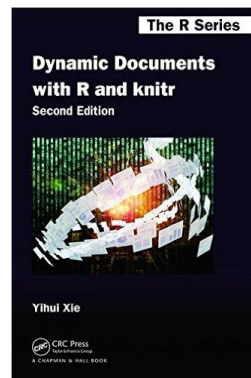


Open Source Software for Automating and Ensuring Quality in Official Statistics: An Example Using R and RStudio for Stratified Simple Random Sampling

Darryl V. Creel



Stratified Simple Random Sampling: Quality Control

Darryl V. Creel, RTI International

Wednesday, October 24, 2018

Quality, Reproducibility, Rigor, Standardization, and Transparency (QR²ST). Quality is an important aspect Federal statistical information.

Quality

- : how good or bad something is
- : a characteristic or feature that someone or something has : something that can be noticed as a part of a person or thing
- : **a high level of value or excellence**

from www.merriam-webster.com

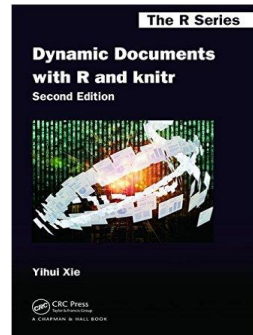
Planning for quality control should begin before the project starts: folder structure, naming conventions (folders, programs, functions, variables, data sets, etc.), automation, inputs/outputs, responsibilities, version control, testing, etc.

```
|----- ProjectName
|         |----- Computing
|         |----- Management
|         |----- Statistics
|         |         |----- A_Planning
|         |         |----- B_FrameDevelopment
|         |         |----- C_Sampling
|         |         |----- D_DataCollection
|         |         |----- E_DataProcessing
|         |         |         |----- EA_Weighting
|         |         |         |----- EB_Editing
|         |         |         |----- EC_Imputation
|         |         |----- F_Analysis
|         |         |----- G_Publication
|         |----- SubjectMatter
|         |----- SurveyMethodology
```

Programs in the sampling folder (C_Sampling).

```
|----- C_Sampling
|         |----- Ca_stratifiedSimpleRandomSampling_selection.Rmd
|         |----- Cb_stratifiedSimpleRandomSampling_qualityControl.Rmd
```

How can we increase quality (better), lower labor costs (cheaper), require less calendar time (faster), and document quality control processes?



Stratified Simple Random Sampling: Quality Control

Darryl V. Creel, RTI International

Wednesday, October 24, 2018

RStudio, an integrated development environment for R.

The screenshot displays the RStudio integrated development environment (IDE) interface. The main window is divided into several panes:

- Code Editor:** Shows an R Markdown file named "stratified_simple_random_sampling...". The code includes a title "Random Sampling: Quality Control' RTI International", a pre-render block for parameters, and R code to load packages (tidyverse, xtable) and set parameters (sampling.unit <- "physician"). A context menu is open over the code, offering options like "Knit to HTML", "Knit to PDF", and "Knit to Word".
- Environment Pane:** Located on the right, it shows the "Global Environment" and indicates that the environment is currently empty.
- Terminal:** At the bottom, it displays the R version (3.5.1), copyright information, and a series of help messages for users, including instructions on how to cite R and use help functions.
- Files Pane:** At the bottom right, it shows the "R: System Date and Time" documentation page, which includes a description of the `date` function, its usage, value format, and references.

```
1 ---
2 title Random Sampling: Quality Control'
3 author RTI International
4 date
5 output
6 pdf
7 word
8 html
9
10 bibliography
11 ---
12
13 {r parameters, echo = FALSE}
14
15
16 ### Include packages
17
18 require(tidyverse, quietly = TRUE)
19 require(xtable, quietly = TRUE)
20
21 packages <- c("base", "dplyr", "forcats", "ggplot2", "knitr", "purrr", "readr", "stringr", "tibble",
22 "tidyr", "xtable")
23 knitr::write_bib ( packages, file = "packages.bib")
24
25 ### Set parameters
26
27 sampling.unit <- "physician"
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

--- Attaching packages ----- tidyverse 1.2.1 ---
v ggplot2 3.0.0 v purrr 0.2.5

4:23 Stratified Simple Random Sampling: Quality Control R Markdown

R version 3.5.1 (2018-07-02) -- "Feather Spray"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |

date (base) R Documentation

System Date and Time

Description

Returns a character string of the current system date and time.

Usage

