In this question, you will apply a function to compute a moving average of timeseries data. In the future, this would be useful in writing your own function to convert daily data to monthly data.

A) Observe the user-defined R function "smoother" that takes three inputs:

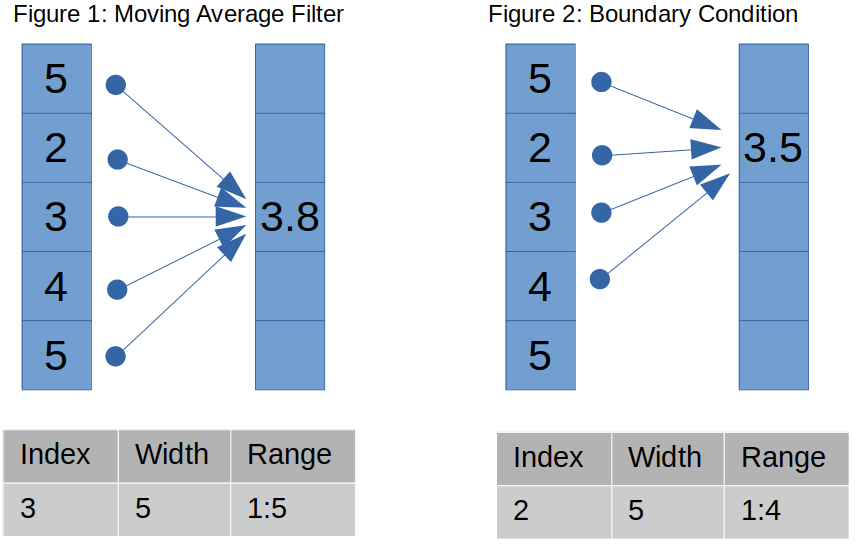
"input": a vector of time series data

"index": the location that we want to compute the mean at

"width": the width of the window that we want to use to compute the mean.

This function returns the mean of all the values of input within the width (centered at index). An example of this is shown in Figure 1 with width=5.

* *Write a loop to iterate through every index of vector “y” (loaded in line 5) and write the output of smoother at that iteration to “my\_smooth”. Use a width of 5. The input data “y” and output data “my\_smooth” should have the same dimensions.*
* *Attach the output image.*
* *Describe the data before/after and how this function works (note this is just test data not rainfall).*



B) Load the rainfall data file into your workspace. Use only the first 8 columns of the data. *Write a loop that smooths the timeseries of each station*. It should loop through each *row* of the data set and compute the smoother function in every column for that row. Use the apply function to do this inside each loop iteration. Hint: apply(DATA\_MATRIX,MARGIN,FUN = smoother,index = i,width = 5). The output data of this apply is a row that should be written into a matrix you name “rain\_smooth”. “rain\_smooth” should be the same size as your input data. This should take about 2.2 minutes to run. The output plot from the test function is just a short section of one of the timeseries.

* *Describe what your loop does.*
* *Attach the plot.*
* *Describe how the data has changed.*
* *Plot the 200-300th rows of the eighth station using any method you want.*

C) Parallelize this loop using a similar method to Assignment 3/4. You may copy/paste your code but will need to make a few changes:

(1) Name the output of foreach "rain\_smooth\_par"

(2) Tell the function what data and function to use in the .export argument. You will need to give it the "smoother" function name and "rain" data. The part inside the loop can just be the apply function you developed in part B. You do not need to assign any output to it as that is done in the .combine argument.

* *Report how long this code took to run with two cores. Did it run faster?*
* *Why is this?*

D) Transfer the entire folder with your completed code to clark (You may want to create a batch script before you transfer or copy your previous one). Instead of syncing to your home folder (/home/YOUR\_PAWPRINT), sync to your class folder (/group/env\_sc\_8001/YOUR\_PAWPRINT/). You will need to change your R script's working directory. Submit your batch job from that folder, and make sure the output data is there (that is where your grade will be).

Use this command (one line) to sync. Do not include a slash after your\_folder

rsync -avhP your\_folder pawprint@clark.rnet.missouri.edu:/group/env\_sc\_8001/pawprint/