to raw response rates, we simulate two mechanisms for unit-level nonresponse: one where the noninterviewed units have similar observed attributes to the interviewed units (ran-

[10] For the present writeup, since we are using PUF data that only contains respondents and simulating unit-level nonresponse using the published N of nonresponders, we focus only on the full sample response rate.

dom data generating process, or DGP); another where noninterviewed units differ in observed ways (biased DGP).[11] We use the simulated data to (1) develop the code for calculating the R-indicator, and (2) characterize its performance under a response mechanism—missing at random—that is plausible for the AHS.

First simulated data: interview status is uncorrelated with measured attributes
How we operationalized:

- Unit nonresponders are missing on all variables except the ones that are available as part of the sampling frame (Table 1); and
- For those variables, the nonresponders' values are a random draw from the vector of values for the interviewed units (which means that, for instance, the probability of value S_a for attribute S (e.g., mobile home for building type) is equal for the two samples).

Second simulated data: interview status is correlated with measured attributes
How we operationalized:

- Similar to the first set of simulated data, noninterviews are missing on all variables except those available in the sampling frame;
- For those variables, their values oversample certain values from the interviewed units:
  - Mobile homes and large apartment buildings are more highly represented among nonresponders; and
  - Units from two census regions (South Atlantic and Mountain) are more highly represented among nonresponders.

R-indicator: definition
The R-indicator is calculated as follows:

1. Estimate a binary regression predicting "interviewed" or not, based on attributes observed for both respondents and nonresponders (S)
2. Calculate $\hat{y}$ using the regression parameters from Step 1
3. Find $SD(\hat{y})$: *less* variability in response propensities (smaller SD) is potentially indicative of less bias via observed attributes; *more variability* in response propensities (larger SD) is indicative of more bias.
4. To get a value between 0 and 1, re-parametrize so that:

$$R = 1 - 2 \times SD(\hat{y})$$

More variability in fitted response propensity $\implies$ higher SD $\implies$ lower R-indicator

Less variability in fitted response propensity $\implies$ lower SD $\implies$ higher R-indicator

So a higher R-indicator is meant to indicate a more "representative" sample along observed attributes (measured in both groups), which could be correlated with other attributes that we either only measure for respondents or that are unobserved in both groups.

R-indicator: calculating
We use the sampling frame attributes to predict response propensity, and a logistic regression to estimate this propensity.

Figure 1 shows the results. When we simulate a random process behind nonresponse, the response propensities are clustered tightly together. When we simulate a biased process, the response propensities are more spread out, which indicates that units with different characteristics have different response propensities. Put differently, this indicates

[11] In the real analysis, we will have access to data on both respondents and nonresponders and can look empirically at how the two groups differ.

systematic bias in the nonresponse. Table 2 summarizes the R-indicator values. It shows how, in the presence of nonresponse bias, the same response rates can conceal different levels of nonresponse bias. In particular, our simulated data have the same raw response rates. But the R-indicator is much lower when we simulate data where the nonresponding units differ in observable ways from the responding ones. Therefore, we will use both measures in our assessment of unit-level nonresponse bias.

```
calc_r_indicator <- function(data, vars_shared, interview_status_var,
                             fit_logit = TRUE, response_propensities = ""){

  if(fit_logit == TRUE){

      ## fit model
      fit_model = glm(formula(sprintf("%s ~ %s", interview_status_var,
                            paste(vars_shared,
                                  collapse = "+"))),
            data = data,
            family = "binomial")

      ## create data
      predict_results = data.frame(fitted = fit_model$fitted.values) %>%
                  mutate(r_indicator = 1-2*sd(fitted))


  } else {

    predict_results = data.frame(fitted = response_propensities) %>%
                  mutate(r_indicator = 1-2*sd(fitted))

  }


  ## return
  return(list(predict_results))

}

## first calc r indicator where
## int status is mcar
rindic_rand = calc_r_indicator(data = df_random_nonint,
                             vars_shared = vars_avail_nonint,
                             interview_status_var = "interviewed")

rindic_corr = calc_r_indicator(data = df_corr_nonint,
                             vars_shared = vars_avail_nonint,
                             interview_status_var = "interviewed")


## combine and plot dist of fitted values
```

```
rindic_both = rbind.data.frame(rindic_rand[[1]] %>% mutate(dgp = "random"),
                               rindic_corr[[1]] %>% mutate(dgp = "biased"))
```

