Exercise #7: Performing a simple bootstrap with the sea level data

Patrick Applegate, patrick.applegate@psu.edu, Ryan Sriver, rsriver@illinois.edu, and Klaus Keller, klaus@psu.edu

7 September 2015

Learning Goals

After completing this exercise, you should be able to

- perform a very simple bootstrap of time series data by resampling residuals
- identify potential problems in applying this approach to "real" data, such as the sea level data from Jevrejeva et al. (2008)

Introduction

Making estimates of potential future changes, with uncertainties, is a key aspect of environmental risk analysis. For example, the amount by which sea level will rise in the future is clearly an important input for the design of flood protection structures; if sea levels are expected to rise by several meters over the next few centuries, then perhaps dikes must be built higher than if they are expected to rise by a few tens of centimeters.

In this exercise, we'll combine techniques from the two preceding chapters to demonstrate techniques that could be used to estimate future sea level rise. In Exercise #5, we fitted a second-order polynomial to sea level data covering the last ~300 yr. In Exercise #6, we learned how to perform a simple bootstrap analysis using coin flips. Here, we'll apply the bootstrap to estimate our uncertainties in future sea level based on curve fitting.

It's important to note that the sea level rise projections you will produce in this exercise are probably too low, and the associated uncertainties are also incorrect. The approach used in this exercise assumes that

- adjacent residuals aren't correlated with one another
- the scatter in the residuals about the "true," underlying curve doesn't change over time
- the "true" curve on which the "wiggles" are superimposed really is a second-order polynomial

In fact, all three of these assumptions are violated by the sea level data. In particular, this analysis mostly ignores sea level contributions from the ice sheets, and neglects autocorrelation and heteroscedasticity in the residuals. However, this simplified analysis is an important stepping stone to methods that do account for ice sheets and provide better estimates of the actual uncertainty.

Tutorial

In Exercise #6, we performed the bootstrap by resampling our *observations*, but we'll need a different approach for the sea level data. What would happen if we were to simply resample the actual sea level observations from Exercise #5? The following code block reads in the data, plots the original data, and then plots a single bootstrap realization of the data. As you can see from the plots, resampling the sea level data destroys any trend that they contain.

```
# Read in the information from the downloaded file.
sl.data <- read.table('data/gslGRL2008.txt', skip = 14, header = FALSE)

# Extract two key vectors from sl.data.
# t, time in years
# obs.sl, global mean sea level in mm
t <- sl.data[, 1]
obs.sl <- sl.data[, 2]

# Resample the observations in obs.sl.
re.sl <- sample(obs.sl, length(obs.sl), replace = TRUE)

# Make some plots.
par(mfcol = c(1, 2))
plot(t, obs.sl, type = 'l', xlab = 'Time (yr)', ylab = 'Sea level anomaly (mm)')
plot(t, re.sl, type = 'l', xlab = 'Time (yr)', ylab = 'Resampled anomalies (mm)')
```

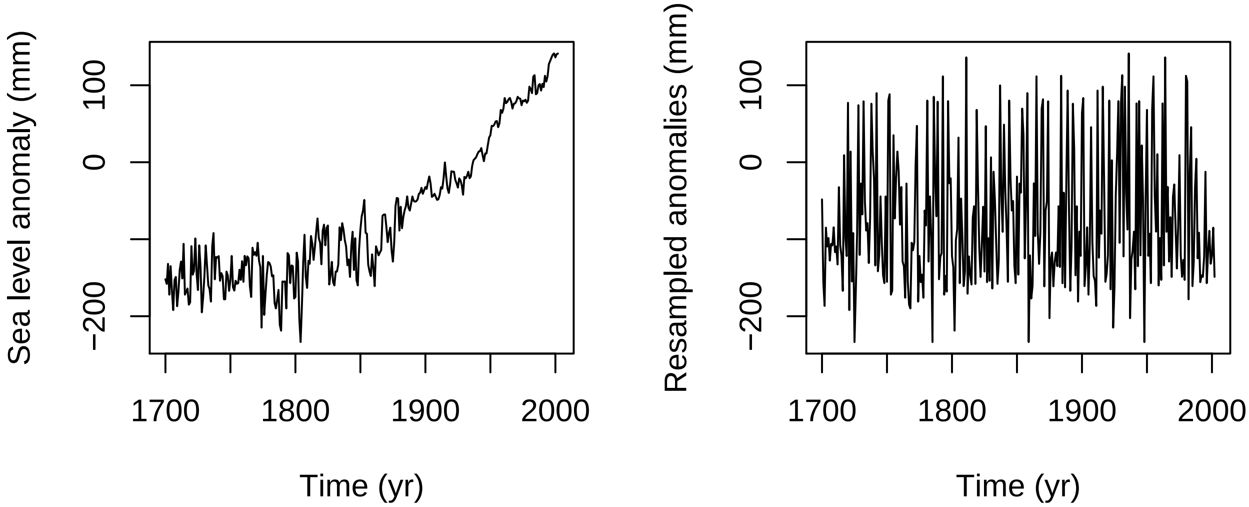

Figure 1: Sea level anomaly data from Jevrejeva et al. (2008; left panel), and the anomalies resampled with replacement (right panel). Simple resampling of the anomaly data destroys any trend or signal in the original data set.

Instead, we'll resample the *residuals* between our best-fit curve from Exercise #5 and the data, then add these resampled residuals back on to the best-fit curve. This approach will preserve the underlying trend in the data and let us learn something about the uncertainties in our estimates of a, b, c, t_0, and future sea level.

