CHAPTER 7

POST-STRATIFICATION, RAKING, AND CALIBRATION

In which the whole tells us about some of the parts.

7.1 INTRODUCTION

Chapter 2 showed the increase in precision that can come from using population data to stratify sampling. Stratification is not always a desirable way to use these population data: there may be too many potential stratification variables, the best strata may be different for different analyses, or the need for cluster sampling may prevent stratification on individual-level variables. Population data may also be available in a form that does not allow for stratification — random-digit dialing, for example, cannot easily stratify on any individual characteristics, because these characteristics are not known before dialing and so cannot be used to select the sample.

This chapter deals with techniques for using known population totals for a set of variables (*auxiliary variables*) to adjust the sampling weights and improve estimation for another set of variables. All of these techniques have the same idea: adjustments

Complex Surveys: A Guide to Analysis Using R. By Thomas Lumley

are made to the sampling weights so that estimated population totals for the auxiliary variables match the known population totals, making the sample more representative of the population. A second benefit is that the estimates are forced to be consistent with the population data, improving their credibility with people who may not understand the sampling process.

There are two quite different applications of these techniques. One is to increase precision of estimation, the other is to reduce the bias from nonresponse, especially *unit non-response*, where a sampled individual refuses to participate or otherwise provides no information for analysis. From the computational viewpoint the main difference between these applications is in the criteria for choosing auxiliary variables, which are discussed in section 7.6. The use for non-response is the more important of the two in large-scale surveys, but it has largely been completed by the time the data arrive at the typical user. Calibration to improve precision will be important in the two-phase epidemiologic designs discussed in Chapter 8. *Item non-response*, where some variables but not others were observed for an individual is addressed in Chapter 9.

7.2 POST-STRATIFICATION

The simplest technique for adjusting sampling weights is *post-stratification*. Suppose we have a division of the population into groups. One possible design would be a stratified random sample using these groups as strata, sampling n_k individuals from population stratum k containing N_k individuals. The sampling weights are $1/\pi_i = N_k/n_k$ and the Horvitz–Thompson estimator $\hat{N}_k$ of the population group size is exactly correct and its standard error is zero. Estimation of other population totals is improved by removing the variability between groups: the contribution of each group to a population total is fixed by design, rather than random.

If N_k were known, but the sampling was not stratified, the estimated population group sizes would not be exactly correct. Post-stratification adjusts the sampling weights so that the estimated population group sizes are correct, as they would be in stratified sampling. The sampling weights $1/\pi_i$ are replaced by weights g_i/π_i where $g_i = N_k/\hat{N}_k$ for the group containing individual i. The estimated group size for the kth group will then be

$$n_k \times \frac{g_i}{\pi_i} = n_k \times \frac{1}{\pi_i} \times \frac{N_k}{\hat{N}_k} = \hat{N}_k \times \frac{N_k}{\hat{N}_k} = N_k.$$

The estimated group size is always the same as the actual group size, and just as for a stratified sample the contribution of each group to population totals is now fixed. These groups are often called *post-strata*.

One issue has been glossed over here. If we are unlucky enough to sample no-one from group k it is not possible to perform the re-weighting. As long as the expected sample sizes in each group are not too small, the probability of ending up with no observations in a group is very small. For example, if the expected sample size in the group is only 10, the probability of having at least one observation in the group is

99.995%. On the rare occasions when there are no observations in the group it would be necessary to merge two post-strata or make some other *ad hoc* correction. If we approximate the standard errors by neglecting the very rare occasions when a group ends up empty, post-stratification on the group variable does give the same standard errors as sampling stratified on the group variable with the same group sample sizes.

The same correction can be used for any sampling design, not just for simple random sampling. If the totals $\{N_k\}$ for a set of population groups are known and the Horvitz–Thompson estimates $\{\hat{N}_k\}$ are computed, the adjustment g_i to the sampling weight for individual i is $N_k/\hat{N}_k$ for the group k containing the individual. As the sample size increases, $\hat{N}_k$ will become more accurate and so the adjustment $g_i = N_k/\hat{N}_k$ will become closer and closer to 1, but the proportional reduction in standard errors from post-stratification will remain roughly constant.

Adjusting the weights to incorporate the auxiliary information is straightforward, but estimating the resulting standard errors is more difficult. For replicate-weight designs it is sufficient to post-stratify each set of replicate weights (Valliant [178]). Since the estimated group size will then match the known population group size for every set of replicates, between-group variation will not produce differences between replicates and so the replicate-weight standard errors will correctly omit the between-group differences.

Standard errors for a population total are computed by decomposing the variable into stratum means and residuals. Writing μ_k for the true population mean in group k and $\mu_{k(i)}$ for the mean in the group containing observation i, $Y_i = (Y_i - \mu_{k(i)}) + \mu_{k(i)}$. The variance of the estimated total of Y is

$$\text{var}\left[\hat{T}_y\right] = \text{var}\left[\sum_{i=1}^{n} \frac{1}{\pi_i}\left(Y_i - \mu_{k(i)}\right)\right] + \text{var}\left[\sum_{i=1}^{n} \hat{N}_{k(i)} \frac{\mu_{k(i)}}{\pi_i}\right] \qquad (7.1)$$

and because $\hat{N}_k$ is fixed, the second variance is zero. This argues for estimating the variance of a post-stratified estimator by subtracting off the group means and taking the Horvitz–Thompson estimator of the variance of the residuals. The variance estimator will not be unbiased, since the residuals from the estimated group mean will tend to be smaller than the residuals from the true group mean. When post-stratifying a simple random sample this bias could be computed and corrected by multiplying group k's contribution to the variance by $n_k/(n_k - 1)$, but there is no simple correction that works for cluster-sampled or multistage designs. The bias in the variance estimator can safely be ignored as long as the group sizes n_k are not too small. The same procedure extends to summary statistics that solve a population equation: the contributions to this population equation are centered at the group means before computing their variances [133].