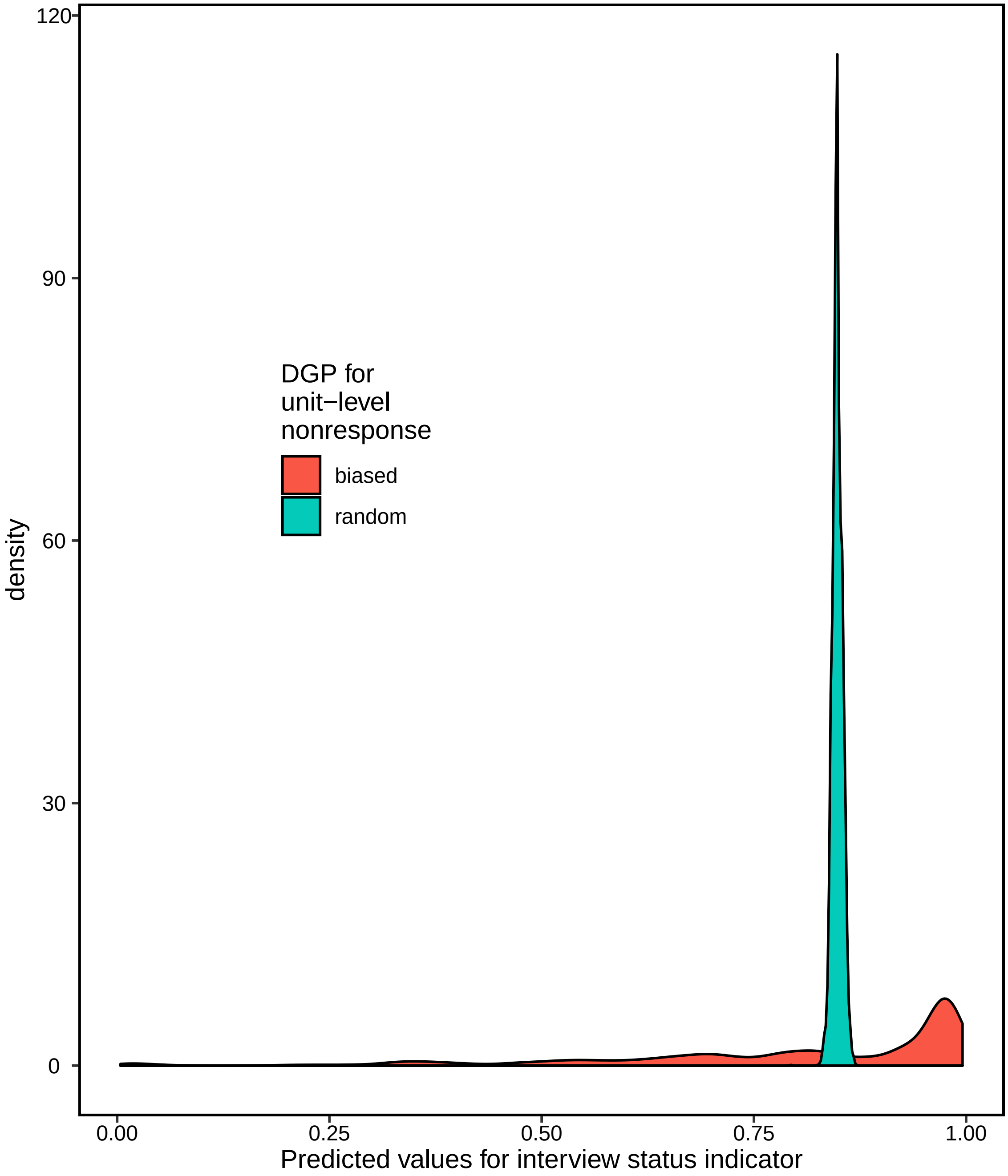

Figure 1: R-indicator values for different forms of missingness

Table 2: Comparing response rate to R-indicator

DGP for nonresponse	Response rate	R-indicator (higher = better)
Random	0.8484319	0.9880549
Biased	0.8484319	0.5745378

R-indicator: inference

Schouten et al. (2009) derive the standard error of $\hat{R}$ through resample bootstrapping. In order to obtain confidence intervals, they assume that $\hat{R}$ is normally distributed. However, our analyses suggest these standard errors are not amenable to the typical Z-score transformation used to obtain p-values in T-tests. We will therefore use a permutation test in order to make an inference about whether we would expect to see the observed $\hat{R}$ simply by chance, or whether the observed $\hat{R}$ is "statistically significant." Specifically, we will randomly shuffle the attrition variable and re-estimate the $\hat{R}$ in order to obtain the sampling distribution under the null of no bias, and compare this to the observed $\hat{R}$.

Namely, we will:

1. Randomly permute the value of the interviewed indicator
2. Re-estimate the R-indicator
3. Repeat the process $m = 1000$ times

This procedure computes the test statistic under the null hypothesis that attrition and covariates are independent (or, that the true R-indicator in an infinite sample is 1). In order to make a decision about the statistical significance of the R indicator we observe, we wish to compare the observed value to the distribution of values we might have estimated, were the null of independence true. Specifically, we conduct a one-sided test: we wish to know the probability of observing, simply by chance, an R-indicator at least as low as the one we observed given that the null of independence is true. We calculate this probability by taking the proportion of permuted R indicators at least as low as the observed one.

Implementing this hypothesis-testing procedure raises three important questions:

1. What is the risk that we will wrongly infer there is no bias when in fact there is (power of $\hat{R}$)?
2. What is the risk that we will wrongly infer there is bias when in fact there is none (false positive rate of $\hat{R}$)?
3. How do these rates compare to more traditional, parametric estimators of goodness-of-fit?

In the simulation study below, we compare the error rates of the hypothesis for $\hat{R}$ to a traditional likelihood-ratio test of the underlying logit model used to calculate $\hat{R}$.

Given that the properties of the R-indicator as a statistical estimator are less well understood than the likelihood ratio test (LRT) that compares two nested models, we can use the LRT as an additional test of the significance of the underlying model used to generate the R indicator.

Specifically, we estimate:

1. *Intercept-only model:* this is a logistic regression where we do not use any attributes to predict response, or:

$$\text{Respond}_i = \alpha + \epsilon_i$$

2. *Model that uses sampling frame attributes:* this is the same logistic regression we use for the R-indicator that

predicts response propensity based on the sampling frame attributes (X_i):

$$\text{Respond}_i = \alpha + \gamma X_i + \epsilon_i$$

The LRT in our case will test whether adding the sampling frame attributes improves the fit of our model estimating nonresponse beyond what one would expect just by random chance.

Below, we simulate these two inferential tests in order to understand their error rates. In the "power analysis," we seek to estimate the probability of rejecting the null of no bias, given that it is false (using random samples of N = 2000 from the "biased" data from above). In the "false positive rate analysis," we seek to estimate the probability of rejecting the null of no bias, given that it is true (using random samples of N = 2000 from the "unbiased" data from above). Ideally, the power will be high and the false positive rate will be equal to our putative false rejection rate of $\alpha = .05$.

Table 3: Summary of power for different measures

Analysis	Method	Rejection rate	Mean estimate
false_positive_analysis	Likelihood Ratio	0.056	-33.088
false_positive_analysis	R-indicator	0.056	0.899
power_analysis	Likelihood Ratio	1.000	-594.289
power_analysis	R-indicator	1.000	0.560

The first column of Table 3 indicates which simulation is being run, the second column indicates what method is being used to assess NRB, the third column shows the rate of null hypothesis rejection at the .05 level, and the final column indicates the average test statistic.

The results indicate that both tests exhibit the correct rejection rates and are highly-powered for this particular analysis. The p-values are, on average, smaller for the likelihood ratio test. Therefore, we will report the p-values from both tests in our main analysis. If only one of the tests is significant at the .05 level, we will consider this suggestive evidence. If both are significant, we will consider this strong evidence that the null of no bias is false. If neither test is significant, we will consider that this particular test did not yield statistically signficant evidence of non-response bias.

3.4 Third measure: using the longitudinal nature of the survey to assess bias

The R-indicator investigates attributes that predict unit-level nonresponse during a focal survey wave, with lower values of the R-indicator indicating more systematic differences between respondents and nonresponders. Another way to explore unit-level nonresponse bias is to investigate how the relationships *between* different attributes vary between (1) units that are responders for multiple survey waves versus (2) units that "attrit" between waves, or that show up in one wave in which they were targeted for sampling but not another.

