

## 22S:166 Computing in Statistics

### The Jackknife

Lecture 9  
September 20, 2006

Kate Cowles  
374 SH, 335-0727  
kcowles@stat.uiowa.edu

### The Jackknife

- another resampling-based method of trying to assess bias, standard error, etc. of a sample-based estimate  $\hat{\theta}$  of a population characteristic  $\theta$
- usually less computationally intensive than the bootstrap
- let  $n$  denote sample size of real dataset
- then there are  $n$  jackknife samples  $\mathbf{x}_{(i)}$ , each obtained by leaving out one observation  $i$  from the original dataset
 
$$\mathbf{x}_{(i)} = x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n$$
- each *jackknife replication*  $\hat{\theta}_{(i)}$  is calculated from the jackknife sample  $\mathbf{x}_{(i)}$  in the same way that  $\hat{\theta}$  was calculated from the whole dataset
- then  $i$ th jackknife *pseudovalue*  $\tilde{\theta}_i$  is calculated as  $n * \hat{\theta} - (n - 1) * \hat{\theta}_{(i)}$

### Jackknife estimate of standard error of $\hat{\theta}$

- Let  $\hat{\theta}_{(\cdot)}$  denote the mean of the jackknife replications  $\hat{\theta}_{(i)}$ . Then

$$\begin{aligned} \widehat{se}_{jack} &= \sqrt{\frac{n-1}{n} \sum_{i=1}^n (\hat{\theta}_{(i)} - \hat{\theta}_{(\cdot)})^2} \\ &= \sqrt{\frac{\sum_{i=1}^n (\tilde{\theta}_i - \tilde{\theta}_{(\cdot)})^2}{n(n-1)}} \end{aligned}$$

- multiplicative factor  $\frac{n-1}{n}$  was chosen to make this work out exactly right for the case of estimating a population mean using  $\hat{\theta} = \bar{x}$

### Jackknife estimate of bias of $\hat{\theta}$

- 

$$\widehat{bias}_{jack} = (n-1)(\hat{\theta}_{(\cdot)} - \hat{\theta})$$

- multiplicative factor  $(n-1)$  is chosen to make this work out exactly right for the case of estimating a population variance using  $\hat{\sigma}^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$ .
- so (first order) jackknife estimator of  $\theta$  is

$$\begin{aligned} \hat{\theta}^{jack} &= n\hat{\theta} - (n-1) \hat{\theta}_{(\cdot)} \\ &= \frac{\sum_{i=1}^n \tilde{\theta}_i}{n} \end{aligned}$$

- why can we expect  $\hat{\theta}^{jack}$  to be less biased than  $\hat{\theta}$

- most estimators are biased
- bias can be approximated as Taylor series expansion of the estimator

$$E(\hat{\theta} - \theta) = \frac{a_1}{n} + \frac{a_2}{n^2} + \frac{a_3}{n^3} + \dots$$

- then it can be shown that

$$E(\hat{\theta}^{jack} - \theta) = -\frac{a_2}{n^2} - \frac{a_2 + a_3}{n^3} - \dots$$

which eliminates term of order  $\frac{1}{n}$  and thus is smaller than the bias of  $\hat{\theta}$

## Relationship between jackknife and bootstrap

- unless  $n$  is large, jackknife is less computationally intensive
- but if jackknife uses fewer samples than bootstrap, then jackknife is using less information
- jackknife may be considered an approximation to the bootstrap
- jackknife does not work well if the quantity being estimated is not “smooth”
  - e.g. median can change sharply with deletion of a single observation