Two issues this leaves open are whether the group means should be estimated using $1/\pi_i$ or g_i/π_i as weights, and whether $1/\pi_i$ or g_i/π_i should be used as weights when computing the Horvitz–Thompson variance estimator. In the absence of non-response this choice makes very little difference, but in the presence of substantial non-response it does matter (Kott [84]). The survey package uses $1/\pi_i$ for the stratum means and g_i/π_i in the Horvitz–Thompson estimator.

```
> data(api)
> clus2_design <- svydesign(id=~dnum+snum, fpc=~fpc1+fpc2,
      data=apiclus2)
> pop.types <- data.frame(stype=c("E","H","M"),
     Freq=c(4421,755,1018))
> ps_design <- postStratify(clus2_design, strata=~stype,
      population=pop.types)
> svytotal(~enroll, clus2_design, na.rm=TRUE)
         total     SE
enroll 2639273 799638
> svytotal(~enroll, ps_design, na.rm=TRUE)
         total     SE
enroll 3074076 292584
> svymean(~api00, clus2_design)
        mean     SE
api00 670.81 30.099
> svymean(~api00, ps_design)
      mean     SE
api00  673 28.832
```

Figure 7.1 Post-stratifying a two-stage sample of schools on school type

The function `postStratify()` creates a post-stratified survey design object. In addition to adjusting the sampling weights, it adds information to allow the standard errors to be adjusted. For a replicate-weight design this involves computing a post-stratification on each set of replicate weights; for linearization estimators it involves storing the group identifier and weight information so that between-group contributions to variance can be removed.

Example: Post-stratifying on school type. In section 3.2 we examined a two-stage sample drawn from the API population, with 40 school districts sampled from California and then up to five schools sampled from each district. Post-stratifying this design on school type illustrates the situations where improvements in precision are and are not available. The code and output are in Figure 7.1.

The first step is to set up the information about population group sizes. This information can be in a data frame (as here) or in a `table` object as is produced by the `table()` function. In the data frame format, one or more columns give the values of the grouping variables and the final column, which must be named `Freq`, gives the population counts. The call to `postStratify()` specifies the grouping variables as a model formula in the `strata` argument, and gives the table of population counts as the `population` argument. The function returns a new, post-stratified survey design object.

Post-stratification on school type dramatically reduces the variance when estimating total school enrollment across California. The standard error was approximately

800,000 for the two-stage sample and less than 300,000 for the post-stratified sample. A large reduction in variance is possible because elementary schools are about half the size of middle schools and about one-third the size of high schools on average. Post-stratification removes the component of variance that is due to differences between school types, and this component is large.

When estimating the mean Academic Performance Index there is little gain from post-stratification. A standardized assessment of school performance should not vary systematically by level of school or it would be difficult to interpret differences in scores. Since API has a similar mean in each school type, the between-level component of the variance is small and post-stratification provides little help. If this were a real survey, non-response might differ between school type, and post-stratification would then be useful in reducing non-response bias.

7.3 RAKING

Post-stratification using more than one variable requires the groups to be constructed as a complete cross-classification of the variables. This may be undesirable: the cross-classification could result in so many groups that there is a risk of some of them not being sampled, or the population totals may be available for each variable separately, but not when cross-classified.

Raking allows multiple grouping variables to be used without constructing a complete cross-classification. The process involves post-stratifying on each set of variables in turn, and repeating this process until the weights stop changing. The name arises from the image of raking a garden bed alternately in each direction to smooth out the soil. Raking can also be applied when partial cross-classifications of the variables are available. For example, a sample could be alternately post-stratified using a two-way table of age and income, and a two-way table of sex and income. The resulting raked sample would replicate the known population totals in both two-way tables.

The effect of matching observed and expected counts in lower-dimensional subtables is very similar to the effect of fitting a hierarchical loglinear model as described in section 6.3.1. In fact, the raking algorithm is widely used to fit loglinear models, where it is known as *iterative proportional fitting*. Raking could be considered a form of post-stratification where a loglinear model is used to smooth out the sample and population tables before the weights are adjusted.

Standard error computations after raking are based on the iterative post-stratification algorithm. For standard errors based on replicate weights, raking each set of replicate weights ensures that the auxiliary information is correctly incorporated into between-replicate differences. For standard errors based on linearization, iteratively subtracting off stratum means in each direction gives residuals that incorporate the auxiliary information correctly.

The `rake()` function performs the computations for raking by repeatedly calling `postStratify()`. Each of these calls to `postStratify()` accumulates the necessary information for standard error computations.

```
pop.ctband <- data.frame(CTBAND=1:9,
        Freq=c(515672, 547548, 351599, 291425,
                        266257, 147851, 87767, 9190, 19670))
pop.tenure <- data.frame(TENURE=1:4,
        Freq=c(1459205,493237, 128189, 156348))
frs.raked <- rake(frs.des, sample=list(~CTBAND, ~TENURE),
        population=list(pop.ctband, pop.tenure))
svymean(~HHINC, frs.raked)
svymean(~HHINC, subset(frs.raked, DEPCHLDH>0))
svymean(~HHINC, subset(frs.raked, DEPCHLDH>0 & ADULTH==1))
```

Figure 7.2 Raking on socioeconomic variables in the Family Resources Survey

Example: Family Resources Survey. In an example in Chapter 5 population data on council tax band and housing tenure in Scotland were used to fit a linear regression model to household income and improve the estimates of mean weekly income. Raking gives another way to use this information, and allows for estimation of mean weekly income in subpopulations as well as for the whole population. Recall that in this example the weights have already been adjusted by raking before the data were provided, so that performing raking in R will not change the estimates, but will reduce the standard errors to the correct value.

The inputs to the `rake()` function are similar to those for `postStratify()`, except that a list of tables for each margin is specified rather than a single table. Figure 7.2 has code for raking the survey design object defined in Figure 5.7. In this case we specify the population tables as two data frames. Each data frame has a variable with the same name as a raking variable in the survey design object and a variable `Freq` with population frequencies for the levels of this variable. A list of two formulas specifies the names of the raking variables and a list of the two data frames (in the same order) specifies the population information. Estimating mean household income for all households using the raked design object gives £483 with standard error £7.5, the same as for the regression estimator and a substantial increase in precision over the standard error of £10.6 for the unraked design. Estimating the mean household income for all households with children gives £611, with a standard error of £12.5; before raking the estimated mean was the same, but the standard error was £15.6.