Tables 4 and 5 provide a partial view into the behavior of those sampled for the AHS in 2015 and 2017.[12] Table 4 focuses on units added in 2015 where an interview would have been possible (unit not under construction, unoccupied, excluded for technical reasons, etc.), and measures the number of non-interviews that arose due to behavioral ("type A") reasons: No one home, temporarily absent, refused, unable to locate, language problems, or other reasons an enumerator could not interview an occupied unit. Table 5 focuses on refusals. Overall, the tables show that there are not only persistent nonresponders (about 5 percent of the original sampling frame), but also those who respond in one

[12] The AHS sampling frame was redrawn in 2015. In 2017, the AHS interviewers visited the same housing units selected in the 2015 sample. More specifically, for the integrated national sample, which is composed of three sources—a representative sample of the nation; representative oversamples of the 15 largest metropolitan areas; representative oversamples of HUD-assisted units—the 2017 interviewers visited all the units selected in 2015. In this analysis of attritors, we restrict our attention to units added in 2015 (excluding units that were added to the sample frame in 2017 based on HUD choosing "newly constructed housing units") and that non-responded for "behavioral" reasons, such as refusal, absence, or language barriers. Units where no interview was conducted for technical reasons unrelated to survey interaction (house or mobile home moved, site unoccupied, under construction, sampling technicalities) are not included.

wave but refuse in another wave (about 10 percent of the original sampling frame). For these "sometimes responders," we can leverage responses from waves where they do respond for added insight.

Table 4: Panel attrition due to respondent behavior among units added in 2015

Category	N units
Interviewed 2015-2017	83,621 (76 percent)
Interviewed 2015, Not interviewed 2017	10,042 (9 percent)
Not interviewed 2015, Interviewed 2017	10,755 (10 percent)
Not interviewed 2015-2017	5,748 (5 percent)

Table 5: Panel attrition due to refusal among units added in 2015

Category	N units
Interviewed 2015-2017	89,433 (81%)
Interviewed 2015, Refused 2017	8,720 (8%)
Refused 2015, Interviewed 2017	8,400 (8%)
Refused 2015-2017	3,613 (3%)

Tests for heterogeneity in attrition

The table shows that about 10,000 respondents interviewed in 2015 were not interviewed in 2017, because the respondent was not home, could not communicate with the enumerator, or refused to participate in the survey.To assess potential bias in this attrition, we use two tests: first, we analyze whether covariates are jointly predictive of panel attrition through the use of F-tests; second, we use the Becketti, Gould, Lillard and Welch (BGLW) pooling test to explore potential bias caused by attrition between the two panels.[13]

Test 1: systematic differences among panel attritors

A first-order question for contexts with panel attrition is whether those who attrit have systematically different attributes from those who do not. We wish to understand both: what the "best" predictors of attrition are and whether, jointly, those predictors explain a statistically significant amount of the variation in year-on-year attrition. The test involves two steps:

1. We use Lasso penalized regression to select a minimal set of covariates from a focal survey wave (e.g., 2015) to predict a dummy for non-response in future survey waves (e.g., 2017 and 2019).[14]

2. Conduct an F-test of the joint significance of the covariates. If the F-test rejects the null that all coefficients on covariates are zero, this is evidence that the attrition produces bias in the sense that some types of units / respondents are more likely to drop out of the panel than others.

The code for the Lasso procedure is included in the appendix. We provide a dummy example of the code we would use

[13] In the final version, we will apply the test to attrition in multiple directions: units present in 2015 and 2017 but not in 2019; units present in 2015 and 2019 but not 2017; units present in 2017 and 2019 but not 2015 (if they were in the original sampling frame). To account for multiple panels, we will (1) stack the waves long-format, and then (2) reshape the data to wide format so that each respondents' values are observed across multiple waves and can be interacted with the attrition indicator.

[14] The lasso procedure that we plan to use features a generalized linear model with lasso penalization, and is implemented in the `glmnet` package for R. The loss function requires selecting a regularization parameter, lambda, that determines the severity of the penalty for including extra covariates. Since this regularization parameter cannot be optimally chosen in advance, we will select it using 10-fold cross-validation. Specifically, for each outcome, we will choose the lambda that minimizes the 10-fold cross-validation error averaged over 10 runs (since the folds are chosen at random). Only the covariates retained by the lasso will be included in the specification.

to run step 2 below.

```
# Test for heterogeneity --------------------------------------------------

# Subset the data to hhs that entered the sample in 2015 and where failure
# to interview in the subsequent year was not a result of mechanical or
# non-behavioral processes
ahs15 <-
  ahs15 %>% filter(YRINTRO == "2015" & type_B_17 == 0 & type_C_17 == 0)

# Covariates selected through CV'd lasso (see appendix code below)
lasso_covariates <- c("RACE1","YRBUILT","HOTWATER","INTMODE","DIVISION")

null_model_type_A_17 <-
  lm(formula = reformulate(termlabels = "1",
                           response = "type_A_17"),
     data = ahs15)
lasso_model_type_A_17 <-
  lm(formula = reformulate(termlabels = lasso_covariates,
                           response = "type_A_17"),
     data = ahs15)
f_test_type_A_17 <- anova(
  null_model_type_A_17,
  lasso_model_type_A_17
)

null_model_refusal_17 <-
  lm(formula = reformulate(termlabels = "1",
                           response = "refusal_17"),
     data = ahs15)
lasso_model_refusal_17 <-
  lm(formula = reformulate(termlabels = lasso_covariates,
                           response = "refusal_17"),
     data = ahs15)
f_test_refusal_17 <- anova(
  null_model_refusal_17,
  lasso_model_refusal_17
)
```

Test 2: heterogeneity in attrition
The previous analysis investigates how attributes relate to a respondent's likelihood of attriting from the survey. Another way of examining potential bias caused by this attrition is to investigate how the attrition changes correlations that researchers might be interested in examining. For instance, if a researcher is interested in investigating how the relationship between housing adequacy and eviction changes over time, but, for instance, those living in inadequate housing who are also evicted are much more likely to attrit than others, this nonrandom attrition causes particular bias for investigating longitudinal trends.

To assess this form of bias, we use the Becketti, Gould, Lillard and Welch (BGLW) pooling test to explore potential bias caused by attrition between the two panels, defined as attrition regardless of the specific cause.[15]

The below code calculates heterogeneity in attrition using the 2015 wave, an attrition indicator that indicates a unit was missing from the 2017 wave, the outcome variable of whether the unit fails the adequacy criteria,[16] and a limited set of explanatory variables.[17]

```
run_attrit_het_reg <- function(data, attrit_varname,
                               attributes_vector, outcome_ofint){

  ## interact each attribute with the attrition
  interact_all_attrit = paste(unlist(lapply(attributes_vector,
       function(x) {sprintf("%s*%s",
                            x, attrit_varname)})),
      collapse = "+")

  ## feed to reg with outcome variable
  attrit_formula = formula(sprintf("%s ~ %s",
                                   outcome_ofint,
                                   interact_all_attrit))

  ## estimate regression
  model_obj = lm(attrit_formula, data = data)

  return(model_obj)

}

calculate_F_attrit <- function(model_obj, attrit_varname){

  get_interact = grep(sprintf("%s:|:%s", attrit_varname, attrit_varname),
                    names(model_obj$coefficients),
                    value = TRUE)
  f_formula = sprintf("%s = 0", get_interact)
  f_test = linearHypothesis(attrit_het_results, f_formula)
  return(f_test)


}

attributes_relevant = sprintf("%s_labeled", vars_avail_nonint)
attrit_het_results = run_attrit_het_reg(data = ahs2015_updated,
```

[15] In the final version, we will apply the test to attrition in multiple directions: units present in 2015 and 2017 but not in 2019; units present in 2015 and 2019 but not 2017; units present in 2017 and 2019 but not 2015 (if they were in the original sampling frame). To account for multiple panels, we will (1) stack the waves long-format, and then (2) reshape the data to wide format so that each respondents' values are observed across multiple waves and can be interacted with the attrition indicator.