### Jackknife for cities population example

```
> jackex
function()
{
  library(boot) # load library and dataset city
  n <- nrow(city)
  attach(city)
  thetahat <- rep(0,n)
  thetawig <- rep(0,n)
  thetahatn <- mean(x) / mean(u)
  for(i in 1:n) {
    thetahat[i] <- mean(x[-i])/mean(u[-i])
    thetawig[i] <- n * thetahatn - (n-1) * thetahat[i]
  }
  thetahat.dot <- mean(thetahat)
  thetawig.dot <- mean(thetawig)
  se.thetan <- sqrt(sum((thetawig - thetawig.dot)^2) / (n * (n-1)))
  se.thetan2 <- sqrt((n-1) * sum((thetahat - thetahat.dot)^2) / n)
  unbiased2 <- n * thetahatn - (n-1) * thetahat.dot

  list(thetahatn = thetahatn, thetahat = thetahat, thetawig = thetawig,
  se.thetan = se.thetan, se.thetan2 = se.thetan2, thetawig.dot = thetawig.dot,
  unbiased2 = unbiased2)
}
> jackex()
$thetahatn
[1] 1.520312

$thetahat
[1] 1.653386 1.588665 1.561313 1.546638 1.516892 1.509121
   1.510638 1.499190
[9] 1.413115 1.446708
```

```
$thetawig
[1] 0.3226469 0.9051360 1.1513115 1.2833853 1.5510980
   1.6210354 1.6073803
[8] 1.7104184 2.4850922 2.1827488

$se.thetan
[1] 0.194791

$se.thetan2
[1] 0.194791

$thetawig.dot
[1] 1.482025

$unbiased2
[1] 1.482025
```

## Jackknife for sample mean as estimator of population mean

```
function()
{
  blood.flow <- read.table("blood.flow.dat")
  print(blood.flow)
  n <- nrow(blood.flow)
  thetahat <- rep(0, n)
  thetawig <- rep(0, n)
  muhatn <- mean(blood.flow[, 1]) #
  for(i in 1:n) {
    thetahat[i] <- mean(blood.flow[ - i, 1])
    thetawig[i] <- n * muhatn - (n - 1) * thetahat[i]
  }
  thetawig.dot <- mean(thetawig)
  se.muhat <- sqrt(sum((thetawig - thetawig.dot)^2)/(n * (n - 1)))
  list(muhatn = muhatn, thetahat = thetahat, thetawig = thetawig,
       thetawig.dot = thetawig.dot, var.muhat = var.muhat)
}

  V1 V2
1 115 138
2 170 172
3 142 159
4 138 147
5 280 166
6 470 345
7 480 387
8 141 131
9 390 375
$muhatn:
[1] 258.4444
```

## Jackknife for sample correlation coefficient as estimator of population correlation coefficient

```
function()
{
  blood.flow <- read.table("blood.flow.dat")
  print(blood.flow)
  n <- nrow(blood.flow)
  thetahat <- rep(0, n)
  thetawig <- rep(0, n)
  thetahatn <- cor(blood.flow)[1, 2] #
  for(i in 1:n) {
    thetahat[i] <- cor(blood.flow[ - i, ])[1, 2]
    thetawig[i] <- n * thetahatn - (n - 1) * thetahat[i]
  }
  thetawig.dot <- mean(thetawig)
  stderr.thetahat <- sqrt(sum((thetawig - thetawig.dot)^2)/(n * (n - 1)))
  list(thetahatn = thetahatn, thetahat = thetahat, thetawig = thetawig,
       thetawig.dot = thetawig.dot, stderr.thetahat = stderr.thetahat)
}

  V1 V2
1 115 138
2 170 172
3 142 159
4 138 147
5 280 166
6 470 345
7 480 387
8 141 131
9 390 375

$thetahatn:
[1] 0.9448479
```

```
$thetahat:
[1] 276.375 269.500 273.000 273.500 255.750 232.000 230.750 273.125 242.000

$thetawig:
[1] 115 170 142 138 280 470 480 141 390

$thetawig.dot:
[1] 258.4444

$se.muhat:
[1] 50.25936
```

```
$thetahat:
[1] 0.9405566 0.9434308 0.9433717 0.9407706 0.9766233 0.9395961 0.9205287
[8] 0.9395995 0.9583461

$thetawig:
[1] 0.9791780 0.9561844 0.9566569 0.9774661 0.6906443 0.9868617 1.1394014
[8] 0.9868350 0.8368616

$thetawig.dot:
[1] 0.9455655

$stderr.thetahat:
[1] 0.04085297
```