

## Regression Estimation

Recall that the method of ratio estimation is appropriate when the response variable  $y$  is linearly related to some auxiliary variable  $x$ , and the value of  $y = 0$  when  $x = 0$ . Sometimes, there is a linear relationship between the response  $y$  and an auxiliary variable  $x$  such that when  $x = 0$ , the value of  $y$  does not equal zero. In such cases, the method of regression estimation can be employed. Regression estimation requires population information on  $x$ , either the population mean  $\mu_x$  or total  $\tau_x$ .

- For example, if  $y$  and  $x$  are both positive variables but are negatively associated, such as weight of a car ( $x$ ) and mpg ( $y$ ), then the relationship could be linear, but does not go through the origin.
- As another example, suppose  $y =$  the sale value of a home in Missoula county, and  $x =$  the appraisal value of the home in the previous tax year. Although we certainly expect  $y$  &  $x$  to be related (perhaps linearly), we have no information about the relationship when  $x$  is near zero, and no reason to believe the relationship, if linear, should be forced to go through the origin.
- Regression estimation can be applied to more situations than just simple linear regression. It can accommodate more than one auxiliary variable or higher order relationships such as quadratic ones. We will only consider simple linear regression estimation here; extensions to multiple linear regression models are straightforward.

### The Linear Regression Estimator

Suppose  $y_1, \dots, y_N$  comprise a population of values such that:  $y_i = A + Bx_i, i = 1, \dots, N$ , where:

$$\begin{aligned}y_i &= \text{response variable on the } i^{\text{th}} \text{ unit,} \\x_i &= \text{auxiliary variable on the } i^{\text{th}} \text{ unit,} \\ \mu_y, \tau_y &= \text{mean and total of the y-values (book just uses } \mu, \tau), \\ \mu_x, \tau_x &= \text{mean and total of the x-values.}\end{aligned}$$

The population total and mean for the y-values are given by:

$$\tau_y =$$

$$\mu_y =$$

- The regression estimator for  $\mu_y$  is:  $\hat{\mu}_L = a + b\mu_x = \bar{y} - b\bar{x} + b\mu_x = \bar{y} + b(\mu_x - \bar{x})$ , where via the method of least squares:

$$b = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \text{ (slope), } a = \bar{y} - b\bar{x} \text{ (intercept),}$$

the usual least squares estimators of the slope and intercept.

- Via the Delta method, the variance of  $\hat{\mu}_L$  is approximated by:

$$\text{Var}(\hat{\mu}_L) \approx \frac{(N-n)}{Nn(N-1)} \sum_{i=1}^N (y_i - A - Bx_i)^2, \text{ where:}$$

$$B = \frac{\sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y)}{\sum_{i=1}^N (x_i - \mu_x)^2}, \quad A = \mu_y - B\mu_x,$$

the population slope and intercept.

- Since  $A$  &  $B$  are unknown population parameters, the estimated variance of  $\hat{\mu}_L$  is given by:

$$\widehat{\text{Var}}(\hat{\mu}_L) = \frac{(N-n)}{Nn(n-2)} \sum_{i=1}^n (y_i - a - bx_i)^2.$$

- A normal-based confidence interval for  $\mu_y$  can be obtained as usual, by appealing to the finite population CLT, although the resulting CI's have been shown to be somewhat conservative (Scott & Wu, 1981).

#### Regression Estimation of the Total $\tau_y$ :

$$\begin{aligned} \hat{\tau}_L &= N\hat{\mu}_L = N(a + b\mu_x) = N(\bar{y} - b\bar{x} + b\mu_x) = N\bar{y} + b(N\mu_x - N\bar{x}) = N\bar{y} + b(\tau_x - N\bar{x}), \\ \text{Var}(\hat{\tau}_L) &= N^2 \cdot \text{Var}(\hat{\mu}_L), \quad \widehat{\text{Var}}(\hat{\tau}_L) = N^2 \cdot \widehat{\text{Var}}(\hat{\mu}_L). \end{aligned}$$

#### Ratio and Regression Estimation in R

To illustrate the use of R in performing ratio and regression estimation, consider the following example.