[16] More specifically, we used the ADEQUACY variable and constructed a binary measure of the unit not being adequate if the response was either moderately or severely inadequate.

[17] For simplicity, we use the same three variables we used for the R-indicator analysis: the CBSA, the census division, and the building type. However, since we are analyzing outcomes among those who responded to a particular wave, for the final analysis, we may expand to include other variables that seem relevant for predicting whether a unit is suffering from adequacy issues.

```
                                    attrit_varname = "attrit",
                                    attributes_vector = attributes_relevant,
                                    outcome_ofint = "inadequate_binary")
f_test = calculate_F_attrit(attrit_het_results,
                            attrit_varname= "attrit")

p_ftest = f_test$`Pr(>F)`[2]
```

The table below summarizes the results, subsetting to the interaction terms for building type for the purposes of presentation. The table shows that there is a different relationship between building type and adequacy among those who attrit from the sample than among those who were interviewed again in 2017. The value of the F-test across all interaction terms is 1.711, with p = 0.007.

Table 6: Heterogeneity between attritors and non-attritors

	Dependent variable:
	inadequate_binary
attrit:BLD_labeledOne-family house, detached	−0.031 (0.013) p = 0.017**
attrit:BLD_labeledOne-family house, attached	−0.041 (0.016) p = 0.011**
attrit:BLD_labeled2 Apartments	−0.024 (0.019) p = 0.197
attrit:BLD_labeled3-4 Apartments	−0.019 (0.018) p = 0.277
attrit:BLD_labeled5-9 Apartments	−0.022 (0.017) p = 0.199
attrit:BLD_labeled10-19 Apartments	−0.014 (0.018) p = 0.437
attrit:BLD_labeled20-49 Apartments	−0.006 (0.019) p = 0.751
attrit:BLD_labeled50 or more apartments	−0.037 (0.017) p = 0.030**
attrit:BLD_labeledBoat, RV, van, etc.	−0.141 (0.117) p = 0.231
Observations	69,493
R^2	0.013
Adjusted R^2	0.012
Residual Std. Error	0.266 (df = 69425)
F Statistic	14.023*** (df = 67; 69425)
Note:	*p<0.1; **p<0.05; ***p<0.01

4 Plan to assess item-level nonresponse bias

The previous analyses focused on an extreme form of item-level nonresponse: a unit does not start an interview, so is missing on all items.

For item-level nonresponse bias, which we can decompose into (1) bias stemming from unit-level nonresponders who did not respond to any items, and (2) bias stemming from those who responded to the survey in general but did not complete a particular item, we have more information to assess bias. In particular, rather than only looking at variables available for both respondents and nonresponders via the sampling frame or other sources, we can use information from items the unit did complete to investigate bias in ones they did not.

4.1 Descriptive exploration of item-level nonresponse

Before moving to measures of bias from item-level nonresponse, we first descriptively explore which items have higher rates of missingness. The AHS uses two methods to treat missing values:

1. The majority of variables for which there is item-level missingness have values imputed, with an ancillary variable then created, the "imputation flag" variable, that indicates which respondents have imputed values for the respective variable. The main variable then contains these imputed values.
2. A smaller subset of variables is not imputed, and the main variable contains missing values.

Figure 2 shows the top 20 items with the most imputation;[18] Figure 3 shows the top 20 items, among those not imputed, that have the highest rate of nonreport. Focusing on items with high rates of nonreport, we see some patterns like potentially-sensitive items about neighborhood safety and family members' care needs having higher, item-level missingness.

[18] This was calculated by (1) looking at variables that have the J prefix indicating an edit flag and (2) looking at the proportion of responses in the 2017 PUF that have a value of 2 for that edit flag variable.

