APPENDIX F
Sampling Design for Forest Biomass Surveys – Chapter 7 Training Module

INTRODUCTION

Please refer to Chapter 7 of the SAR Handbook for further background for the following exercises and related scripts. Scripts and data used for this tutorial can be found here: https://bit.ly/2HNZXWu as well as on the SERVIR global website training page.

Step 1: Download and install R. The download link is available here: https://ftp.osuosl.org/pub/cran/

Optionally, install R Studio (free license) as a GUI to develop your scripts from here: https://www.rstudio.com/products/rstudio/download/#download

After installation, you will have a platform as shown in **Figure 1.1**. Please note these scripts have been created to run on a Windows operating system. When running them in a Linux or Mac environment, they will need slight modifications to function properly. This document provides guidelines to do those changes.

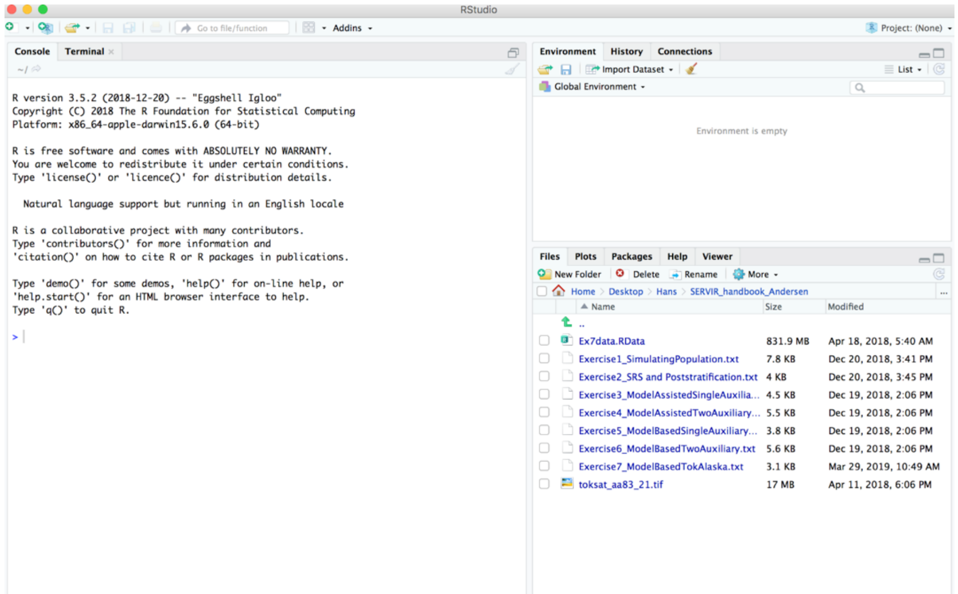


Figure 1.1 *The R Studio interface. The section in the lower right allows navigation to the folder containing the scripts for the exercises in this module.*

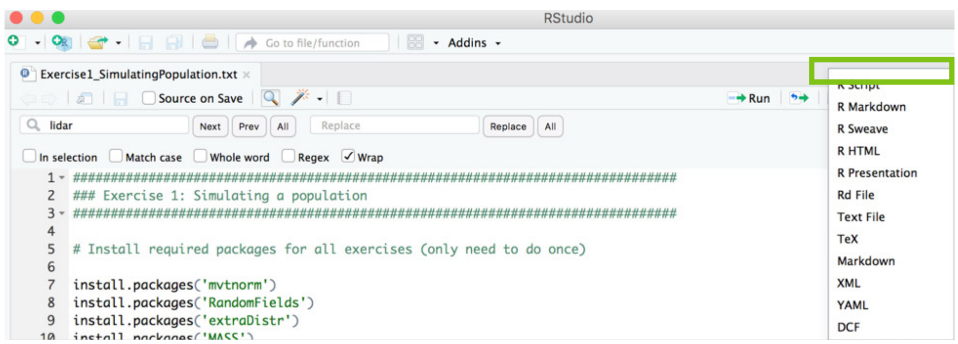


Figure 1.2 *A window displaying options for selecting the file type in R Studio, with "R Script" highlighted.*

EXERCISE 1: Simulating a population

Here we use simulation, implemented in the R statistical software package, to demonstrate the implementation of several SAR-assisted, multi-level sampling designs. Proficiency in R programming is not required to carry out the exercises, since the scripts can be run by simply copying and pasting the code at the R command line. Simulation can be a useful approach to gain insight into the statistical properties of various survey estimators, especially in the case of somewhat complex, multi-level sampling designs (Ene et al., 2016; Saarela et al., 2017).