The improvement in precision is almost non-existent for the smaller subpopulation of single-parent households, where the standard error of estimated mean weekly income is £8.5 before raking and £8.4 after raking. This is partly because of the size of the subpopulation, and partly because the particular raking variables used here provide more detailed information about higher income levels. For example, households with exactly two adults and two children make up about the same fraction of the population and of the sample as single-parent households, but the estimated mean weekly income for these households is £714 with standard error £22.3 before raking and £714 with standard error £19.7 after raking, a larger gain in precision.

The benefit of raking decreases for smaller subpopulations because the full population information is less relevant to these smaller subgroups. In general, using population auxiliary information does not provide much extra precision in subpopulations unless the population information is specific to the subpopulations being analyzed. Similar conclusions hold for the impact of raking on estimation of regression coefficients, as is illustrated in the next section. Chapter 8 looks at some examples where there is detailed auxiliary information specific to a particular regression model and useful gains in precision are realized.

In addition to allowing estimation of mean income in subpopulations, raking also allows the auxiliary information to be used in estimating other statistics, such as quantiles with `svyquantile()`. In the original FRS design the estimated median weekly income was £355 with a 95% confidence interval £338–£372. After raking the 95% confidence interval is narrower: £341–£369.

7.4 GENERALIZED RAKING, GREG ESTIMATION, AND CALIBRATION

There are two related ways to view what is happening in post-stratification that allow extensions to continuous auxiliary variables and to a wide variety of other problems. As in section 7.2, post-stratification can be seen as making the smallest possible changes to the weights that result in the estimated population totals matching the known totals. This view of post-stratification leads to *calibration* estimators (Deville et al. [41], Deville & Särndal [40]). An alternative is to note that the estimated population total after post-stratification is a regression estimator, for a working regression model that uses indicator variables for each post-stratum as predictors (see Exercise 7.5). This view leads to *generalized regression* or *GREG* estimators (eg Särndal et al. [152]; Lehtonen and Veijanen[92]; Wu and Sitter[190]; Rao et al. [133])

A thorough review of calibration is given by Särndal [149], who distinguishes between "calibration thinking" and "regression thinking" in constructing estimators using auxiliary information. "Regression thinking" is useful in understanding why large increases in precision are possible — it is surprising to many biostatisticians that small changes in sampling weights can increase precision dramatically, but it is easier to understand that a good regression model can reduce the unexplained variation. "Calibration thinking" often leads to simpler formulas and simpler software implementation, since an explicit model is not needed.

Given auxiliary variables X_i whose population totals T_X are known, the regression estimator of the population total was constructed in section 5.2.2. A regression model is fitted to the sample to obtain regression coefficients β and the estimated population total of Y is the population total of the fitted values

$$\hat{T}_{\text{reg}} = \sum_{i=1}^{N} x_i \hat{\beta}_i = \left(\sum_{i=1}^{N} X_i \right) \hat{\beta} = T_X \hat{\beta}. \tag{7.2}$$

The parameter estimates $\hat{\beta}$ from weighted least squares estimation can be written as a weighted sum of the sampled Y_i, with weights depending on X_i and on the

population totals of X. This implies that $T_X\hat{\beta}$ can also be written as a weighted sum of Y. That, is

$$\hat{T}_Y^{(\text{reg})} = \sum_{i=1}^{n} \frac{g_i}{\pi_i} Y_i$$

for *calibration weights* g_i that do not depend on Y. This is the *calibration estimator* of the population total.

Because the weights do not depend on Y, they would be the same for estimating the population total of any variable, and in particular of X, so

$$\hat{T}_X^{(\text{reg})} = \sum_{i=1}^{n} \frac{g_i}{\pi_i} X_i.$$

Since a regression estimate of the population total of X that uses the known population total of X will be exactly correct, $\hat{T}_X^{(\text{reg})} = T_X$, and the calibration weights must satisfy the *calibration constraints*

$$T_X = \sum_{i=1}^{n} \frac{g_i}{\pi_i} X_i. \tag{7.3}$$

The calibration weights g_i make the estimated and known population totals agree. If the Horvitz–Thompson estimate of T_X is too small, g_i will give more weight to large values of X; if the Horvitz–Thompson estimate is too large, g_i will downweight large values of X. When X and Y are correlated, the calibration weights that give exact estimation of T_X will also give improved estimation of T_Y.

The calibration constraints in equation 7.3 do not uniquely define the weights. For example, if there is only one auxiliary variable X it is always possible to satisfy equation 7.3 by making a large change to the weight for just one observation. One way to completely specify g_i is to also require that the calibrated weights are as close as possible to the sampling weights. That is, for some distance function $d(\,,\,)$, the calibration weights are chosen to make

$$\text{weight change} = \sum_{i=1}^{n} d\left(\frac{g_i}{\pi_i}, \frac{1}{\pi_i}\right)$$

as small as possible while still satisfying the calibration constraints. The regression estimator of the population total corresponds to one choice of $d(\,,\,)$ and the ratio estimator from section 5.1.3 to a different choice. Raking corresponds to the same choice of $d(\,,\,)$ as ratio estimation. In linear regression calibration, the calibration weights g_i are a linear function of the auxiliary variables; in raking calibration the calibration weights are a multiplicative function of the auxiliary variables.

Ratio, raking, and regression estimators were already in widespread use before calibration was developed as a unifying description. The free choice of $d(\,,\,)$ also allows new estimates to be constructed. The distance function can be chosen to force upper and lower bounds for g_i, for example, specifying that $g_i > 0.5$ and

$g_i < 2$, to prevent individual observations from becoming too influential and to prevent computational problems from zero or negative weights.

The regression and calibration approaches to using auxiliary data do not always lead to the same estimates, but they do agree for a wide range of situations that arise frequently. It is also worth noting that some of the situations where regression and calibration approaches differ according to Särndal[149] can be unified by considering regressions where the outcome variable is the individual contribution to a population equation. That is, "calibration thinking" can be duplicated by combining "regression thinking" with "influence function thinking", as it has been in biostatistics. Calibration with influence functions is described in section 8.5.1 and details for the example given by Estevao and Särndal are discussed in the Appendix, in section A.5.