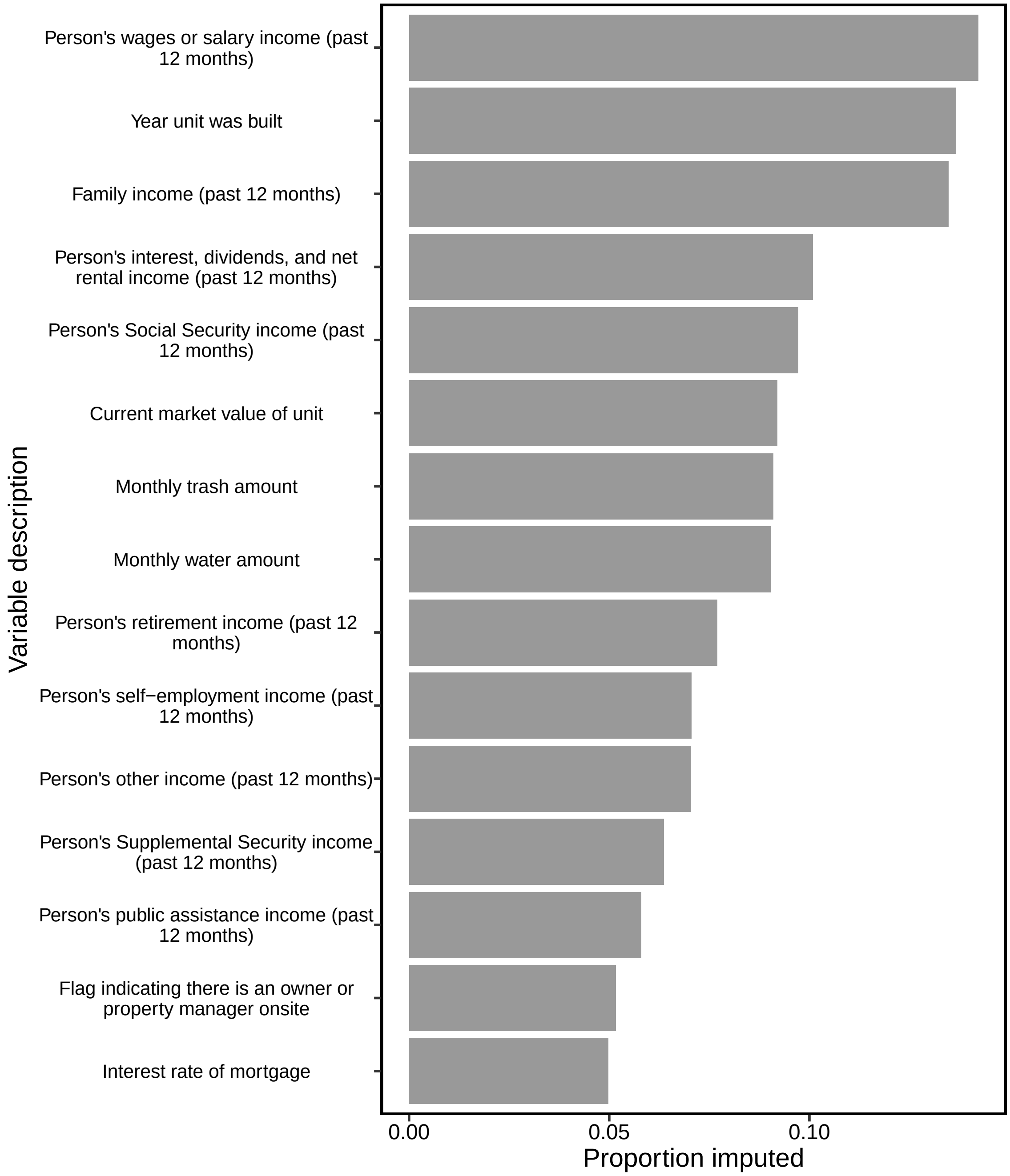

Figure 2: Top 20 items with the highest rates of imputation

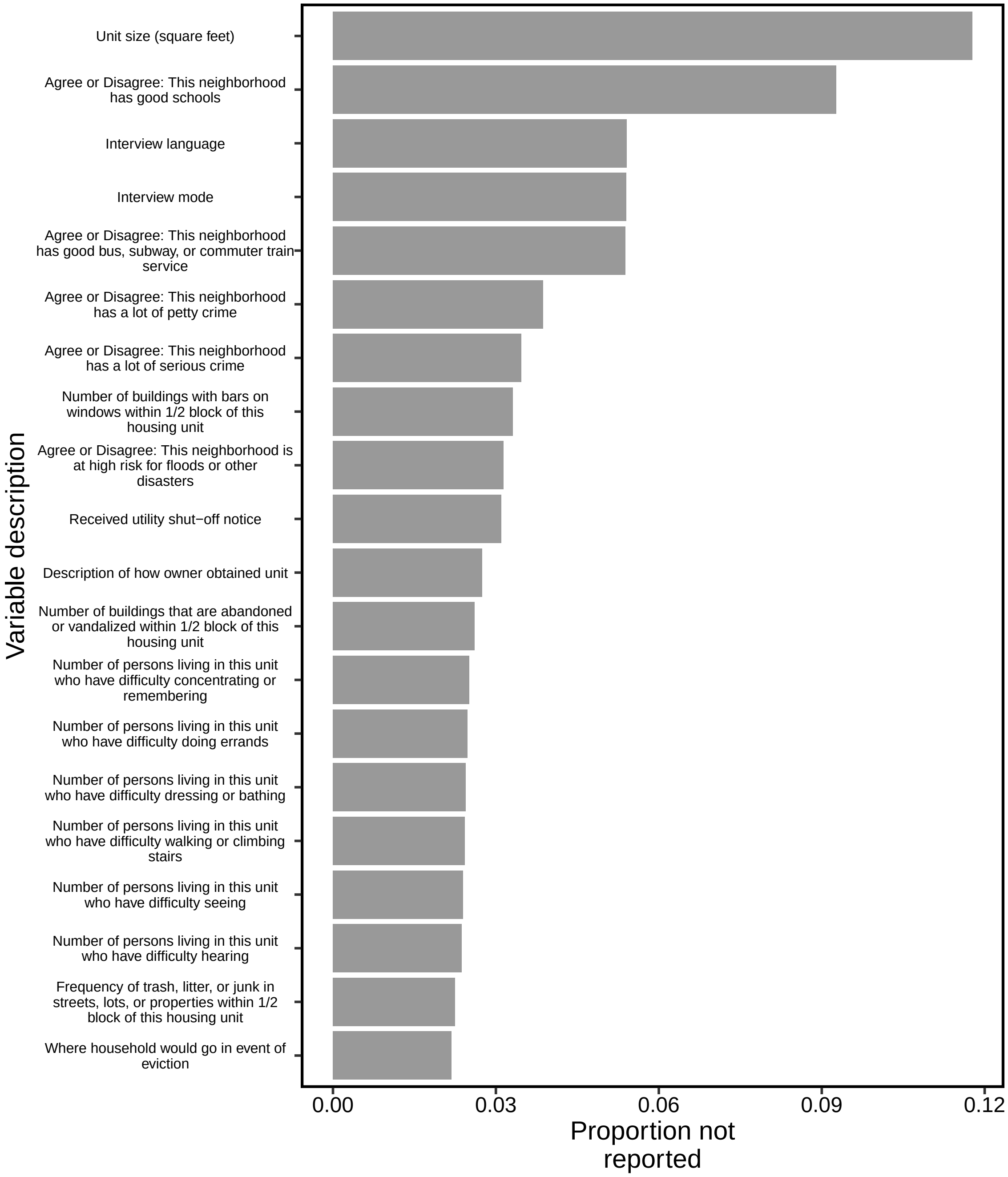

Figure 3: Top 20 items with the highest rates of non-imputed missing values

4.2 First measure of item-level nonresponse bias: explore impact of an item's placement in the survey on response

Figures 2 and 3 show certain items have high levels of missingness–for instance, the unit's market value and perceptions of whether the neighborhood has good schools. This missigness could stem from two sources:

- Missingness due to the item itself– for instance, sensitive questions having higher missingness; or
- Missingness due to the item appearing later in the survey, a point at which respondents may have more survey

fatigue and may either be (1) more likely to stop the survey altogether, or (2) complete the survey but skip more items to reduce time.