Open Script #1:

Double click on the script so it appears in the upper left window in RStudio. Make sure in the editor this file is selected as a R Script (**Figure 1.2**).

Install Packages:

In RStudio, install the following packages by copying and pasting the following script lines:

```
install.packages('mvtnorm')
install.packages('RandomFields')
install.packages('extraDistr')
install.packages('MASS')
install.packages('psych')
install.packages('fields')
install.packages('CompRandFld')
install.packages('raster')
install.packages('RColorBrewer')
```

Note these lines are also included in the Exercise 1 script, and are commented with (#). Make sure all these packages are installed correctly before proceeding.

Modify script appropriately:

In line 249, starting with "save.image", modify where the final file will be saved (the path) and check the file name to which it will be saved; in this case, "servirsimdata.RData" (highlighted section below):

```
# save the simulated population to a RData file - this
might take a minute or two...
# this is the file that will
be loaded at the beginning of
all subsequent exercises
save.image("C:\\Users\\handersen\\Desktop\\SERVIR _ handbook\\servirsimdata.RData")
```

Run Script:

Then select the rest of the script, starting with "library.." and click on Run (**Figure 1.3**). The purpose of this script is to create and save the simulated data file: "servirsimdata.RData". This file will be used in subsequent exercises. As final step, check that such file was created. Navigate to the folder where it should be and make sure it exists.

Additional note: If you are working on a Windows computer, the single backslash will work when writing paths in the script, but if you work on a Mac, the double backslash is preferred.

EXERCISE 2: Assessment of simple random sampling estimator via simulation

The statistical properties of the various estimators can be assessed using the simulated population developed previously. In this exercise, at each iteration a simple random sample (SRS) of n first-phase elements is drawn from the population. The point estimator $\hat{\mu}$ and the variance estimator $\hat{V}(\hat{\mu})$ are calculated, as well as the coverage probability of the 95% confidence interval of the point estimator. These statistics indicate the bias and precision of the point estimator and the variance estimator.

Step 1: Modify this line with the appropriate folder and path for the file created in first script:

```
load("C:\\Users\\handersen\\Desktop\\Handbook\\servirsimdata.RData")
```

Step 2: Make sure the "R Script" is selected on the low right corner (**Figure 1.2**).

Step 3: After loading the file, make sure all objects were created. You can enter:

```
<ls()
```

to see all the objects contained in the file.

Step 4: If you're running the script on a Mac, the Progress Bar may not work, as it is a Windows-specific function. Therefore, comment out lines 98, 99 and 129 in this

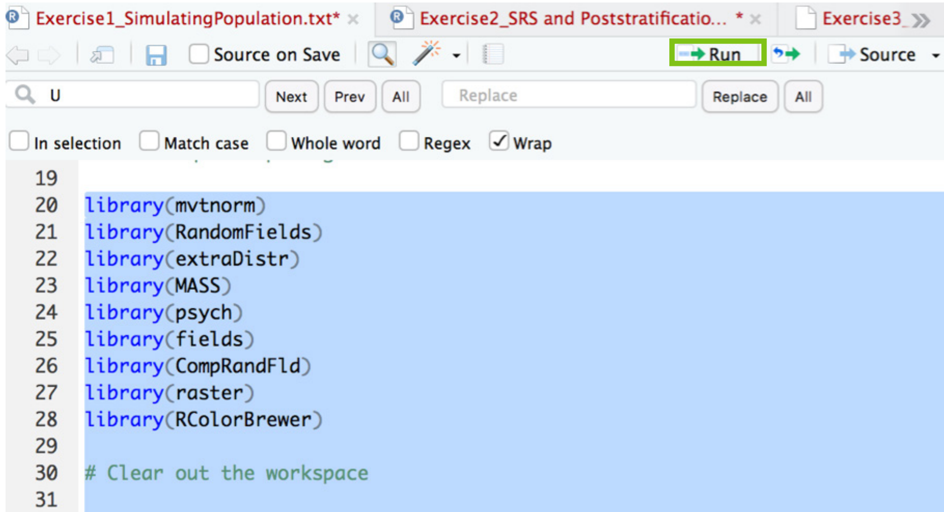


Figure 1.3 *Screenshot of the R Studio interface with the "Run" option highlighted.*

