Sample Size Calculations Using R PracTools Package

January 22, 2020
Agenda

• Introductions
• Sample size theory
• PracTools package and functions
• Questions / Suggestions
Introductions

• George Zipf
  • Chief Statistician, Dept. of Transportation, Office of Inspector General
  • UMD JPSM graduate (2017)
  • George.Zipf@oig.dot.gov

• Dr. Richard Valliant
  • Research Professor Emeritus at Universities of Michigan & Maryland at University of Michigan
  • Author of “Practical Tools for Designing and Weighting Survey Samples”
  • Maintains PracTools package
Practical Tools for Designing and Weighting Survey Samples
Sample Size: Why Sample

• Census is expensive
• Sampling saves time and money and can still have reasonable precision
• Sample size calculations allow the survey designer to define “reasonable precision”
SRS Sample Size: Formula

• Variance formula is: 
  \[ V(\bar{y}) = \left(1 - \frac{n}{N}\right) \frac{S^2}{n} \]

• This can be re-written as: 
  \[ n = \frac{n_0}{1 + \frac{n_0}{N}} \text{ where } n_0 = \frac{S^2}{\bar{y}^2} \cdot CV_0^2 \].

• It gets more complicated from here
• This is why we use PracTools
Sample Size: What you need

• N (Population Size)
• $\alpha$ for Confidence Interval
• Variance or Coefficient of Variation (CV)
• Sample unit cost, if these vary
• $\beta$, if power is needed
Sample Size constraints

- Standard Error of Estimate
- Coefficient of Variation (CV) = Standard Deviation / Mean
- Margin of Error – Probability within Fixed Bound
- Cost
- Power – Probability of Detectable Difference
PracTools

• If you haven’t already, please bring up R and:

install.packages("PracTools")
library("PracTools")
## PracTools Sample Size Functions

<table>
<thead>
<tr>
<th>Simple Random Sample (SRS) Design</th>
<th>PracTools function(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRS – Continuous variable</td>
<td>nCont, nContMoe</td>
</tr>
<tr>
<td>SRS – Proportions</td>
<td>nProp, nPropMoe</td>
</tr>
<tr>
<td>SRS – Strata</td>
<td>strAlloc</td>
</tr>
<tr>
<td>SRS – Continuous, power required</td>
<td><code>power.t.test</code> (stats package)</td>
</tr>
<tr>
<td>SRS – Proportion, power required</td>
<td><code>power.prop.test</code> (stats package)</td>
</tr>
<tr>
<td>SRS – 2 stage</td>
<td>clusOpt2</td>
</tr>
</tbody>
</table>
SRS sample size for continuous variable

• Example: You are the primary researcher on a survey about business behavior. While there are many variables of interest, contract amount is a key one. Let’s assume that for this population, the average contract is $200,000 and the variance is $980,000,000,000.

\[
n_{\text{Cont}}(CVO=NULL, V0=NULL, S2=NULL, ybarU=NULL, CVpop=NULL, N=\infty)\]

\[
> \text{nCont}(CVO = 0.30, S2 = 980000000000, ybarU = 200000)\\
[1] 272.2222
\]
\[
> \text{nCont}(CVO = 0.30, CVpop = \sqrt{980000000000}/200000)\\
[1] 272.2222
\]
SRS sample size: N matters

- The finite population correction (fpc) factor can make a big difference. Note that the default is N=Infinity.

```r
nCont(CV0=NULL, V0=NULL, S2=NULL, ybarU=NULL, CVpop=NULL, N=Inf)
```

```r
> nCont(CV0 = 0.30, S2 = 980000000000, ybarU = 200000)
[1] 272.2222
> nCont(CV0 = 0.30, S2 = 980000000000, ybarU = 200000, N=5000)
[1] 258.1665
```
SRS sample size for continuous variable: MOE

• Let’s say you need to be able to bound the margin of error (MOE). This is good for statements such as “the population value is within 5% of the sample estimate.”