To examine these two sources, we will use data that timestamps each item (indexed by k) for each respondent (indexed by i) to calculate the following:

Relative duration of item (for those to whom the item was posed): Time item$_{ik}$ $-$ Time start$_{ik}$

We will then estimate the following model with linear regression, with γ_i as a respondent-specific fixed effect that captures general response propensities across items and δ_k as an item-specific fixed effect that captures general propensity to respond to an item net of its order:

$$\text{Respond to item (1 = yes)}_{ik} = \alpha + \beta_1 \text{Relative duration}_{ik} + \gamma_i + \delta_k + \epsilon_{ik}$$

The model thus exploits between-respondent variation in when an item occurs relative to the start of the survey for different respondents (e.g., due to different skip logics). If the coefficient on β_1 is significant and negative, it means that respondents are less likely to respond to items later in the survey.[19]

4.3 Second measure of item-level nonresponse bias: compare sample means for attributes measured in AHS 2017 to a benchmark data source

The order effects analysis allows us to estimate the impact of an item's placement on nonresponse. Another way to measure bias, which can occur regardless of an item's placement, is to compare summary statistics for an item to summary statistics from benchmark data sources.

The previous section shows, for instance, that items measuring respondents' perceptions of their neighborhood have higher than average levels of missingness.[20] While items like that one are unique to the AHS, they may be correlated with neighborhood demographics. One way to assess item-level bias is thus to:

- Find a benchmark data source, a data source that is either (1) a sample that likely has less nonresponse bias than the AHS, or (2) administrative data that contains the universe of program participants;
- Examine attributes measured in both the benchmark data source and the AHS; and
- Compare the two, with divergence indicating potential bias. We will conduct using t-tests of the differences for each attribute, where each row is a CBSA and we adjust for multiple testing using the Holm correction. We will also conduct an omnibus test for the joint significance of all differences, using the d^2 described in Hansen and Bowers (2008).
- For the Census benchmark analysis, conduct the above process for two versions of the AHS data, using the multivariate $\mathcal{L}_1$ distance between the AHS versus Census to summarize how much the re-weighting decreases the distance:

    1. The data after we have applied the weights for oversampling, but before we have applied the current AHS nonadjustment weights: this provides a comparison of how well the AHS compares before the sample is

[19] We will estimate two versions. One will subset to respondents who reach the end of the surveys but skip certain items, which will capture the order effects only among those who skip items but reach the end. The other will include respondents who start the survey, and the order effects will thus capture a combination of (1) people skipping later items, (2) people ending the survey early and thus never being posed later items.

[20] For the final analysis, we will try to link information on question ordering to investigate the extent to which item-level missingness is related to intrinsic features of a question or whether later items, regardless of content, tend to have higher missingness.

reweighted using current methods for nonresponse bias.[21]

2. The data after we have applied the weights for both oversampling and for nonresponse adjustment

What will we use as the benchmark data source?
To illustrate the benchmark analysis, we use the public-use files of the American Community Survey (ACS) 2017 5-year estimates as the benchmark sample for the 2017 AHS, reporting differences at the CBSA level.[22] For the final analysis, we aim to use:

1. **Microdata from HUD on the demographic characteristics of those in HUD-assisted units:** since the AHS oversamples those in HUD-assisted units, and since the HUD administrative data contains the full universe of program recipients rather than a sample, this would provide a valid benchmark for that subset of individuals.

2. **2010 Decennial Census:** While the HUD benchmark data can help us compare characteristics for HUD-assisted units, to compare attributes for the full-sample, we will use demographic variables from the 2010 decennial census aggregated to the CBSA level. We will filter each data source to (1) heads of households, (2) living in occupied units within the AHS sampling frame (i.e., exclude those in the Census living in group quarters).

[21] In particular, the AHS oversamples two sets of units. First are those located in the 15 largest metropolitan areas, which gives us sufficient sample size to conduct a CBSA-level benchmark analysis. Second are HUD-subsidized housing units. This latter oversample complicates the benchmark analysis, since the demographics of those in HUD-subsidized units diverge from the demographics of the CBSA as a whole. The analysis with the IUF has a weight that accounts for oversampling but that, importantly, does not contain a noninterview adjustment factor. In contrast, the 2017 AHS PUF only contains the ”final weight” variable that contains both a noninterview adjustment factor and oversampling. It does not contain the ”basic weight” variable that only accounts for the oversampling. For both the ACS and the AHS, we will use this weight to reweight by the inverse probability of selection. We will use this to reweight the AHS data to adjust for oversampling of assisted units.

[22] We look at attributes at the CBSA level, since: (1) that level of geography is contained in the PUF, and (2) has enough respondents to reliably estimate summary statistics (in contrast to levels like the tract that contain fewer respondents).

Illustrating the approach

We focus on X, covariates that are present in both the AHS and in other surveys,[23] and calculate the following, where $\bar{X}$ represents the mean or proportion of the attribute in a benchmark survey and $E[\hat{X}]$ represents the expected estimate of its mean across repeated samples in the AHS. The further away the quantity is from zero, the more potential there is for bias:

$$D = \bar{X} - E[\hat{X}]$$

For X, we include:[24]

- Race;
- Educational attainment; and
- Whether receiving SNAP benefits.

Figure 4 shows the differences. The analysis is illustrative, since the PUF does not allow us to account for oversampling without also accounting for unit-level nonresponse—so, for instance, New York City's lower rate of White respondents in the ACS than the AHS might stem from the composition of public housing residents. With these limitations in mind, we see variation across CBSAs in the distance between the AHS and ACS proportions, with areas like Boston and New York City having less divergence than areas like Miami.

[23] The analysis assumes that the question wording for these items is also the same across surveys; however, differences, for instance, in how race/ethnicity options are presented could lead to divergence in items like who identifies as being of some "other" racial group.

[24] We may expand these to look at other demographic variables that are correlated with important AHS variables.