Standard error estimates after calibration follow similar arguments to those for post-stratification. When estimating a population total, a variable Y_i can be decomposed into a true population regression value μ_i and a residual $Y_i - \mu_i$. An unbiased estimator of variance of the estimated totals would be the Horvitz–Thompson variance estimator applied to $Y_i - \mu_i$. Since μ_i is unknown, we apply the Horvitz–Thompson estimator to $Y_i - \hat{\mu}_i$ and obtain a variance estimator that is nearly unbiased as long as the number of parameters in the regression model is not too large (Särndal et al. [152]).

$$\text{var}\left[\hat{T}_Y^{(\text{reg})}\right] = \text{var}\left[Y - \mu\right] \approx \text{var}\left[Y - \hat{\mu}\right]. \tag{7.4}$$

When estimating other statistics the residuals are taken after linearization, as described in Appendix C.1. As usual, the estimation is more straightforward with replicate weights. Once the calibration procedure is applied to each set of replicate weights, the variances then automatically incorporate the auxiliary information.

7.4.1 Calibration in R

The survey package provides calibration with the `calibrate()` function. As with `postStratify()` and `rake()`, this function takes a survey design object as an argument and returns an updated design object with adjusted weights and all the necessary information to compute standard errors. The auxiliary variables are specified using a model formula, as for post-stratification, and the population totals are specified as the column sums of the population regression design matrix (predictor matrix) corresponding to the model formula.

Linear regression calibration. Figure 7.3 uses calibration to repeat the estimates of US 2008 presidential election totals from Figure 5.8. The auxiliary variables are the votes in the 2000 election for George W. Bush and Al Gore. The `formula` argument specifies the auxiliary variables, and the `population` argument gives the population totals. In this example there are population totals for the intercept, and for the variables `BUSH` and `GORE` — the intercept is always included implicitly. In Figure 5.8 separate regression models were needed to estimate totals for Obama and McCain. Calibrating the survey design object incorporates the auxiliary information in all estimates, so the totals for both candidates can simply be estimated with `svytotal()`.

```
> cal.elect<-calibrate(srsdes, formula=~BUSH+GORE,
   population=c('(Intercept)'=3049, BUSH=data2000$BUSH,
      GORE=data2000$GORE))
> svytotal(~OBAMA+MCCAIN, cal.elect)
          total      SE
OBAMA  58233372 6577358
MCCAIN 51161922 4478711
> cal.elect2<-calibrate(srsdes, formula= ~BUSH+GORE,
   population=c('(Intercept)'=3049, BUSH=data2000$BUSH,
      GORE=data2000$GORE),
   variance=c(0,1,1))
> svytotal(~OBAMA+MCCAIN, cal.elect2)
          total      SE
OBAMA  60918323 4448316
MCCAIN 53509686 3353345
```

Figure 7.3 Calibrating US 2008 election data to 2000 totals

The second call to `calibrate()` specifies a different distance function $d(\ ,\ )$, one that is optimal when the variability in Y is proportional to the calibration variables. In this case, the vector `c(0, 1, 1)` specifies calibration weights that would be optimal if the variance of Y is proportional to $1 \times$ `BUSH` $+ 1 \times$ `OBAMA`, i.e., to the number of votes. This calibration gives very similar results to the regression estimators of totals using the working model with variance proportional to mean in Figure 5.8. The results would be identical if the coefficients from the regression models in Figure 5.8 were used in the `variance` argument.

Raking calibration. Repeating the example of raking from the Family Resources Survey (Figure 7.2) using the `calibrate()` function gives the code in Figure 7.4. In a regression model the two factor variables `CTBAND` and `TENURE` would be coded as an intercept and an indicator variable for each category except the first. The population total for the intercept is the population size, and this is concatenated with the counts for each of the auxiliary variables, dropping the first category of each. The option `calfun="raking"` requests the calibration distance function that is equivalent to raking. The estimates of mean (and median) weekly income based on the calibrated design are identical to those obtained from the raked design in Figure 7.2. The advantage of raking calibration using `calibrate()` over raking using `rake()` is that `calibrate()` can use continuous auxiliary variables and `rake()` can use only discrete variables.

Logit calibration and bounded weights. A popular calibration function in Europe gives so-called *logit calibration*. This distance function requires the user to specify upper and lower bounds on the calibration weights. If possible, the `calibrate()` will return calibration weights that satisfy the bounds. It may be impossible to satisfy both the bounds and the calibration constraints (Equation 7.3),

```
pop.size <- sum(pop.ctband$Freq)
pop.totals <- c('(Intercept)'=pop.size, pop.ctband$Freq[-1],
   pop.tenure$Freq[-1])
frs.cal <- calibrate(frs.des,
   formula=~factor(CTBAND)+factor(TENURE),
   population=pop.totals, calfun="raking")
svymean(~HHINC, frs.cal)
svymean(~HHINC, subset(frs.cal, DEPCHLDH>0))
svymean(~HHINC, subset(frs.cal, DEPCHLDH>0 & ADULTH==1))
```

Figure 7.4 Calibration (raking) on socioeconomic variables in the Family Resources Survey

in which case `calibrate()` will give an error. The `force=TRUE` option forces `calibrate()` to return a survey design in which the weights satisfy the bounds even if the calibration constraints are not met. This is useful to allow a series of simulations to be completed, and may be useful for data analysis if the calibration constraints are nearly met, which can be easily checked using `svytotal()`.

Figure 7.5 shows code for logit calibration on a cluster sample of school districts from the Academic Performance Index population. Using linear calibration with the same auxiliary variables gives calibration weights ranging from 0.4 to 1.8, with logit calibration the weights vary from slightly above the lower bound of 0.7 to slightly below the upper bound of 1.7.

