Statistical computing MATH10093
Computer lab 7

Finn Lindgren

13/3/2019

Summary

In this lab session you will develop code for bootstrap estimation of confidence intervals for probabilities and quantiles.. You will not hand in anything, but you should keep your code script file for later use.

1. Complete lab 6!
2. Initialise lab 7: Open `RStudio` and
   (a) Make sure you have the files from lab 6 in your project folder.
   (b) Open a new R script file for your lab 7 code.
3. Write a function `boot_resample(data)` that takes input

   `data`: A `data.frame` with one observation per row

   The output should be a `data.frame` with a bootstrap sample from the rows of `data`.

   Test the function:

```
data_boot <- boot_resample(TMINallobs)
```

   Compare the full data with the subsample; they should be different:

```
# In R, to view the first rows of the data frames:
head(TMINallobs)
head(data_boot)
```

```
# In Rmd or Rnw, with library(xtable) and code chunk option results="asis":
print(xtable(head(TMINallobs)), size = "\\scriptsize")
print(xtable(head(data_boot)), size = "\\scriptsize")
```

4. We are interested in the probability of freezing weather at Balmoral, i.e. $\theta = \mathrm{P}(Y < 0)$. A simple estimator is $\widehat{\theta} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\{Y_i < 0\}$.

(a) Compute an estimate $\widehat{\theta}$ of θ when the population of interest is when we pick a random day of the year (i.e. similar to CWA Q1 where we did not take seasons into account; here we also ignore the climate change issue).

```
## [1] 0.3073125
```

(b) Compute a vector of 12 monthly estimates $\widehat{\theta}_m$ of the monthly probabilities θ_m, $m = 1, \ldots, 12$, to see the seasonal variation.

```
plot(theta_m_hat)
```

The `tidyverse filter()` function can be used to construct the needed data subsets (based on station, and the also on month). Create a new `data.frame` called `balmoral` that contains only the data from that station, so that you don't need to filter on that everywhere.

```
# Example of filter() use; extract the data from the first of all months, for Balmoral.
# The "pipe operator" "%>%" is helpful for structuring this kind of data wrangling,
# where the result of one operation is used (silently) as the first parameter of
# the next operation.
library(tidyverse) # You only need this line once in your script file
TMINallobs %>%
  filter(ID == "UKE00105875", Day == 1) %>%
  as.data.frame() %>%
  head()
```

5. Construct a bootstrap distribution of θ estimates, $\{\widehat{\theta}^{(1)}, \ldots, \theta^{(J)}\}$, $J = 1000$. Recall your `boot_resample()` function:

```
boot_resample(TMINallobs %>%
                filter(...))
```

The result should look something like this:

```
summary(theta_hat_boot)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.2953  0.3047  0.3071  0.3072  0.3096  0.3205
```

6. Construct a 95% bootstrap confidence interval for θ. See Lecture 6. The `quantile()` function is helpful.

7. Construct bootstrap distributions for the monthly estimates of θ_m, with one set $\{\widehat{\theta}_m^{(1)}, \ldots, \theta_m^{(J)}\}$ for each month. Store the results in a `data.frame` with one column for each month. You can reduce J to 250 to save computing time in the lab.

8. Construct 95% bootstrap confidence intervals for each θ_m. Store the CI:s in a matrix with 12 rows and two columns (one column each for the left and right interval endpoints).

   Plot the results with code similar to this:

```
plot(1:12, theta_m_hat, type = "l")
lines(1:12, theta_m_CI[,1], lty = 2)
lines(1:12, theta_m_CI[,2], lty = 2)
```

```
library(ggplot2)
ggplot(data = data.frame(Month = 1:12,
                         theta_m_hat = theta_m_hat,
                         CI_lower = theta_m_CI[,1],
                         CI_upper = theta_m_CI[,2])) +
  geom_ribbon(aes(x = Month, ymin = CI_lower, ymax = CI_upper), fill = "Grey") +
  geom_line(aes(Month, theta_m_hat))
```

9. We are now interested in the upper 90% quantiles of the daily minimum temperature, i.e. the value θ such that $\mathrm{P}(Y \geq \theta) = 0.9$. Produce bootstrap confidence intervals for the monthly quantiles θ_m.

10. Redo task 8 (and the needed previous tasks) for each day of the year instead of each month. To avoid leap year problems, we let $t = 1, \ldots, 365$ and define each data point to belong to day t if `floor(DecYear * 365) %% 365 == t-1`. Reduce J to, for example, $J = 100$ to save computing time in the lab (this may introduce a lot of Monte Carlo variance in the bootstrap estimates, so larger J should be used when possible!).