script. Alternately, do not run those lines, as seen below:

```
# update progress bar
#Sys.sleep(0.1)
#setWinProgressBar(pb, t, title=paste(round(t/iterations*100, 0), "% done"))
# close the progress bar
#close(pb)
```

Step 5: You should get values and figures, and your console should look like this:

```
> # output results
> print(paste("n: ", n))
[1] "n:  15"
> print(paste("Relative bias for SRS estimator: ", (mean(Muhat_vec.srs)-Mu.U)/Mu.U  ))
[1] "Relative bias for SRS estimator:  0.00807804934249307"
> print(paste("Relative SE for SRS estimator: ", sd(Muhat_vec.srs)/mean(Muhat_vec.srs)))
[1] "Relative SE for SRS estimator:  0.273858702223831"
> print(paste("95% coverage prob for SRS estimator: ", sum(cov_vec.srs)/iterations))
[1] "95% coverage prob for SRS estimator:  0.883"
>
> # output results
> print(paste("Relative bias for PS estimator: ", (mean(Muhat_vec.ps)-Mu.U)/Mu.U  ))
[1] "Relative bias for PS estimator:  0.00983612308208642"
> print(paste("Relative SE for PS estimator: ", sd(Muhat_vec.ps)/mean(Muhat_vec.ps)))
[1] "Relative SE for PS estimator:  0.279487533308954"
> print(paste("95% coverage prob for PS estimator: ", sum(cov_vec.ps)/iterations))
[1] "95% coverage prob for PS estimator:  0.877"
```

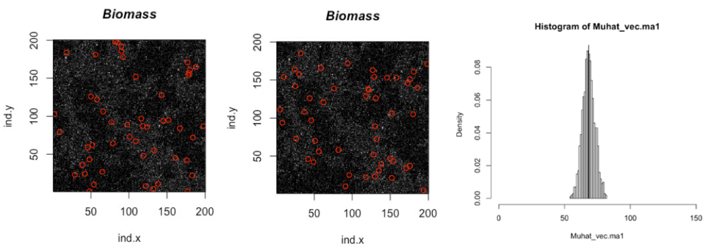
You can use the <ls() command to list all the objects that are now available. See below for a partial list:

```
> ls()
 [1] "AB"            "bio"           "CI.95perc"     "cov_vec.ps"
 [5] "cov_vec.srs"   "fnf"           "hx"            "ind.x"
 [9] "ind.y"         "iterations"    "lidar"         "model"
[13] "Mu.U"          "Muhat_vec.ps"  "Muhat_vec.srs" "Muhat.ps"
[17] "Muhat.srs"     "n"             "N"             "n.f"
[21] "n.nf"          "p"             "randsamp"      "s"
```

EXERCISE 3: Assessment of model-assisted estimator with single auxiliary via simulation

In this exercise, we assess the statistical properties of the model-assisted estimator with one source of auxiliary data (assumed to be collected wall-to-wall, such as SAR imagery) and various sample sizes of field plots.

Repeat steps 1 through 4 of Exercise 2. If running on a Mac computer, comment out the script lines for the Progress Bar as shown in the previous section.
This will result in the creation of the following figures, shown below:



In addition, you should get the following output results:

