ratio estmation

Peter Lugtig

10 oktober, 2021

For the next exercise, you will need the following libraries:

```
require(sampling)
require(survey)
require(dplyr)
require(ggplot2)
```

Introduction

In this short exercise, you will prctice with specifying a ratio estimator in R, and hopefully will learn more about why ratio estimators are so useful in specific circumstances.

We will use a new dataset for this exercise that centers on my favourite drink in the world: coffee. Imagine a situation where Utrecht University would like to keep track of how many coffees are being drunk on the buildings on campus in a month. Perhaps they want to know how much coffee is being taken from the coffee machines, perhaps they simply need to know how many coffees are being drunk to prepare a new coffee contract with a supplier. We will first concentrate on estimating the mean number of coffees per machine, but can also estimate the total directly.

There are 1000 machines on campus, some of them used more often than others, but lets imagine the university is not willing to use any information on for example the type of building to inform a potentially stratified design. Instead, they opt to equip a SRS of 100 machines with a device that counts the number of coffees drunk.

Let's sample some data. First the population data (pretend for now you don't know about these data, like in real life.)

```
set.seed(11)
coffees <- round(rpois(1000,350)+rnorm(100,0,sd=150))
coffees[coffees<0] <- 0
# and energy use
energy <- 0.072*coffees+rnorm(n=1000,mean=0,sd=1)
energy[energy<0] <- 0
coffeedata <- as.data.frame(cbind(coffees,energy))
names(coffeedata) <- c("n","energy")
```

And sample some coffee data from the population:

```
set.seed(11)
coffeedata$srs <- srswor(100,nrow(coffeedata))
coffee <- subset(coffeedata, srs==1)
```

The first task would be to specify the survey design object

```
coffee$fpc <- 1000
srsdesign <-svydesign(ids=~1,fpc=~fpc,data=coffee)
```

and estimate the mean number of coffees using normal SRS estimation (as covered before)

```
svymean(~n, design=srsdesign) # se =15.415)
# or estimate the confidence interval directly
confint(svymean(~n, design=srsdesign)) #[312.237;372.663]
```

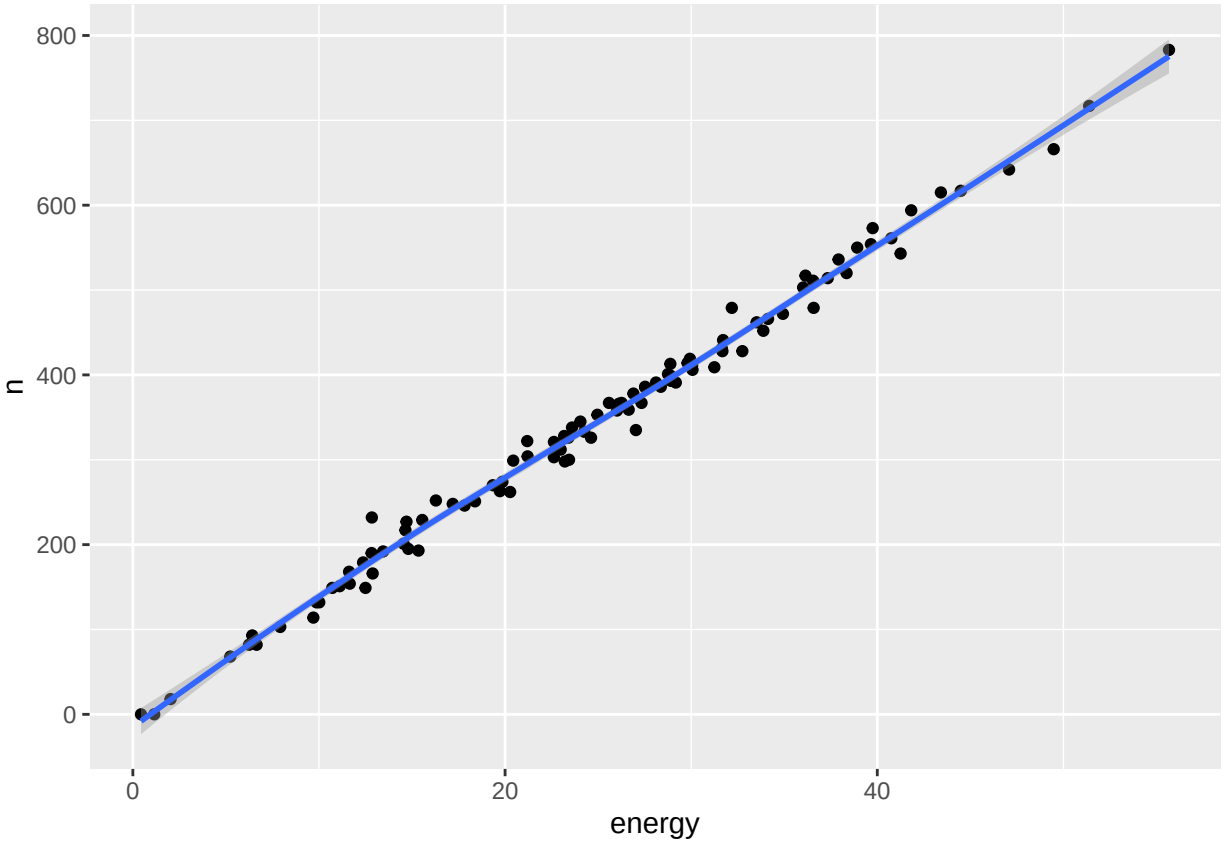
As the width of the confidence interval is more than 60 cups of coffee, the university hires a survey researcher to design a more efficient sample and estimate the total number of coffees drunk more efficiently.
The researcher finds out that the although the university does not keep track of the number of coffees made per machine, it does track the energy consumption per machine. Can this information be used? The dataset (including info from the SRS sample) looks as follows.

```
# First, delete the dependent variable (coffee) in our population (apart from the sample), r
coffeedata2 <-coffeedata
coffeedata2$n <- NA
```