Let's examine how to generate one bootstrap replicate using this method and how we would make a projection of future sea level based on that replicate. Your code from Exercise #5 should include a vector that contains the differences between the best-fit second-order polynomial and the data. Assuming that this vector is called `resids`, the following command will create a vector of bootstrapped residuals:

```
# Generate a bootstrap replicate of the residuals.
boot.resids <- sample(resids, length(resids), replace = TRUE)
```

But these residuals only represent the "wiggles," not the long-term trend in the data. Your code should include a vector that contains the past long-term change in sea level based on the best-fit second-order polynomial. Let's assume that this vector is called `best.sl`. We can get a plausible estimate of past sea levels, including both the underlying trend and noise, by adding our resampled residuals onto this vector.

```
# Add the resampled residuals to the best-fit second-order polynomial.
boot.sl <- best.sl+ boot.resids
```

To see why the code block above works, recall that R adds vectors of the same length element-by-element.

Now we fit a curve to the new sea level “data” and extract the best-fit parameters from it, using code like that we saw in Exercise #5:

```
# Fit a second-order polynomial to the residuals+ trend.
boot.start <- c(best.a, best.b, best.c, best.t.0)
boot.fit <- optim(boot.start, sl.rmse, gr = NULL, time = t,
                  sea.levels = boot.sl)

# Extract the new parameter values from boot.fit.
boot.a <- boot.fit$par[1]
boot.b <- boot.fit$par[2]
boot.c <- boot.fit$par[3]
boot.t.0 <- boot.fit$par[4]
```

Note that the code block above assumes that you have a function `sl.rmse` and best-fit parameter values stored in `best.a`, `best.b`, `best.c`, and `best.t.0`.

We can make an estimate of past and future trends in sea level by defining a new, longer, vector of time values and using `boot.a`, `boot.b`, and so on to calculate values of a second-order polynomial:

```
# Define a vector of years extending from 1700 to 2100.
proj.t <- seq(1700, 2100)

# Estimate the trends in past and future sea levels based on boot.a, etc.
proj.sl <- boot.a* (proj.t- boot.t.0)^ 2+ boot.b* (proj.t- boot.t.0)+ boot.c
```

Exercise

Before you proceed, **make sure that you have a working version of the script from Exercise #5.** Open your script in RStudio and `source()` it. Examine the value in `best.rmse`, perhaps by typing `best.rmse` in the Console sub-window and pressing Enter. You should get a value of `[1] 24.4101`. If not, check your script against the instructions in Exercise #5. Once you’re satisfied that your script from Exercise #5 is working properly, **make a copy** by saving it under a different file name.

Part 1: Carrying out the bootstrap. Modify your new script so that it generates `n.boot <- 10^3` bootstrap realizations of past sea levels and fits a second-order polynomial to each realization, using a `for` loop. As in Exercise #6, you’ll need to create vectors *before* the `for` loop to contain the results of your calculations, and then store the results in different elements of these vectors using the `for` loop’s index variable. You’ll also need to create a `matrix()`, perhaps called `proj.sl`, for storing the trends in past and future sea levels based on `boot.a`, `boot.b`, and so forth.

Your script should make the following plots:

1. a two-panel plot with the top panel showing the original sea level data, displayed as `type = 'l'`, with the best-fit second-order polynomial on the same set of axes and a separate panel showing the residuals (as in Exercise #5)
2. a plot showing the second-order polynomials from each of the bootstrap realizations as gray curves (use `matplot()`), the original sea level data in black, and the best-fit second-order polynomial to the original data as a red curve

3. histograms of `boot.a`, `boot.b`, `boot.c`, and `boot.t.0`, with the values from the original data shown as vertical, red lines – make sure to label the axes of your plots in a sensible way and include the units of each quantity
4. a histogram of your sea level estimates in 2100 (these values are stored in `boot.sl`), with the estimate from the original data shown as a vertical, red line

Part 2: Identifying problems with this approach. Apply the `acf()` function to both the original residuals and one bootstrap replicate of the residuals. Plot these autocorrelation functions in two panels, one on top of the other, with sensible labels. On these plots, any bars that extend above or below the blue, dashed lines indicate that there is significant autocorrelation between residuals that have spacings shown on the x-axis.

Questions

1. Compare your sea level rise estimates in 2100 to the four scenarios in NOAA (2012). Which scenario do your sea level rise estimates match most closely? Does this scenario represent an optimistic or a high-end estimate of future sea level rise, according to the NOAA report? How was this scenario constructed?
2. Examine the residuals (Part 1) and the autocorrelation functions (Part 2) for evidence of heteroscedasticity and autocorrelation. Do the residuals, based on the original data and the best fit to them, show evidence of these problems? What about the resampled residuals?

References

Jevrejeva, S., Moore, J. C., Grinsted, A., and Woodworth, P. L., 2008. Recent global sea level acceleration started over 200 years ago? Geophysical Research Letters 35, L08715.

NOAA (National Oceanic and Atmospheric Administration), 2012. Global sea level rise scenarios for the United States National Climate Assessment. NOAA Technical Report OAR CPO-1. Available online at http://cpo.noaa.gov/AboutCPO/AllNews/TabId/315/ArtMID/668/ArticleID/80/Global-Sea-Level-Rise-Scenarios-for-the-United-States-National-Climate-Assessment.aspx