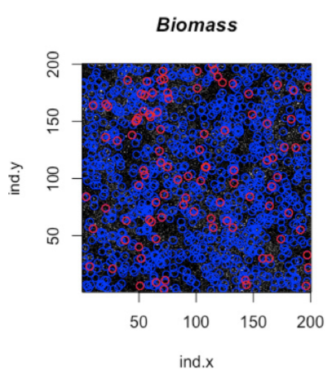
```
> # output results
> print(paste("n: ", n))
[1] "n:  50"
> print(paste("Relative bias for SRS estimator: ", (mean(Muhat_vec.srs)-Mu.U)/Mu.U  ))
[1] "Relative bias for SRS estimator:  0.00566173451784223"
> print(paste("Relative SE for SRS estimator: ", sd(Muhat_vec.srs)/mean(Muhat_vec.srs)))
[1] "Relative SE for SRS estimator:  0.143681227009878"
> print(paste("95% coverage prob for SRS estimator: ", sum(cov_vec.srs)/iterations))
[1] "95% coverage prob for SRS estimator:  0.946"
>
> print(paste("Relative bias for MA estimator: ", (mean(Muhat_vec.ma1)-Mu.U)/Mu.U  ))
[1] "Relative bias for MA estimator:  0.00172107039170955"
> print(paste("Relative SE for MA estimtor: ", sd(Muhat_vec.ma1)/mean(Muhat_vec.ma1)))
[1] "Relative SE for MA estimtor:  0.0682497752244089"
> print(paste("95% coverage prob for MA estimator: ", sum(cov_vec.ma1)/iterations))
[1] "95% coverage prob for MA estimator:  0.918"
>
```

EXERCISE 4: Assessment of model-assisted estimator with two auxiliaries via simulation

In this exercise, we assess the statistical properties of a model-based estimator of mean biomass. In order to illustrate the perils of an incorrectly specified model in the context of model-based estimation, for this exercise the model is developed from a sample selected only from the forested plots within the population. This is then used to estimate biomass–using both model-assisted and model-based estimators–over the entire population.

Repeat steps 1 and 2 from Exercise 2. If running on a Mac computer, comment out the script lines for the Progress Bar as shown in Exercise 2, step 4.

This will result in the creation of the following figure, shown below:



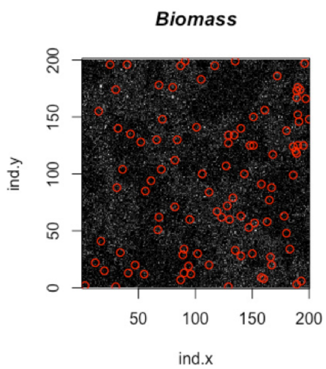
You should also get the following output results:

```
> print(paste("n.1: ", n.1))
[1] "n.1:  1000"
> print(paste("n.2: ", n.2))
[1] "n.2:  100"
> print(paste("Relative bias for SRS estimator: ", (mean(Muhat_vec.srs)-Mu.U)/Mu.U  ))
[1] "Relative bias for SRS estimator:  0.00772093714715049"
> print(paste("Relative SE for SRS estimator: ", sd(Muhat_vec.srs)/mean(Muhat_vec.srs)))
[1] "Relative SE for SRS estimator:  0.0897102448658417"
> print(paste("95% coverage prob for SRS estimator: ", sum(cov_vec.srs)/iterations))
[1] "95% coverage prob for SRS estimator:  0.99"
>
> print(paste("Relative bias for MA estimator (1 aux): ", (mean(Muhat_vec.ma1)-Mu.U)/Mu.U  ))
[1] "Relative bias for MA estimator (1 aux):  0.00573356140916007"
> print(paste("Relative SE for MA estimtor (1 aux): ", sd(Muhat_vec.ma1)/mean(Muhat_vec.ma1)))
[1] "Relative SE for MA estimtor (1 aux):  0.0770069221725778"
> print(paste("95% coverage prob for MA estimator (1 aux): ", sum(cov_vec.ma1)/iterations))
[1] "95% coverage prob for MA estimator (1 aux):  0.95"
>
> print(paste("Relative bias for MA estimator (2 aux): ", (mean(Muhat_vec.ma2)-Mu.U)/Mu.U  ))
[1] "Relative bias for MA estimator (2 aux):  0.00587626584655108"
> print(paste("Relative SE for MA estimtor (2 aux): ", sd(Muhat_vec.ma2)/mean(Muhat_vec.ma2)))
[1] "Relative SE for MA estimtor (2 aux):  0.048642993404468"
> print(paste("95% coverage prob for MA estimator (2 aux): ", sum(cov_vec.ma2)/iterations))
[1] "95% coverage prob for MA estimator (2 aux):  0.93"
>
```

EXERCISE 5: Assessment of model-based estimator with single auxiliary via simulation

In this exercise, we assess the statistical properties of the model-assisted estimator with two sources of auxiliary data (e.g. field plots, photo plots, and wall-to-wall SAR imagery). This estimator does not assume that the model is correct, but does assume that the underlying sample of field and/or remote sensing data is collected via a probability sample.