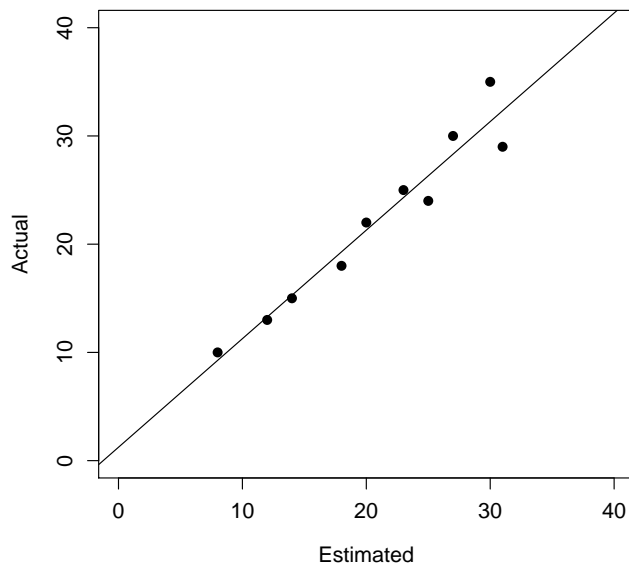
An investigator wishes to estimate the total number of trees on a 250-acre plantation. She divides the plantation into 1000 1/4-acre plots. She has aerial photographs from which she

can easily estimate the total number of trees on each plot. She counts the actual number of trees on an SRS of 10 plots in order to calibrate her estimates from the photographs. Based on the aerial photographs, she estimates there are a total of 23,100 trees on the whole plantation or 23.1 per plot. For the SRS of ten plots, she finds the following:

Plot	1	2	3	4	5	6	7	8	9	10
Actual # Trees	25	15	22	24	13	18	35	30	10	29
Photo Estimate	23	14	20	25	12	18	30	27	8	31

A scatterplot of the data is below. Does it appear that a ratio estimate is appropriate or should a regression estimate be used (or neither)?

```
> x <- c(23,14,20,25,12,18,30,27,8,31)
> y <- c(25,15,22,24,13,18,35,30,10,29)
> plot(x,y,xlim=c(0,40),ylim=c(0,40),pch=16,xlab="Estimated",ylab="Actual",
      cex=1.2,cex.lab=1.2,cex.axis=1.2)
> reg <- lsfit(x,y) # least-squares fit
> abline(reg) # add least squares line to plot
```



Treating the photo estimate as an auxiliary variable, suppose we first use a ratio estimate to estimate the total number of trees  $\tau_y$ . First, estimate  $R$ , the ratio of the total actual number of trees to the photo estimate. The SE of  $r$  is also estimated, although we really are not interested in  $R$  itself.

```

> r <- mean(y)/mean(x)
> r
[1] 1.0625
> sr2 <- (1/9)*sum((y-r*x)^2)
> sr2
[1] 4.225694
> sqrt((990/1000)*sr2/(10*mean(x)^2)) # if the mean of x were unknown
[1] 0.03109591

```

$$\left( s_r^2 = \frac{1}{n-1} \sum_{i=1}^n (y_i - rx_i)^2 \right)$$

$$\left( SE(r) = \sqrt{\left( \frac{N-n}{N} \right) \frac{1}{\bar{x}^2} \cdot \frac{s_r^2}{n}} \right)$$

```

> se.r <- sqrt((990/1000)*sr2/(10*23.1^2)) # Use this since mux = 23.1 is known
> se.r
[1] 0.02799978

```

$$\left( SE(r) = \sqrt{\left( \frac{N-n}{N} \right) \frac{1}{\mu_x^2} \cdot \frac{s_r^2}{n}} \right)$$

To compute a 99% confidence interval for  $r$ :

```

> c(r - qt(.995,9)*se.r, r + qt(.995,9)*se.r)
[1] 0.9715053 1.1534947

```

To estimate  $\mu_y$ , the average number of trees per plot:

```

> muhat <- r*(23.1)
> muhat
[1] 24.54375
> se.muhat <- sqrt((990/1000)*sr2/10)
> se.muhat
[1] 0.646795

```

$$(\hat{\mu}_y = r \cdot \mu_x)$$

$$\left( SE(\hat{\mu}_y) = \sqrt{\frac{N-n}{N} \cdot \frac{s_r^2}{n}} \right)$$

To estimate  $\tau_y$ , the total number of trees:

```

> tauhat <- r*23100
> tauhat
[1] 24543.75
> se.tauhat <- 1000*se.muhat
> se.tauhat
[1] 646.795

```

$$(\hat{\tau}_y = N \cdot \hat{\mu}_y)$$

$$(SE(\hat{\tau}_y) = N \cdot SE(\hat{\mu}_y))$$

A 99% CI for  $\tau_y$ :

```
> c(tauhat - qt(.995,9)*se.tauhat,tauhat + qt(.995,9)*se.tauhat)
[1] 22441.77 26645.73
```

