Stat 410 Lab: Scatterplot Smoothing

Dennis Do

10/24/21

Spring migration arrival of birds in the Northeast U.S.: Ecologists believe increases in temperatures (global warming) may cause some bird species to change their migratory patterns and fly north earlier in the spring season than their normal dispersal patterns. The Ornithology Laboratory at Cornell University has been collecting data on the first arrival of many bird species over the past century, as determined by expert bird-watchers in the Finger Lakes region. In this lab, we will study trends in arrival of birds over time using scatterplot smoothers.

The class Canvas site contains data on the first arrival of the purple martin (`puma.txt`). This bird species is believed to have changed migratory behavior, flying north earlier in the spring season in reaction to changes (increase) in southeastern temperatures. Download the data into your working directory. The following code chunk will set it up for the lab.

```
# Data available as a text (ascii) file
# Read in purple martin data
# Note: make sure you have the correct path for the data set
puma = matrix(scan("puma.txt"), ncol=2, byrow=T)
# Set up arrival time and year variables for analysis
arrival=puma[,2]
year=puma[,1]
n = length(arrival)
```

Task 1: Write your own running mean smoother function

Write your own running mean smoother function (RMS) in R. The function should take in three input arguments: vector of data x (year), vector of data y (arrival), and a scalar bandwidth h. The function should output a vector of values $\hat{m}(x)$ on a set of points over the range of x.

Recall that the running mean smoother presented in lecture: Suppose we wish to smooth data pairs $(x_i,\ y_i),\ i = 1, \ldots, n$ to study the relationship y vs. x. The running mean smooth at any value in the support of the independent variable x is

$$\hat{m}(x) = \sum_{i=1}^{n} w_i(x) y_i,$$

where

$$w_i = \frac{I\{|x_i - x| \leq h\}}{card(\{x_i : |x_i - x| \leq h\})}$$

for bandwidth h. Here $I(A)$ denotes the indicator function on a set A and $card(B)$ denotes the cardinality of (number of elements in) a set B. Thus, the predicted value $\hat{m}(x)$ is the mean of all values y_i associated with x_i in the interval $(x \pm h)$ or h units from x.

Code set-up

- You will perform the running mean smooth by looping through 50 points equally spaced in the range of the year data (denoted x above), computing the function $\hat{m}(x)$ in the equation above for each point. The following code fragment gives a suggested structure for your function. Note that since this code chunk is incomplete, the option `eval=FALSE` is set so RMarkdown will not run it. *Remove this option before running your own code.*

```
rms = function(x, y, h){
 ys = 0 # store y-values of the smooth
 xs = seq(min(x), max(x), length = 50) # x-values over which smooth computed
 for(i in 1:length(xs)){
 ys[i] = mean(y[(x>xs[i]-h)&(x<xs[i]+h)])
 }
 list(y=ys, x=xs) # return the x and y values for the smooth
}
```

- Of note, the "CODE THE RMS HERE" piece in the code chunk in the last bullet can be performed in one line using the function $\hat{m}(x)$ in the equation above and subsetting over the appropriate neighborhood. For example, for bandwidth h, consider averaging over the values `y[(x>xs[i]-h) & (x<xs[i]+h)]`.

Task 2: Evaluate bandwidths in running mean smoother

We will plot on the same set of axes the data scatterplot with three running mean smooths with different bandwidths.