Calibration using the 1999 Academic Performance Index gives a dramatic reduction in standard error for estimating the mean of the 2000 Academic Performance Index. In a regression model with Academic Performance Index as the outcome there is a substantial increase in precision for the intercept but not for the slope estimates. In this model the predictors are proportion of "English Language Learners" (`ell`), proportion of students who are new to the school (`mobility`) and proportion of teachers with only emergency qualifications (`emer`). These predictors and the outcome are quite strongly associated with the auxiliary variables, but this is not enough to give an increase in precision for comparisons across levels of the predictors. The problem of calibration to increase precision for coefficient estimates in regression models is discussed further in Chapter 8. Gains in precision are possible, but they require calibration targeted to the specific model. Calibration of large surveys carried out when the data is collected is not likely to increase precision for regression models fitted in secondary data analysis. On the other hand, the reduction in uncertainty for the intercept can (and in this example does) translate to a useful reduction in standard errors for fitted values.

Comparing calibration methods. In the absence of non-response, the choice of calibration function makes little difference to the resulting estimated totals, with the difference decreasing as sample size increases [40]. In finite samples or with non-response it is possible for the results to differ, but they typically do not. Kalton and Flores-Cervantes [69] gave an artificial example of calibration comparing a range of

```
>  clus1 <- svydesign(id=~dnum, weights=~pw, data=apiclus1,
     fpc=~fpc)
> logit_cal <- calibrate(clus1, ~stype+api99,
+   population= c( 6194, 755, 1018, 3914069),
+   calfun="logit",bounds=c(0.7,1.7))
> svymean(~api00, clus1)
        mean     SE
api00 644.17 23.542
> svymean(~api00, logit_cal)
        mean   SE
api00 665.46 3.42
> summary(svyglm(api00~ell+mobility+emer, clus1))
Call:
svyglm(api00 ~ ell + mobility + emer, clus1)
Survey design:
svydesign(id = ~dnum, weights = ~pw, data = apiclus1,
    fpc = ~fpc)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 780.4595    30.0210  25.997 3.16e-11 ***
ell          -3.2979     0.4689  -7.033 2.17e-05 ***
mobility     -1.4454     0.7343  -1.968  0.07474 .
emer         -1.8142     0.4234  -4.285  0.00129 **

> summary(svyglm(api00~ell+mobility+emer, logit_cal))
Call:
svyglm(api00 ~ ell + mobility + emer, logit_cal)
Survey design:
calibrate(clus1, ~stype + api99, population = c(6194, 755, 1018,
    3914069), calfun = "logit", bounds = c(0.7, 1.7))
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 789.1015    17.7622  44.426 9.18e-14 ***
ell          -3.2425     0.4803  -6.751 3.15e-05 ***
mobility     -1.5140     0.6436  -2.352 0.038318 *
emer         -1.7793     0.3824  -4.653 0.000702 ***
> predict(m0, newdata=data.frame(ell=5,mobility=10,emer=10))
    link     SE
1 731.37 26.402
> predict(m1, newdata=data.frame(ell=5,mobility=10,emer=10))
    link    SE
1 739.96 15.02
```

Figure 7.5 Logit calibration on cluster sample of school districts from the Academic Performance Index population

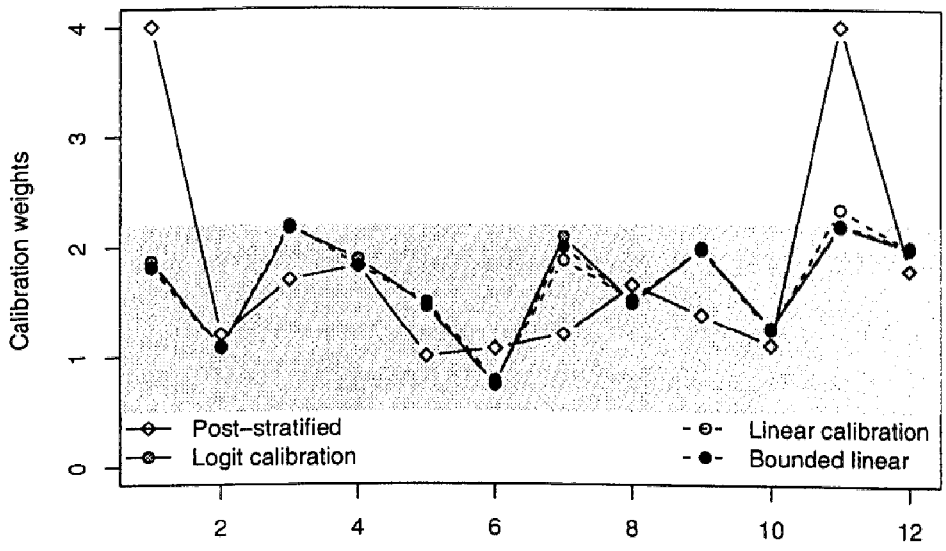

Figure 7.6 Impact of different calibration metrics. Artificial data from Kalton and Flores-Cervantes [69]. The gray rectangle shows the calibration bounds (0.5, 2.2)

calibration methods. Code to reproduce their analyses is in the `tests` subdirectory of the survey package, in the file `kalton.R` and Figure 7.6 shows some of the resulting sets of calibration weights. The data form a 3×4 table and there are three sets of calibration weights using the margins of the table, and the set resulting from post-stratification on the cells of the table. The three sets of weights based on the margins of the table use linear calibration, logit calibration, and bounded linear calibration, with both logit and bounded linear calibration having a lower bound of 0.5 and an upper bound of 2.2, as indicated by the shaded region. The two bounded techniques give weights inside the shaded region; linear calibration gives one weight slightly outside the region at 2.34.

It is clear from the graph that the choice of auxiliary variables is much more important than the choice of calibration function. The three sets of weights using the margins of the table are very similar, and are quite different from the weights using all the cell counts in the table. Adding additional calibration variables will always give increased precision (or reduced non-response bias) with sufficiently large sample sizes, but for any given sample size there will be a point where the added uncertainty from estimating parameters in the calibration model outweighs the gains. Judkins et al. [68] describe one approach to assessing whether added auxiliary variables are helpful.