Repeat steps 1 and 2 of Exercise 2. If running on a Mac computer, comment out the script lines for the Progress Bar as shown in Exercise 2, step 4.
The following plot should be created:



And the console should show output results as follows:

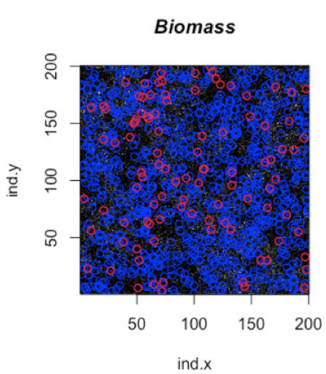
```
> print(paste("n: ", n))
[1] "n:  100"
> print(paste("Relative bias for SRS estimator: ", (mean(Muhat_vec.srs)-Mu.U)/Mu.U  ))
[1] "Relative bias for SRS estimator:  -0.00384690540648969"
> print(paste("Relative SE for SRS estimator: ", sd(Muhat_vec.srs)/mean(Muhat_vec.srs)))
[1] "Relative SE for SRS estimator:  0.10153260962062"
> print(paste("95% coverage prob for SRS estimator: ", sum(cov_vec.srs)/iterations))
[1] "95% coverage prob for SRS estimator:  0.946"
>
> print(paste("Relative bias for MB estimator (1 aux): ", (mean(Muhat_vec.mb1)-Mu.U)/Mu.U  ))
[1] "Relative bias for MB estimator (1 aux):  0.00280324872056235"
> print(paste("Relative SE for MB estimtor (1 aux): ", sd(Muhat_vec.mb1)/mean(Muhat_vec.mb1)))
[1] "Relative SE for MB estimtor (1 aux):  0.0779438850708484"
> print(paste("95% coverage prob for MB estimator (1 aux): ", sum(cov_vec.mb1)/iterations))
[1] "95% coverage prob for MB estimator (1 aux):  0.943"
>
```

EXERCISE 6: Model-based estimation with two auxiliaries

In this exercise, we assess the statistical properties of the model-based estimator with two sources of auxiliary data (e.g. field plots, photo plots, and wall-to-wall SAR imagery). This estimator assumes that the model is correct (unbiased), but does not assume that the underlying sample of field and/or remote sensing data is collected via a probability sample.

Repeat steps 1 and 2 of Exercise 2. If running on a Mac computer, comment out the script lines for the Progress Bar as shown in Exercise 2, step 4.

This will result in the creation of the following figure, shown below:



You should also get the following output results:

```
> print(paste("n.1: ", n.1))
[1] "n.1:  1000"
> print(paste("n.2: ", n.2))
[1] "n.2:  100"
> print(paste("relative bias srs: ", (mean(Muhat_vec.srs)-Mu.U)/Mu.U  ))
[1] "relative bias srs:  0.00772093714715049"
> print(paste("relative se srs: ", sd(Muhat_vec.srs)/mean(Muhat_vec.srs)))
[1] "relative se srs:  0.0897102448658417"
> print(paste("coverage prob srs: ", sum(cov_vec.srs)/iterations))
[1] "coverage prob srs:  0.99"
>
> print(paste("relative bias mb1: ", (mean(Muhat_vec.mb1)-Mu.U)/Mu.U  ))
[1] "relative bias mb1:  0.00573356140916007"
> print(paste("relative se mb1: ", sd(Muhat_vec.mb1)/mean(Muhat_vec.mb1)))
[1] "relative se mb1:  0.0770069221725777"
> print(paste("coverage prob mb1: ", sum(cov_vec.mb1)/iterations))
[1] "coverage prob mb1:  0.95"
>
> print(paste("relative bias mb2: ", (mean(Muhat_vec.mb2)-Mu.U)/Mu.U  ))
[1] "relative bias mb2:  0.00494724740208265"
> print(paste("relative se mb2: ", sd(Muhat_vec.mb2)/mean(Muhat_vec.mb2)))
[1] "relative se mb2:  0.0502598157306141"
> print(paste("coverage prob mb2: ", sum(cov_vec.mb2)/iterations))
[1] "coverage prob mb2:  0.96"
```