Question 1:

Now what? Can we do better? Perhaps! But we have to model the relation between energy use and the number of coffees. Lets make a plot (using just the survey data), and decide whether we can use a ratio estimator. Do you think a ratio estimator is here a good idea judging the plot?

```
ggplot(coffee,aes(y=n, x=energy))+
        geom_point()+
        geom_smooth()
```



Queston 2:

Run the code below to first estimate the ratio between coffees/energy, and then use that ratio to predict the mean number of coffees. What happens to the precision when we use a ratio estimator?

```
# first limit ourselves to the observed srs data
srsdesign <- svydesign(ids=~1, data=coffee)
```

```
## Warning in svydesign.default(ids = ~1, data = coffee): No weights or
## probabilities supplied, assuming equal probability
```

```
# estimate the ratio
ratio <- svyratio(~n,~energy, design=srsdesign)
# what is this ratio?
print(ratio) # with 1 KwH 13.81 coffees are made on average
```

Question 3:

In our dataset, the standard error becomes much smaller because of ratio estimation. Why then not always use a ratio-estimator? First, we need to have a dependent variable that is of ratio measurement level. The ratio estimator will not work when we for example try to estimate weight or height in the "boys" dataset, as boys never have a height or weight of zero. Secondly, we need to have a covariate on our sampling frame

(like "energy use") that is ideally a perfectly linear predictor of our dependent variable. The relationship between height and weight is not perfectly linear. The relationship between energy use and the number of coffees from a machine is however. Or is it not?

Investigate whether the ratio estimator you used in question is actually unbiased. What is the prediction of the total number of coffees drunk, and how much bias is there in this estimate?

Question 4:

Can you think of a reason why there is some bias in the ratio estimator for the total number of coffees drunk at the UU? Think about an answer first, and then run the code below. What do you find?

```
 # why?
summary(glm(n~energy, data=coffee))
```

A possible solution to the fact that the ratio estimator can in some cases be biased, is to extend the model with more covariates. In such a case, we move from using a "model-assisted" estimator like ratio estimation to a truly "model-based" estimator. In order to understand how model-based estimators work, look at the exercise "Regression estimation"

Question 5 (if time left):

The ratio estimator is in practice used quite often in cluster samples. Can you think of why?
(see .Rmd file for answers)

– end of document –