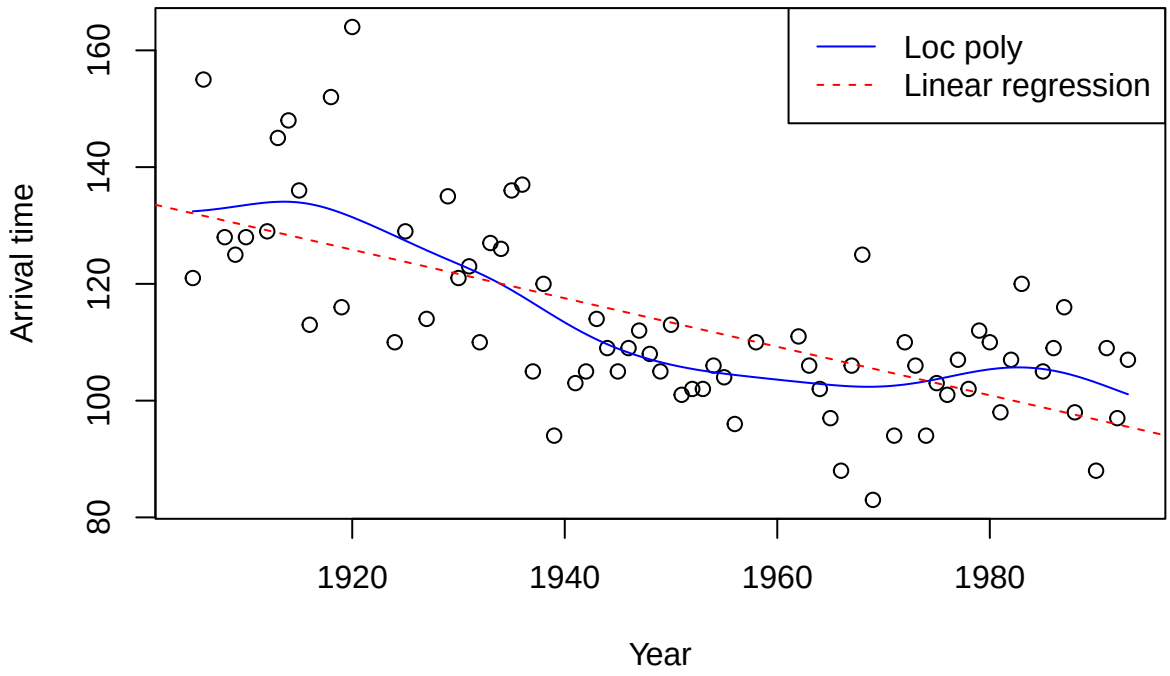
Code set-up

- A reminder that for multiple smooths on a single scatterplot, after drawing the scatterplot using the plot function, use your rms function and the R smoothing functions in the `lines` function. To set different line, line widths, and color types, use the `lty`, `lwd`, and `col` options in the lines function. For example, assuming data variables arrival and year and bandwidth h, the code chunk below plots a local polynomial regression smooth over the scatterplot. You can add more smooths with additional `lines` functions.
- A reminder that for a legend on a plot with multiple lines, use the `legend` function. Again `lty`, `lwd`, and `col` options allow you to specify the corresponding line type and color for the legend. Consider a `title` option as well to place a legend title. Additional smooths can be added similarly within each option. The code chunk below provides an illustration.

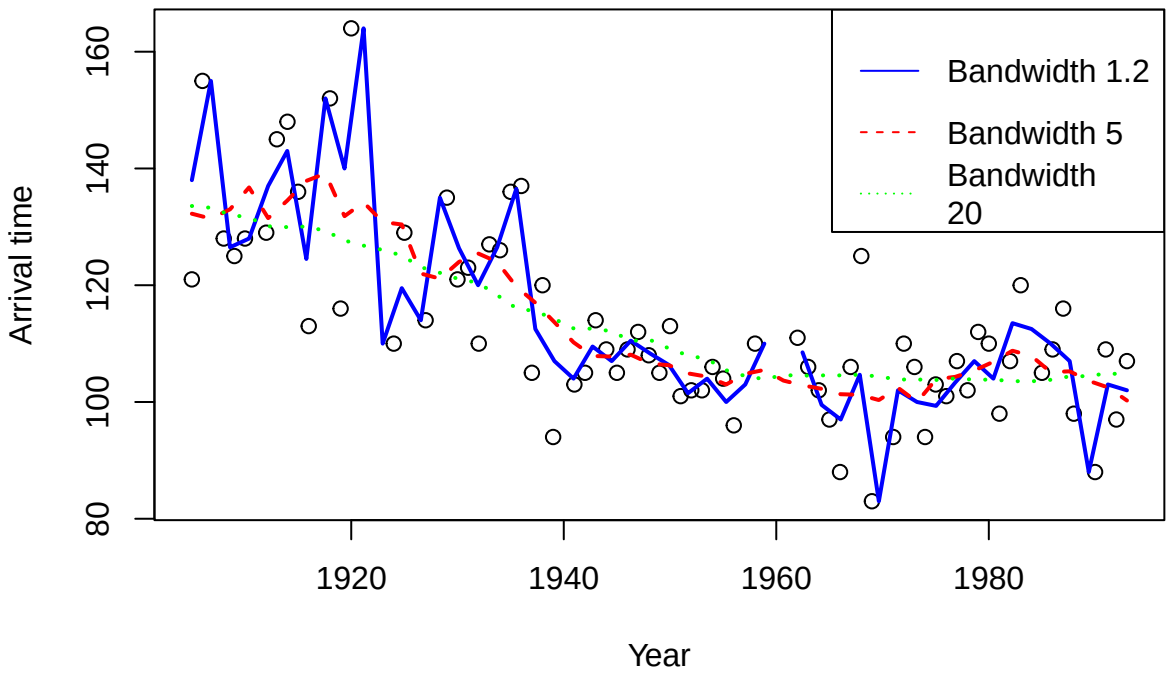
In this code chunk I am using `abline` and linear regression purely for illustration. Note that you will not be overlaying a regression fit in your lab work. I am also using local polynomial regression which we will play with for Exam 2.

```
plot(year, arrival, ylab="Arrival time", xlab="Year")
h = dpill(year, arrival) # optimal plug-in bandwidth for locpoly
# local polynomial regression
lines(locpoly(year, arrival, bandwidth = h), lty = 1, col = 'blue')
# linear regression
# Note that this is for illustration only, you will not use this for the lab
abline(lm(arrival~year), lty = 2, col = 'red')
```

```
legend("topright", legend = c("Loc poly", "Linear regression"),
 lty = c(1,2), col=c("blue","red"))
```




```
plot(year, arrival, ylab="Arrival time", xlab="Year")
lines(rms(year, arrival, 1.2), col="blue", lty=1, lwd=2)
lines(rms(year, arrival, 5), col="red", lty=2, lwd=2)
lines(rms(year, arrival, 20), col="green", lty=3, lwd=2)
legend("topright", legend = c("Bandwidth 1.2", "Bandwidth 5", "Bandwidth
20"),
 lty = c(1:3), col=c("blue","red", "green"))
```



Report the following

- Present, on the same set of axes, the data scatterplot and your running mean smooth function of the data with bandwidths 1.2, 5, and 20. Compute the running mean smooth over 50 points in the range of year x, that is the set `seq(min(x), max(x), length = 50)`. Label the axes, use a different line type and color for each smooth, consider line widths for greater line clarity on the graphic, and include a legend on the plot. You will have three smooths on this scatterplot.
- Discuss the effect of bandwidth (smoothing parameter) on the running mean smooths in bullet item 1 above. Which bandwidth best describes the data and why?

As the value of the bandwidths increase, the smooths get flatter and smoother. While the bandwidth value of 20 is the smoothest of all the curves, the curve with a bandwidth value of 5 seems to be the most accurate representation for the curve. When compared to the optimal bandwidth of 6.85 that was calculated earlier, 5 is close to that value and follows the curve on the other graph.