EXERCISE 7: Model-based estimation with field, lidar plots, and SAR data from Tok, Alaska, USA

In this exercise, we apply the model-based estimators with 2 sources of auxiliary data using an actual dataset collected for a region of interior Alaska (USA). The data consist of: 1) estimates of aboveground tree biomass (Mg/ha) collected over relatively sparse sample of field plots (n_2= 30 1/30th ha circular plots), height-based metrics collected over a denser (systematic) sample of lidar plots (n_1= 325 1/30th ha circular plots) and 3) wall-to-wall L-band satellite SAR derived imagery. Tree height, tree diameter and species were collected for each tree on the plots, and allometric models were applied to these measurements to estimate tree-level and aggregated plot-level biomass (Yarie et al., 2007).

Please note that for this exercise, we will use different input datasets provided along with the scripts: "Ex7data.RData" and L-band ALOS PALSAR backscatter image, "toksat_aa83_21.tif"

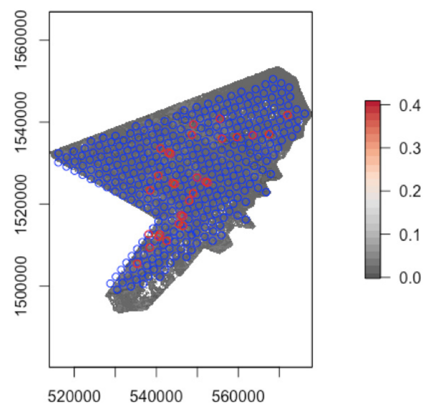
Step 1: Ensure the right path is included in the script for these input files:

```
# Load the data into the workspace
load("C:\\Users\\handersen\\Desktop\\SER-
VIR _ handbook\\Ex7data.RData")
```

Step 2: Select the script and run it. In case you run into an error stating "perhaps you need to install rgdal first", install "rgdal" with the command:

```
> install.packages('rgdal')
```

Then, run the script again. The following plot should be generated:



And the console should show output results as follows:

```
[1] "n.1:  325"
> print(paste("n.2: ", n.2))
[1] "n.2:  30"
>
>
> print(paste("Mean biomass estimate - Model-based w/ SAR only: ", Muhat.mb1))
[1] "Mean biomass estimate - Model-based w/ SAR only:  49.3490722208938"
> print(paste("Relative SE - Model-based w/ SAR only: ", SE_Muhat.mb1/Muhat.mb1))
[1] "Relative SE - Model-based w/ SAR only:  0.191143807433193"
>
> print(paste("Mean biomass estimate - Model-based w/ SAR and lidar plots: ", Muhat.mb2))
[1] "Mean biomass estimate - Model-based w/ SAR and lidar plots:  50.8277659152922"
> print(paste("Relative SE - Model-based w/ SAR and lidar plots: ", SE_Muhat.mb2/Muhat.mb2))
[1] "Relative SE - Model-based w/ SAR and lidar plots:  0.139095094263627"
```

REFERENCES

Ene, L., E. Naesset, and T. Gobakken. 2016. Simulation-based assessment of sampling strategies for large-area biomass estimation using wall-to-wall and partial coverage airborne laser scanning surveys. Remote Sensing of Environment 176:328-340.

Saarela, S., Holm, S., Grafström, A., S. Schnell, E. Naesset, T. Gregoire, R. Nelson, and G. Ståhl. 2016. Hierarchical model-based inference for forest inventory utilizing three sources of information. Annals of Forest Science 73: 895

Yarie, J., E. Kane, and M. Mack. 2007. Aboveground biomass equations for the trees of interior Alaska. AFES Bulletin 115, University of Alaska-Fairbanks, Fairbanks, Alaska, USA (https://www.uaf.edu/files/snre/B115.pdf) last accessed 12/20/2018.