• Expanding on the previous example where average contract is $200,000 and the variance is $980,000,000,000, let’s say you need to be able to bound the margin of error.

\[
n_{\text{ContMoe}}(\text{moe.sw, e, alpha=0.05, CV.pop=NULL, S2=NULL, ybarU=NULL, N=Inf})
\]

• In the \text{nContMoe} function, there are two new parameters: \text{moe.sw}, and \text{e}
SRS sample size for cont. var. – Margin of Error, cont’d

• moe.sw
  • 1 = CI half-width on the variance of the mean; requires S2.
  • 2 = CI half-width on the coefficient of variation; requires CVpop or S2 and ybarU

• e is the desired margin of error in percentage terms
  • 0 < e < 1

• nContMoe examples:
  > nContMoe(moe.sw=1, e=20000, alpha=0.05, S2=980000000000)
    [1] 9411.574
  > nContMoe(moe.sw=1, e=50000, alpha=0.05, S2=980000000000)
    [1] 1505.852
  > nContMoe(moe.sw=2, e=0.10, alpha=0.05, S2=980000000000, ybarU=200000)  # Moe is 10% of 200,000 = 20,000
    [1] 9411.574
  > nContMoe(moe.sw=2, e=0.25, alpha=0.05, S2=980000000000, ybarU=200000)
    [1] 1505.852
SRS sample size for proportion

• Functions: nProp and nPropMoe
  • nProp(CV0=NULL, V0=NULL, pU=NULL, N=Inf)
  • nPropMoe(moe.sw, e, alpha, pU, N=Inf)

• nProp examples:
  > nProp(CV0 = 0.1, pU = 0.5)
  [1] 100
  > nProp(CV0 = 0.1, pU = 0.1)
  [1] 900
  > nProp(CV0 = 0.1, pU = 0.9)
  [1] 11.11111

  > nProp(V0 = 0.0025, pU = 0.5)
  [1] 100
  > nProp(V0 = 0.0025, pU = 0.1)
  [1] 36
  > nProp(V0 = 0.0025, pU = 0.9)
  [1] 36
SRS sample size for proportion, cont’d

- nPropMoe is frequently what people think “should” be the sample size calculation
- Formula is in Cochran, *Sampling Techniques*, 3rd Edition, Section 4.4, pg. 76
- nPropMoe examples:
  - > nPropMoe(moe.sw=1, e=0.10, alpha=0.10, pU = 0.5)
    [1] 67.63859
  - > nPropMoe(moe.sw=1, e=0.10, alpha=0.05, pU = 0.5)
    [1] 96.03647
  - > nPropMoe(moe.sw=1, e=0.05, alpha=0.05, pU = 0.5)
    [1] 384.1459
  - > nPropMoe(moe.sw=2, e=0.10, alpha=0.10, pU = 0.5)
    [1] 270.5543
  - > nPropMoe(moe.sw=2, e=0.10, alpha=0.05, pU = 0.5)
    [1] 384.1459
  - > nPropMoe(moe.sw=2, e=0.05, alpha=0.05, pU = 0.5)
    [1] 1536.584
SRS sample size for strata

strAlloc(n.tot = NULL, Nh = NULL, Sh = NULL, cost = NULL, ch = NULL, V0 = NULL, CV0 = NULL, ybarU = NULL, alloc=)

• Constraint: n
  • Proportional allocation
  • Equal allocation
  • Neyman allocation

• Cost constrained to total budget
• Precision constrained to overall variance or CV
• strAlloc computes all the above except equal allocation, but requires different inputs
• alloc must be one of “prop”, “neyman”, “totcost”, “totvar”
SRS sample size for strata, cont’d

Nh <- c(215, 65, 252, 50, 149, 144)  # Strata size
Sh <- c(26787207, 10645109, 6909676, 11085034, 9817762, 44553355) # Strata SD
ch <- c(1400, 200, 300, 600, 450, 1000) # Strata unit cost