Cluster-level weights. When the last stage of a probability design involves cluster sampling, such as sampling all individuals in a household, the sampling weights for individuals in the same cluster are necessarily identical. Especially for official statistics, it may be desirable for the calibrated weights for individuals in the same cluster to also be identical. This provides internal consistency between estimates, though at a small cost in efficiency. For example, in collecting data on

```
> cal2 <- calibrate(clus2_design, ~stype,
    pop=c(4421+755+1018,755,1018),
    aggregate.stage=1)
> svytotal(~enroll, cal2,na.rm=TRUE)
         total     SE
enroll 3084777 246321
> svytotal(~enroll, ps_design,na.rm=TRUE)
         total     SE
enroll 3074076 292584
> range(weights(cal)/weights(clus2_design))
[1] 1.075826 1.265473
> range(weights(cal2)/weights(clus2_design))
[1] 0.6418424 1.6764125
```

Figure 7.7 Forcing calibration weights to be constant within a school district

a sample of births, where every infant must have exactly one mother, it would be undesirable for estimated subpopulation totals of mothers to add up to more than the estimated total number of infants.

The `aggregate.stage` argument to `calibrate()` specifies the stage of sampling at which calibration weights must be constant within clusters. This argument is available only for survey design objects that include the sampling design. For survey design objects based on replicate weights the `aggregate.index` argument specifies a vector of cluster identifiers so that calibration weights will be made equal within the same cluster.

As an example, consider the two-stage cluster sample from Academic Performance Index population. The first stage of the design samples school districts and the second stage samples up to five schools within each district. Sampling probabilities, and thus sampling weights, are the same for schools within the same school district. When the sample was post-stratified on school type in Figure 7.1 the calibration weights were not the same for schools within the same cluster, varying up to 20% between school type. In Figure 7.7 the weights are forced to be constant within a school district, that is, within each stage-1 sampling unit. There is a slight cost in precision, with the standard error for the estimated total enrollment increasing by about 20%. This loss of precision occurs because the weights become more variable across school districts to compensate for the added constraints. The calibration weights in the simple post-stratified design range from 1.08 to 1.27, with constant weights within school district the range is from 0.64 to 1.68.

It is also possible to do the converse: to use auxiliary information where population totals are available only for sampled clusters at some stage of sampling, not for the whole population. This is done by supplying a list of population totals for the sampled clusters, with the names on the list matching the cluster identifiers, and giving the `stage` argument to `calibrate()` to indicate which stage of sampling the totals belong to. An example using the same two-stage sample from the Academic

Performance Index population is given in the `tests` subdirectory of the survey package, in the file `caleg.R`.

7.5 BASU'S ELEPHANTS

Basu [3, 97] gave an example that was intended to show unreasonable behavior of design-based inference, but that can be interpreted in terms of poor use of auxiliary information. In his story a circus owner had 50 elephants and wanted an estimate of their total weight based on weighing only one elephant. In the absence of any additional information a sensible approach would be to take a simple random sample of one elephant and weigh it (E_1), and then multiply by the sampling weight.

$$\hat{T}_{\text{srs}} = \frac{1}{\pi_1} E_1 = 50 \times E_1.$$

This is a valid design-based estimate, although no unbiased estimate of the standard error is available from a sample of size one.

The circus owner knows that five years previously, when all the elephants were last weighed, a particular mid-sized elephant, Sambo, had very nearly the average weight. A reasonable model-based estimate of the total weight of the elephants now would be

$$\hat{T}_{\text{model}} = 50 \times E_{\text{Sambo}}.$$

This is not a valid design-based estimate; no unbiased design-based estimate can be obtained when the sampling probabilities are zero for 49 of 50 elephants.

Basu imagines a compromise design worked out between the circus owner and the circus statistician, in which Sambo is sampled with very high probability ($\pi_{\text{Sambo}} = 0.99$) and the remaining probability is divided up among the other elephants ($\pi_i = 1/5000$). When the sampling is performed, Sambo is in fact chosen. The circus owner expects the estimate to be $50 \times E_{\text{Sambo}}$, but the statistician points out that the Horvitz–Thompson estimator is

$$\hat{T}_{\text{HT}} = \frac{1}{\pi_i} E_i = \frac{100}{99} \times E_{\text{Sambo}}.$$

This estimate is clearly silly. Worse still, if one of the other elephants, e.g., the largest elephant, Jumbo, had been weighed, the Horvitz–Thompson estimator would be

$$\hat{T}_{\text{HT}} = \frac{1}{\pi_i} E_i = 5000 \times E_{\text{Jumbo}}.$$

The Horvitz–Thompson estimator is exactly unbiased, but this is a property defined by averaging over possible realizations of the sampling design. For any single realization of the sampling design the result will be clearly unreasonable.

This example is somewhat embarrassing for design-based inference, so it is worth considering why such poor results are obtained. To some extent the unreasonable result is the fault of the small sample. In large samples we can be confident that the

value of an estimate will be close to its expected value, so it is useful to know that the expected value is equal to the true population value. In small samples, where an estimate need not be close to the expected value, unbiasedness is not sufficient to ensure reasonable behaviour. However, the Horvitz–Thompson estimate is supposed to be useful even in small samples, so something is still wrong.

The first question is whether the "compromise" design is actually a sensible use of the prior knowledge if the Horvitz–Thompson estimator is going to be used. The second question is whether the Horvitz–Thompson estimator is appropriate, given the auxiliary information that is available. The answer to both questions is "No."

Using auxiliary information in design. In choosing which elephant to weigh it is helpful to consider a situation where the sample size is slightly larger. If the circus were weighing three elephants rather than one, they could consider stratified sampling. Equation 2.6 in section 2.6 gives the optimal allocation: proportional to the number in each stratum and to the standard deviation in the stratum. If the elephants were divided into small, medium, and large by eye (or by previous weight), taking one elephant at random from each stratum would make sense. This leads to reasonable estimates and makes use of available information. A stratified sampling approach would lead to sampling probabilities π_i that did not vary much between elephants.

A similar approach, ranked-set sampling [72, 107, 123], is used in some ecological applications. Three samples of three elephants would be taken at random, and the elephants in each sample ranked by size. The smallest elephant from the first sample, the middle elephant from the second sample, and the largest elephant from the third sample are weighed. The ranking procedure has a similar effect to stratification, but ranking is often easier — lining up all the elephants by size would be more work than judging which of three is largest. Under ranked-set sampling the sampling weight π_i is the same for every unit in the population.