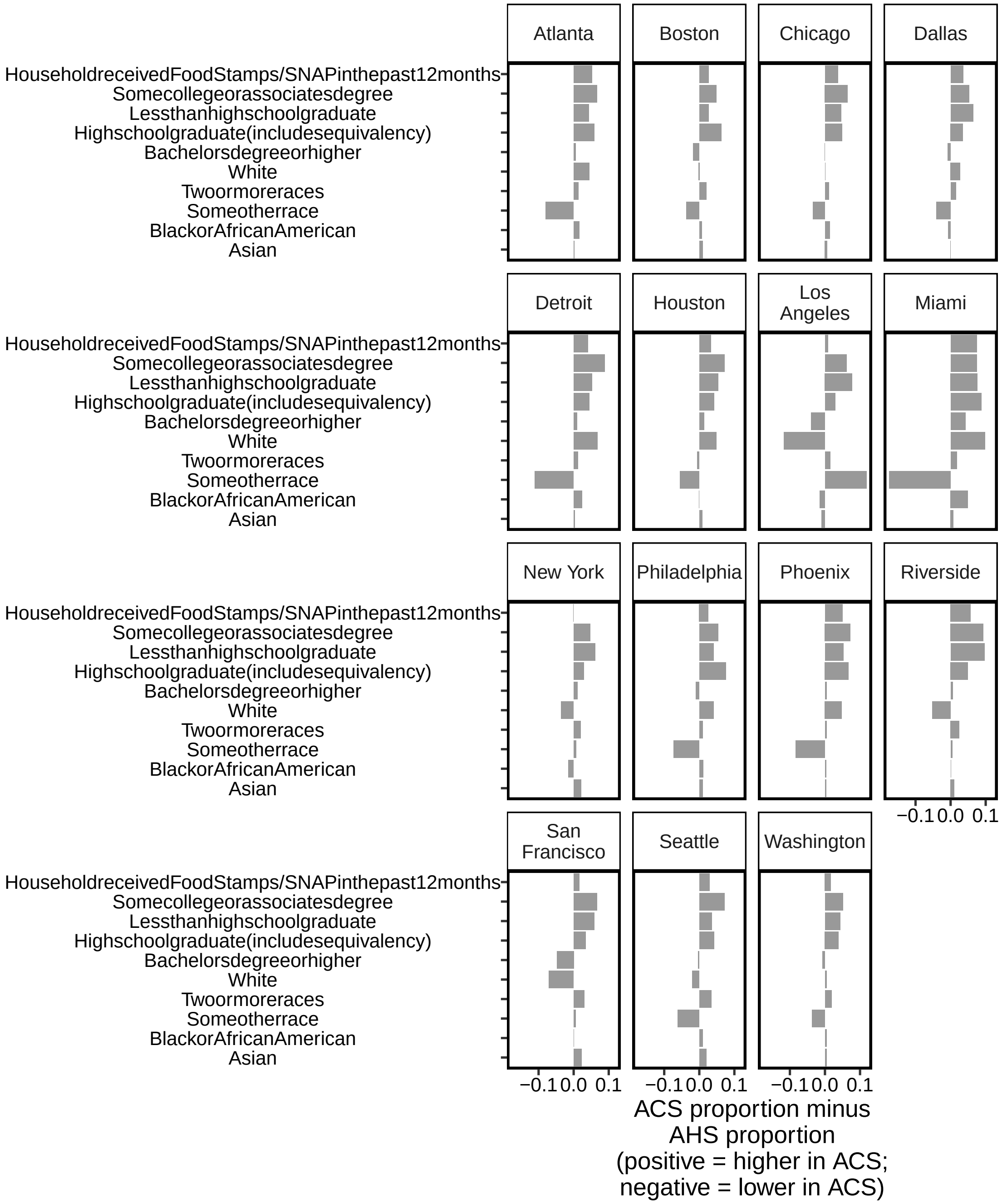

Figure 4: Comparing demographics between AHS and ACS at CBSA level. We do not yet have the appropriate weights to adjust the AHS rates to account for the oversampling of HUD-assisted units, so the graph is illustrative.

4.4 Third measure of item-level bias: fraction of missing information (FMI)

A drawback of the benchmark analysis is that it focuses on demographic variables that are measured more reliably in another data source. Since the main value of the AHS is in measuring attributes that are *not* measured elsewhere, we use a second measure of item-level bias that can be calculated for all items, rather than only items that the AHS shares with other data sources. In particular, we will use the Fraction of Missing Information (FMI), a metric that is typically used for imputation. Wagner (2010) argues that, if we think of nonresponse for question X as a form of missing data, we can repurpose the FMI to analyze potential for nonresponse bias from the Missing at Random mechanism.

FMI: definition
The method involves:

1. Use multiple imputation to relate the observed variables (for nonresponders to the survey, this is data from the sampling frame; for respondents, this is data from the sampling frame and the survey itself) to missing values, in order to impute the latter;

2. For a particular item, calculate the between-imputation variance, which is defined as the variance of the estimates between each of the imputed datasets; and

3. The larger the between-imputation variance and FMI, the larger the assumed bias on an item (since imputation based on observed values does a worse job). This means that we have less information than we could from the item due to the difficulty imputing values.

FMI: calculation
We will focus on the FMI values for (1) the top 20 items with the most missingness, but that have values imputed in the survey (Figure 2),[25] and (2) the top 20 items with the most missingness, but that do not have values imputed in the survey (Figure 3). For the final analysis, we will perform the imputation separately for two groups:

- Nonresponders, whose missingness on the item is a function of them not having completed any of the survey, and who also have many fewer observed attributes to use for imputation; and
- Respondents who are missing values for that particular item: the missingness thus reflects opting out of a question, a different source, and these respondents also have many more attributes to use for imputation.

We will use the `CART` method in the `mice` package with $m = 20$ replicates. We opt for the CART (Classification and regression trees) method over parametric methods that employ, for example, multivariate normal distributions because the latter run into many issues with collinearities when using large, sparse matrices. By contrast, CART is fully non-parametric and flexible to the data, and is robust to collinearities.

5 Summary

Table 7 summarizes the unit-level and item-level measures we will use to investigate nonresponse bias in the AHS. We may add additional analyses, like ones that test the sensitivity of our measures of bias to violations of the missing at random assumption (Andridge and Little 2011). Taken together, the analyses will help us *characterize bias* to improve the design of an incentives experiment to potentially *reduce bias*.

Table 7: Summary of measures

Level	Measure
Unit	Response rate
Unit	R-indicator
Unit	Likelihood ratio test
Unit	Panel attrition; Attritor heterogeneity
Item	Order effects analysis
Item	Benchmark analysis
Item	Fraction of missing information

[25] This will rely on intermediate versions of the survey before the final PUF, since the PUF just contains imputed values for these items accompanied by an edit flag.