```
date ()
```

Value

The string has the form "Fri Aug 20 11:11:00 1999", i.e., length 24, since it relies on POSIX's `ctime` ensuring the above fixed format. Timezone and Daylight Saving Time are taken account of, but *not* indicated in the result.

The day and month abbreviations are always in English, irrespective of locale.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

See Also

[Sys.Date](#) and [Sys.time](#); [Date](#) and [DateTimeClasses](#) for objects representing date and time.

Literate programming weaves the text and program code together.

In a sampling stratum, the sample size should equal the sum of the probabilities of selection. For the h^{th} sampling stratum, the sample size, n_h , should equal the sum of the probabilities of selection, p_{hi} . That is, in the h^{th} sampling stratum, the check to ensure that the probability of selection was calculated correctly is

```
$$

$$n_h = \sum_{i=1}^{N_h} p_{hi}.$$

$$

```{r probSelSum, type = "latex", results = "asis", echo = FALSE}

psSumPop <- sampSize %>% select(samplingStratum, sampleSize)

psSumSamp <- frame %>% select(samplingStratum, probabilityOfSelection) %>% group_by(samplingStratum) %>% summarize(psSum =
sum(probabilityOfSelection))

psSumCheck <- full_join(psSumPop, psSumSamp, by = "samplingStratum") %>% mutate(diff = round(psSum - sampleSize))

psSumDiff <- psSumCheck %>% filter(diff != 0 | is.na(diff))

if (dim(psSumDiff)[1] == 0) {
 cat("Pass: All sampling strata have the sum of the probabilities of selection equal to the sample size.\n")
} else {
 cat("Fail: At least one samling stratum does not have the sum of the probabilities of selection equal to the sample size.\n")
 psSumDiff.xt <- xtable(psSumDiff)
 caption(psSumDiff.xt) <- "Sum Probabilites of Selection not Equal Sample Size"
 print(psSumDiff.xt, include.rownames = FALSE, caption.placement = "top", comment = FALSE)
}

```
```

Using the pipe, `%>%`, in R. It comes from the **magrittr** package by Stefan Milton.

```
sampSize <- read_csv("sampleSize_01.csv") %>%  
mutate(posPop = sampleSize/populationCount, dw =  
populationCount/sampleSize)
```

```
sampSize <- read_csv("sampleSize_01.csv")  
sampSize$posPop <-  
sampSize$sampleSize/sampSize$populationCount  
sampSize$dw <-  
sampSize$populationCount/sampSize$sampleSize
```


Using the pipe, %>%, in R. It comes from the **magrittr** package by Stefan Milton.

```
psSumSamp <- frame %>% select(samplingStratum,  
probabilityOfSelection) %>% group_by(samplingStratum)  
%>% summarize(psSum = sum(probabilityOfSelection))
```

```
psSumSamp2 <- tapply(X = frame$probabilityOfSelection,  
INDEX = frame$samplingStratum, FUN = sum)*
```

* Not quite right does not have information when samplingStratum is missing

In a sampling stratum, the sample size should equal the sum of the probabilities of selection. For the h^{th} sampling stratum, the sample size, $n_{\{h\}}$, should equal the sum of the probabilities of selection, $p_{\{hi\}}$. That is, in the h^{th} sampling stratum, the check to ensure that the probability of selection was calculated correctly is

\$\$

$$n_{\{h\}} = \sum_{i=1}^{N_h} p_{\{hi\}}.$$

\$\$

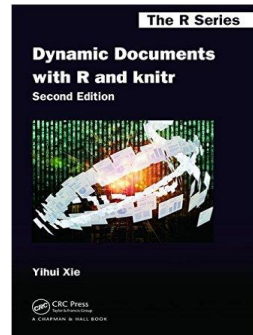
In a sampling stratum, the sample size should equal the sum of the sample indicators. For the h^{th} sampling stratum, the sample size, n_h , should equal the sum of the sample indicators, s_{hi} . That is, in the h^{th} sampling stratum, the check to ensure that the sample indicators were calculated correctly is

$$n_h = \sum_{i=1}^{N_h} s_{hi} .$$

Rstudio includes the table derived from the code.

```
# A tibble: 2 x 4
  samplingStratum sampleSize psSum  diff
      <int>         <int> <dbl> <dbl>
1           2          20   NA     NA
2          NA          NA   0.3    NA
```

How can we increase quality (better), lower labor costs (cheaper), require less calendar time (faster), and document quality control processes? RStudio and knitr.



Stratified Simple Random Sampling: Quality Control

Darryl V. Creel, RTI International

dcreel@rti.org

(301) 770-8229