Now, consider obtaining a regression estimate of  $\mu_y$  &  $\tau_y$ . First, perform a linear regression of  $y$  on  $x$ :

```
> reg <- lsfit(x,y)
> reg$coef
  Intercept          X
  1.239003  1.002933
> reg$residual
[1]  0.6935484 -0.2800587  0.7023460 -2.3123167 -0.2741935 -1.2917889  3.6730205
[8]  1.6818182  0.7375367 -3.3299120
> muhat <- mean(y)+reg$coef[2]*(23.1-mean(x))
> muhat
      X
24.40674
> se.muhat <- sqrt((990/1000)*(1/(10*8))*sum(reg$residual^2))
> se.muhat
[1] 0.6683408
```

$$\left( \text{SE}(\hat{\mu}_y) = \sqrt{\left(\frac{N-n}{N}\right) \cdot \frac{1}{n(n-2)} \sum_{i=1}^n \hat{e}_i^2} \right)$$

To estimate  $\tau_y$ :

```
> tauhat <- 1000*muhat
> tauhat
      X
24406.74
> se.tauhat <- 1000*se.muhat
> se.tauhat
[1] 668.3408
```

A 99% confidence interval for  $\tau_y$  is given by:

```
> c(tauhat - qt(.995,8)*se.tauhat,tauhat + qt(.995,8)*se.tauhat)
      X      X
22164.20 26649.29
```

Note that the regression and ratio estimates are very similar, but the standard error for the regression estimate is higher than for the ratio estimate. This is caused by the fact that the least squares line goes almost through the origin and that estimating an extra parameter for the regression estimate has not helped much. Based on the scatterplot and the fact that a linear relationship through the origin seems very plausible,, we might want to stick with the

ratio estimate.

**Some R Functions for Ratio & Regression Estimation:** Functions for performing ratio and regression estimation were written and can be found on the course web page under the names “ratio.R” and “regress.R” as script files. To use these functions, simply run the script files (which places their definitions into the memory for R), and then call them, as illustrated below.

To illustrate the use of these functions (both given below), consider the tree counting example from earlier in this handout. Recall that there are 1000 plots with the mean number of trees  $\mu_x$  estimated from aerial photos as 23.1. Ten random plots are ground-truthed.

```
> x <- c(23,14,20,25,12,18,30,27,8,31) # estimated no. of trees from photo
> y <- c(25,15,22,24,13,18,35,30,10,29) # actual number of trees
> ratio.est(x,y,23.1,1000)
r= 1.0625 SE= 0.02799978
mu-hat= 24.54375 SE= 0.646795
tau-hat= 24543.75 SE= 646.795
```

A regression estimate of the mean number of trees per plot,  $\mu_y$ , can be found as:

```
> regr.est(x,y,23.1,1000)
mu-hat= 24.40674 SE= 0.6683408
tau-hat= 24406.74 SE= 668.3408
```

The functions:

```
ratio.est <- function(x, y, mux = NA, N = NA) {
  # estimate of a ratio and ratio estimate of population mean and total.
  # x is auxiliary variable, y is response, mux is population mean
  # of x (xbar is used if no value is given),
  # N is population size (assumed infinite if no value given),
  if(length(x) != length(y)) stop("x and y must be same length")
  n <- length(x)
  fpc <- 1
  if(!is.na(N))
    fpc <- (N - n)/N
  r <- sum(y)/sum(x)
  sr2 <- (1/(n - 1)) * sum((y - r * x)^2)
  if(is.na(mux)) mx <- mean(x) else mx <- mux
}
```

```

cat("r=", r, " SE=", sqrt((fpc * sr2)/(mx^2 * n)), "\n")
if(!is.na(mux))
  cat("mu-hat=", r * mux, " SE=", sqrt((fpc * sr2)/n), "\n")
if(!is.na(N) & !is.na(mux))
  cat("tau-hat=", N*r * mux, " SE=", N*sqrt((fpc * sr2)/n), "\n")
}

regr.est <- function(x, y, mux, N = NA) {
  # regression estimator of a population mean and total.
  # x is auxiliary variable, y is response, mux is population mean
  # of x, N is population size (assumed infinite if no value given).
  if(length(x) != length(y)) stop("x and y must be same length")
  n <- length(x)
  fpc <- 1
  if(!is.na(N))
    fpc <- (N - n)/N
  ab <- lsfit(x, y)
  cat("mu-hat=", ab$coef[1] + ab$coef[2] * mux, " SE=", sqrt((fpc *
    sum(ab$residual^2))/(n * (n - 2))), "\n")
  if(!is.na(N))
    cat("tau-hat=", N*(ab$coef[1] + ab$coef[2] * mux), " SE=", N*sqrt((fpc *
    sum(ab$residual^2))/(n * (n - 2))), "\n")
}

```