# Neyman allocation
str Alloc(n.tot = 100, Nh = Nh, Sh = Sh, alloc = "neyman")

# cost constrained allocation
str Alloc(Nh = Nh, Sh = Sh, cost = 100000, ch = ch, alloc = "totcost")

# allocation with CV target of 0.05
str Alloc(Nh = Nh, Sh = Sh, CV0 = 0.05, ch = ch, ybarU = 11664181, alloc = "totvar"
SRS sample size for strata, cont’d

• Here are the results for the Neyman allocation:

```r
strAlloc(n.tot = 100, Nh = Nh, Sh = Sh, alloc = "neyman")
```

allocation = neyman

\[
\begin{align*}
\text{Nh} & = 215, 65, 252, 50, 149, 144 \\
\text{Sh} & = 26787207, 10645109, 6909676, 11085034, 9817762, 44553355 \\
\text{nh} & = 34.641683, 4.161947, 10.473487, 3.333804, 8.798970, 38.590108 \\
\text{nh/n} & = 0.34641683, 0.04161947, 0.10473487, 0.03333804, 0.08798970, 0.38590108
\end{align*}
\]

anticipated SE of estimated mean = 1727173

• The output for the other allocation types is similar.
SRS sample size using power calculations

• Functions:
  • power.t.test
  • power.prop.test

• What is required:
  • Delta (Detectable Difference Level) between two populations
  • Probability of obtaining a significant result when the true difference is Delta (Note: Power = 1 – β)
  • Significance Level α

• Part of R stats package
power.t.test

- Exactly one of the parameters n, delta, power, sd, and sig.level must be passed as NULL, and that parameter is determined from the others.

```r
power.t.test(sd=1, sig.level = 0.05, power=.8, delta=0.1, alternative = "two.sided", type = "two.sample")
```

Two-sample t test power calculation

- n = 1570.737
- delta = 0.1
- sd = 1
- sig.level = 0.05
- power = 0.8
- alternative = two.sided

NOTE: n is number in *each* group
power.prop.test

• Exactly one of the parameters n, p1, p2, power, and sig.level must be passed as NULL, and that parameter is determined from the others.

• Sample size can be function of relative risk, where P1 – P2 = P2(Rel. Risk – 1)

power.prop.test(p1 = 0.15, p2 = 0.18, sig.level = 0.05, power=.8, alternative = "one.sided")

Two-sample comparison of proportions power calculation
n = 1891.846
p1 = 0.15
p2 = 0.18
sig.level = 0.05
power = 0.8
alternative = one.sided
NOTE: n is number in *each* group
Two-stage sampling

clusOpt2(C1, C2, delta, unit.rv, k=1, CV0=NULL, tot.cost=NULL, cal.sw)

• Parameters
  • Cost
    • C1 = PSU cost
    • C2 = SSU cost
    • tot.cost = total budget
  • CV0 = Target CV
  • delta = homogeneity; 0 < delta < 1
  • unit.rv = unit relvariance
    • k = ratio of $B^2 + W^2$ to unit relvariance
    • cal.sw = 1 for fixed total budget and = 2 for target CV0

• delta will affect how sample is optimized over PSUs and SSUs
• Either tot.cost or CV0 must be provided
Two-stage sampling

clusOpt2(C1=750, C2=100, delta=0.05, unit.rv=1, k=1, tot.cost=100000, cal.sw=1)

C1 = 750
C2 = 100
delta = 0.05
unit relvar = 1
k = 1
cost = 1e+05
m.opt = 51.4
n.opt = 11.9
CV = 0.0502
Conclusions:

• We have covered R PracTools functions for sample size calculations.
• For single stage designs, the results of these functions are pretty straightforward.
• If design effect is an issue, remember to multiply the calculated sample size by the design effect for the survey sample size.
• Multi-stage designs are more complicated, but R PracTools does have functions to address the issues.
• It’s always a good idea to discuss survey design with other statisticians.
• And please send us topics for future presentations!

• Questions?