Section 3.3 considered sampling clusters proportional to size, or more generally, sampling observations proportional to some auxiliary variable available for the whole population. If the sampling probabilities are roughly proportional to the variable being analyzed, the variance of the estimated total will be small. In applying this principle to the elephants the circus owner could use either the weights recorded at the previous weighing, or if these are lost, an approximate measure of elephant volume as height × length × width. Under this PPS sampling scheme the sampling probability would be highest for the largest elephant. The variation between elephants in sampling probabilities would still be much smaller than for Basu's design, since the record weight for an elephant is only about twice the typical adult weight.

Based on the standard techniques used for designing complex samples it appears that if the Horvitz–Thompson estimator were to be used in analysis, a good design would have much less variation in sampling probabilities than Basu's design and that sampling larger elephants with modestly higher probability would be helpful.

Using auxiliary information in analysis. A better way to use auxiliary information about the elephants would be by some form of calibration. For example, calibrating the estimated population size $N = 1/\pi_i$ to the known population size

$N = 50$ gives calibrated weights $g_i/\pi_i = 50$. The resulting model-assisted estimate of the total is

$$\hat{T}_{\text{cal:N}} = \frac{g_i}{\pi_i} \times E_i = 50 \times E_i$$

whichever elephant is sampled, the estimate that the circus owner wanted to use. This estimate was proposed by Hájek [57] in the discussion of Basu's original essay, based on the fact that dividing by the estimated (rather than known) population size often gives improved estimates of the population mean; it is also known as the Hájek ratio estimator of the total.

The estimate $\hat{T}_{\text{cal:N}}$ is not unbiased. It would be approximately unbiased for a large sample size, and in that sense is a design-based estimator, but 1 is not a large sample size. On the other hand, for almost any imaginable population configuration it will be better than $\hat{T}_{\text{HT}}$. Using this design and estimator will give more accurate estimates than taking a simple random sample and using $\hat{T}_{\text{srs}}$. Exercise 7.2 examines this issue by simulation.

It is possible to do better using the information from the previous weighing. If X_i is the weight of elephant i five years ago and T_X is the total of X, the weights can be calibrated on X giving the calibration constraints

$$\frac{g_i}{\pi_i} X_i = T_X.$$

Two solutions to the calibration constraints are

$$\frac{g_i^{(\text{ratio})}}{\pi_i} = \frac{T_X}{X_i}$$

and

$$\frac{g_i^{(\text{diff})}}{\pi_i} = \frac{T_X - 50X_i}{X_i} + 50$$

corrresponding to the ratio working model

$$E_i = \alpha X_i + \epsilon_i$$

and the difference working model

$$E_i = \alpha + X_i + \epsilon_i.$$

The resulting estimates of the total are

$$\hat{T}_{\text{ratio:X}} = \frac{T_X}{X_i} E_i = \frac{E_i}{X_i} T_X,$$

where the previous total is multiplied by the relative increase in weight for the measured elephant, and

$$\hat{T}_{\text{diff:X}} = T_X + 50 \times (E_i - X_i)$$

where 50 times the absolute increase in weight for the measured elephant is added to the previous total.

These estimators are likely to be more precise than the Horvitz–Thompson estimator because they only need to estimate the relatively small change in weight since the previous weighing. Using these estimators it is no longer desirable to sample Sambo with higher probability, in contrast to $\hat{T}_{\text{cal:N}}$, because the auxiliary information is already being used efficiently.

The fact that only one elephant is weighed limits the choice of working models to the simple ratio and difference models. If the circus owner had wanted to weigh five elephants out of 250 it would be possible to do better using a working model with more parameters, such as

$$E_i = \alpha + \beta X_i + \epsilon_i.$$

This analysis shows that the circus statistician's failure in Basu's example was not adherence to design-based inference but ignorance of calibration estimators as the appropriate way to use auxiliary information. It is true that some of the approaches described here use more information than just the fact that

$$T \approx 50 \times E_{\text{Sambo}}.$$

If the circus owner knew that this was a sufficiently accurate approximation it would obviously be sensible just to weigh Sambo, but it is hard to imagine being sure of this without knowing anything else. For example, it is hard to imagine how the owner would know that the weight gains of the elephants over the past five years add up to 50 times Sambo's weight gain without also knowing that, e.g., the weight gains have been similar for each elephant.

The point of this example is not to argue that design-based estimates are to be preferred to model-based estimates, but to show that when a design-based estimate gives a clearly inappropriate estimate it is probably because it is not a good design-based estimate.

7.6 SELECTING AUXILIARY VARIABLES FOR NON-RESPONSE

Post-stratification, raking, and calibration are widely used to reduce the bias from *unit non-response*, people who cannot be contacted or refuse to participate in surveys. This problem is increasing over time, especially for telephone surveys; the University of Michigan's Survey of Consumer Attitudes found response rates decreasing by about 1% per year from 1979 to 2003 (Curtin et al.
citecurtin-nonresponse), and the response rate in the Behavioral Risk Factor Surveillance System declined from about 70% in 1991 to about 50% in 2001. Reweighting for non-response is not a purely design-based method; it relies implicitly on models for the missing data.

In the simplest case, post-stratification, the model is that non-response is independent of the outcome variable within groups defined by the auxiliary variables. For example, suppose the probability of responding to a telephone survey about health

insurance was higher for landline than cellphone users but within each group the probability of responding was independent of whether the individual had health insurance. An estimate using sampling weights based on the design would give too little weight to cellphone users (who tend to be younger) and so would be biased. If the telephone companies could be persuaded to give population numbers of landline and cellphone numbers, the analysis could be post-stratified by telephone type to give a valid estimate. This example also illustrates some of the limits of non-response adjustment: no reweighting, however sophisticated, will allow a telephone survey to give information about people without telephone service.