6 Appendix

6.1 Python code for binary classifiers

```
def get_features_labels(ids, features, labels):
    feature_cols = list(set(features.columns).difference(['CONTROL', 'fold']))

    label_nonarray = labels.loc[labels.Research_ID.isin(ids)]
    label = np.array(label_nonarray[['FSI_invite']])
    features = np.array(features.loc[features.Research_ID.isin(ids), feature_cols])
    ids_toreturn = label_nonarray.Research_ID
    print("Dimensions of feature matrix:" + str(features.shape))
    return(label, features, ids_toreturn)

def evaluate_models(y_pred, label_test):
    all_results = precision_recall_fscore_support(label_test,
                                            y_pred)
    all_results_1 = [i[0] for i in all_results][0:3]
    return(all_results_1)

## function to estimate models across folds
def estimate_models(model_list, names_list, features, labels):
    evals_df = {}
    store_pred_allmodels = {}
    for j in range(0, len(model_list)):
        ## pull out model
        one_model = model_list[j]
        print("fitting model: " + str(one_model))
        ## iterate over folds to estimate and evaluate
        store_evals_fold = []
        store_pred_allfolds = []
        for i in range(1, 6):
            ## ids for fold
            which_fold = [i]
            train_folds = list(set(list(range(1, 6))).difference(which_fold))
            train_ids = features.Research_ID[features.fold.isin(train_folds)]
            test_ids = features.Research_ID[features.fold.isin(which_fold)]
            ## label and features
            (label_train, training_features,
              train_final_ids) = get_features_labels(train_ids, features, labels)
            (label_test, test_features,
              test_final_ids) = get_features_labels(test_ids, features, labels)
            ## fit the model and evaluate
            print("estimating for fold:" + str(i))
            one_model.fit(training_features, label_train)
            print("estimated model")
            y_pred = one_model.predict(test_features)
            y_score = one_model.predict_proba(test_features)[:, 1]
            print("generated predictions")
            ## store predictions
            store_pred_allfolds.append(pd.DataFrame({'CONTROL': test_final_ids,
                               'Model': names_list[j],
                                'Binary_pred': y_pred,
                                'Score': y_score,
                                'Observed_label': label_test.tolist()}))
            ## store evaluations
            store_evals_fold.append(evaluate_models(y_pred, label_test))
            evals_df[names_list[j]] = np.mean(store_evals_fold, 0)
            store_pred_allmodels[names_list[j]] = pd.concat(store_pred_allfolds)
    return(evals_df, store_pred_allmodels)

def evalsarray_to_df(eval_array, model_name):
  eval_df= pd.DataFrame.from_dict(eval_array, orient = "index")
  eval_df.columns = accuracy_metrics
  eval_df['type'] = model_name
  eval_df['model'] = eval_df.index
  return(eval_df)

def predict_to_df(predictions, model_name):
  pred_df = pd.concat(predictions).reset_index()
  pred_df['type'] = model_name
  return(pred_df)

############################## Full classifiers
## create a list of model objects
classifiers_list = [DecisionTreeClassifier(random_state=0, max_depth = 5),
                    DecisionTreeClassifier(random_state=0, max_depth = 50),
                    RandomForestClassifier(n_estimators = 100,
                                           max_depth = 20),
                    RandomForestClassifier(n_estimators = 1000,
                                           max_depth = 20),
                    GradientBoostingClassifier(criterion='friedman_mse',
                                               n_estimators=100),
                    GradientBoostingClassifier(criterion='friedman_mse',
                                               n_estimators=1000),
                AdaBoostClassifier(),
                LogisticRegression(),
                LogisticRegressionCV(),
                LogisticRegression(penalty = "l1"),
                LogisticRegressionCV(solver = "liblinear",
                                 penalty = "l1")]
print("Length of classifier list is:" + str(len(classifiers_list)))
names_list = ['dt_shallow', 'dt_deep',
              'rf_few', 'rf_many',
              'gb_few', 'gb_many',
              'ada',
              'logit', 'logitcv', 'logitl1',
              'logitl1cv']
print("Length of classifier list is:" + str(len(names_list)))


############################## Estimate models on test classifiers
(model_evaluations, model_predictions) = estimate_models(classifiers_list,
                          names_list,
                          AHS_features,
                          AHS_responselabel)
```

6.2 Code for lasso covariate selection

```
# Core functions for the lasso analysis -----------------------------------

# This function performs k-fold cross-validation, calculating the
# tuning parameter (lambda) with the lowest average (across folds)
# test error.

lasso_cv <- function(outcome_name, covariates, data, N_folds = 30, ...){
    # glmnet only takes matrices
    Y <- as.matrix(data[ ,outcome_name])
    X <- as.matrix(data[ ,covariates])

    cv.glmnet(x = X, y = Y,
                        # Use MSE minimization for CV
                        type.measure = "deviance",
                        # Number of folds
                        nfolds = N_folds,
                        # Alpha = 1 sets glmnet() to use lasso penalty
                        alpha = 1,
                        ...)
}

# This function takes a set of cross-validated lasso models and returns the
# non-zero coefficients from the model that uses a lambda that minimizes the
# mean cross-validated error

tidy_lasso_covariates <- function(lasso_fit, lambda = "lambda.min"){
    coefs <- coef(lasso_fit, s = lambda)
    coefs <- ifelse(is.list(coefs),
                                    as.list(coefs),
                                    list(coefs))
    coefs %>%
        do.call(what = cbind) %>%
        rowMeans() %>%
        data.frame() %>%
        rownames_to_column() %>%
        select(rowname, ".") %>%
        rename(term = rowname, estimate = ".") %>%
        filter(estimate > 0) %>%
        filter(term != "(Intercept)")
}

# This function is a wrapper for lasso_cv and tidy_lasso_covariates that
# performs cross-validated lasso sims times, averages the lambdas that
# minimize CV error across the sims runs, and returns the covariates selected
# using that average lambda

select_covariates <- function(outcome_name, covariates, data, N_folds = 30, sims = 10,...){
    print(paste0("Selecting covariates for ", outcome_name))
    # Do k-fold CV sims times, returning sims CV models
    lambdas <- lapply(X = 1:sims,
                                        FUN = function(i)
                                          lasso_cv(outcome_name = outcome_name,
                                                            covariates = covariates,
                                                            data = data,
                                                            N_folds = N_folds,
                                                            ... = ...))
    # Get lambda that minimizes mean CV error for each
    min_lambdas <- sapply(lambdas, get, x = "lambda.min")
    # Use the first model (doesn't matter which one) with
    # average lambda to get covariates with optimal lambda
    tidy_lasso_covariates(
        lasso_fit = lambdas[[1]],
        lambda = mean(min_lambdas)) %>%
        mutate(outcome = outcome_name) %>%
        select(outcome, term, estimate)
}
```

References

Andridge, Rebecca R, and Roderick JA Little. 2011. “Proxy Pattern-Mixture Analysis for Survey Nonresponse.” *Journal of Official Statistics* 27 (2): 153.

Bailey, Michael. 2019. “Designing Surveys to Account for Non-Ignorable Non-Response.” *Working Paper*.

Groves, Robert M, and Emilia Peytcheva. 2008. “The Impact of Nonresponse Rates on Nonresponse Bias: A Meta-Analysis.” *Public Opinion Quarterly* 72 (2): 167–89.

Schouten, Barry, Fannie Cobben, Jelke Bethlehem, and others. 2009. “Indicators for the Representativeness of Survey Response.” *Survey Methodology* 35 (1): 101–13.

Wagner, James. 2010. “The Fraction of Missing Information as a Tool for Monitoring the Quality of Survey Data.” *Public Opinion Quarterly* 74 (2): 223–43.