One way to look at the effect of post-stratification is that the correct sampling probability for a homogenous group of people is not the intended probability π_i but the achieved probability. If we try to sample 100 women aged 25–35 from a population of 10,000 and only 76 of them respond, the actual sampling fraction is not $100/10,000$ but $76/10,000$. This correction is exactly what post-stratification does

$$\frac{\pi_i}{g_i} = \pi_i \times \Pr[\text{response rate}].$$

There are two relevant ways that a group can be homogenous. If the non-response probability is homogenous within the group, as above, then post-stratification will give correct sampling weights, in the sense that the population total for any variable is correctly estimated. On the other hand, if a particular outcome variable is homogenous within the group, the population total for that outcome variable will be correctly estimated even if response is not homogenous: the relative weighting for individuals within the group will be wrong, but as the outcome is the same this incorrect relative weighting does not matter. More generally, if a regression model using the auxiliary variables can explain most of the variation in an outcome, the residual variation will be small and bias in analyzing it will be small in absolute terms even if large in relative terms. These two possibilities are analogous to the two constructions of calibration by "regression thinking" and "calibration thinking" in section 7.4.1.

For large-scale surveys there are usually only a small number of possible auxiliary variables and the goal must be to be produce correct results for all analyses, not just for a few chosen variables, so homogeneity of response is much more important. Of course, there is no realistic prospect of achieving true homogeneity of response using the few auxiliary variables typically available, but the estimates after post-stratification are probably less biased than those before post-stratification.

Keeter et al. [73] published results from an encouraging experiment that administered the same questionnaire in two telephone surveys, one of which made very extensive efforts to reduce non-response. The non-response rates were 36% and 60% for the two surveys. Even before any form of reweighting, the differences in political and social attitudes between responders to the two surveys wer much smaller than the differences in demographic variables. This shows that even quite high levels of non-response in an otherwise well-conducted survey may still give reasonable results. It also suggests that post-stratification or calibration should work well, since the demographic variables most likely to be used for reweighting the sample appeared more sensitive to response rates than the outcome variables being studied.

7.6.1 Direct standardization

Direct standardization of rates is the term used in epidemiology and demography for reweighting a sample from one population so that the distribution of variables such as age group and sex matches a different population. Direct standardization is used either to extrapolate to an estimate of the rate in the target population or to compare the extrapolated rate to the observed rate in the target population. For example, comparing the outcomes of neonatal care in different hospitals is difficult because the hospitals may have different numbers of high-risk, low birth weight infants. Direct standardization for birthweight allows a comparison based on the same distribution of birth weight in different hospitals.

Post-stratification for non-response and direct standardization are mathematically equivalent, but post-stratification is usually done with the intention of getting an improved estimate in the target population, and direct standardization is more often done to compare the known rates in the target population with rates extrapolated from the source sample. That is, post-stratification for non-response relies on the assumption that category-specific rates will be the same in the sample and target population, where direct standardization is often used to evaluate whether or not the category-specific rates are the same.

7.6.2 Standard error estimation

In theory, the same approaches to standard error estimation that apply to post-stratification and calibration for precision also apply to post-stratification and calibration for non-response. When conducting secondary analysis of large-scale surveys it is often not possible to use the correct standard error estimates, because the information needed to compute the residuals is not published. In these cases, rather than computing the standard errors from residuals as in equation 7.1 and 7.4, the standard errors are computed as if the calibrated weights g_i/π_i are simply the sampling weights. This approximation, like the single-stage approximation for multistage sampling, is typically conservative.

When survey data are published with replicate weights, as in the California Health Interview Survey, it is possible to produce the correct standard errors by ensuring that each set of replicate weights is post-stratified or calibrated appropriately. Alternatively, if the non-response adjustments to the weights are sufficiently straightforward it may be possible to reproduce them at the time of analysis, as in the example of the Family Resources Survey in section 7.3.

EXERCISES

7.1 Using the Washington State Crime population, take a stratified random sample of five police districts from King County and five counties from the rest of the state.

a) Calibrate the sample using stratum and population as the auxiliary variables. Estimate the number of murders and number of burglaries in the state using the calibrated and uncalibrated sample.

b) Convert the original survey design object to use jackknife replicate weights (with `as.svrepdesign()`). Calibrate the replicate-weight design using the same auxiliary variables and estimate the number of burglaries and of murders in the state.

c) Calibrate the sample using the population and the number of burglaries in the previous year as auxiliary variables, and estimate the number of burglaries and murders in the state.

d) Estimate the ratio of violent crimes to property crimes in the state, using the uncalibrated sample and the sample calibrated on population and number of burglaries.

7.2 Write an R function that accepts a set of 50 elephant weights and simulates repeatedly choosing a single elephant and computing the Horvitz–Thompson and ratio estimators of the total weight, reporting the mean and variance over the repeated simulations. Explore the behavior for several sets of elephants weights. Verify that the Horvitz–Thompson estimator is always unbiased, but that it is usually further from the truth than the ratio estimator.

7.3 ⋆ Write an R function that accepts a set of 50 elephant weights and performs the ranked-set sampling procedure on page 150 to choose three of them. By simulation, compare the bias and variance of the estimated total from ranked-set sample to estimated totals from a simple random sample of three elephants.

7.4 Estimate the proportions of people in California with normal weight, overweight, and obesity using the BRFSS 2007 data (California is `X_STATE` = 6, and BMI categories are `X_BMI4CAT`). Post-stratify the California data to have the same age and sex distribution (`X_AGE_G` and `X_SEXG_`) as the data for Florida (`X_STATE` = 12) and compute the directly standardized estimates of proportions for the BMI categories. Compare the raw and standardized estimates based on California data to estimates from the data for Florida to see if the differences in BMI between the states are explained by differences in age distribution. [You will need to load the data subset for California directly into memory, as `postStratify()` does not currently support database-backed designs.]

7.5 ⋆ Consider a categorical post-stratification variable with K categories having population counts $N_1, N_2, \ldots, N_k$. Suppose we are interested in estimating the total of a variable Y

a) Show that the post-stratified estimate is

$$\hat{T}_{\text{ps}} = \sum_{k=1}^{K} N_k \hat{\mu}_k$$

where $\hat{\mu}_k$ is the estimated mean of Y in group k before post-stratification.

b) Show that the regression estimate from a model with indicator variables for each group is also

$$\hat{T}_{\text{reg}} = \sum_{k=1}^{K} N_k \hat{\mu}_k .$$