Biased correction of SPP: Linear and Quantile Mapping Technique

Presented by:
Begum Rabeya Rushi
Regional Associate,
NASA/SERVIR-HKH
Earth System Science Center | University of Alabama, Huntsville.

Steps

- Making all cell size 0.05 and extent similar
- Biased correction of CHIRPS using gauge data
- Biased correction of SPP using CHIRPS

Table 1. SPPs Used in This Study and Their Specifications

SPP	Provider	Primary Sensor Type	Spatial Resolution	Temporal Resolution	Spatial Coverage	Temporal Coverage
PERSIANN-CCS[a] [*Hong et al.*, 2004]	UCI	Infrared	0.04°×0.04°	3 hourly	37.8°N–40.6°S 28.0°W-56.2°E	2006–2010
CMORPH[b] [*Joyce et al.*, 2004]	NOAA-CPC	Infrared + Passive Microwave	0.25°×0.25°	3 hourly	60°N–60°S 180°E-180°W	1998 to near present
TMPA-RT[c] [*Huffman et al.*, 2007]	NASA	Visual + Infrared + Passive Microwave + Active Microwave	0.25°×0.25°	3 hourly	50°N–50°S 180°E-180°W	1998 to near present
CHIRPS[d] [*Funk et al.*, 2014]	UCSB	Merged Products + In-situ precipitation observations	0.05°×0.05°	daily	50°N–50°S 180°E-180°W	1981 to near present

[a]Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks-Cloud Classification System.
[b]Climate Prediction Center Morphing Technique.
[c]Tropical Rainfall Measuring Mission (TRMM) Multisatellite Precipitation Analysis - Real Time.
[d]Climate Hazards Group InfraRed Precipitation with Station data.

- 3 Hourly to Daily Conversion

Roy et al. 2016

Linear Method Equation

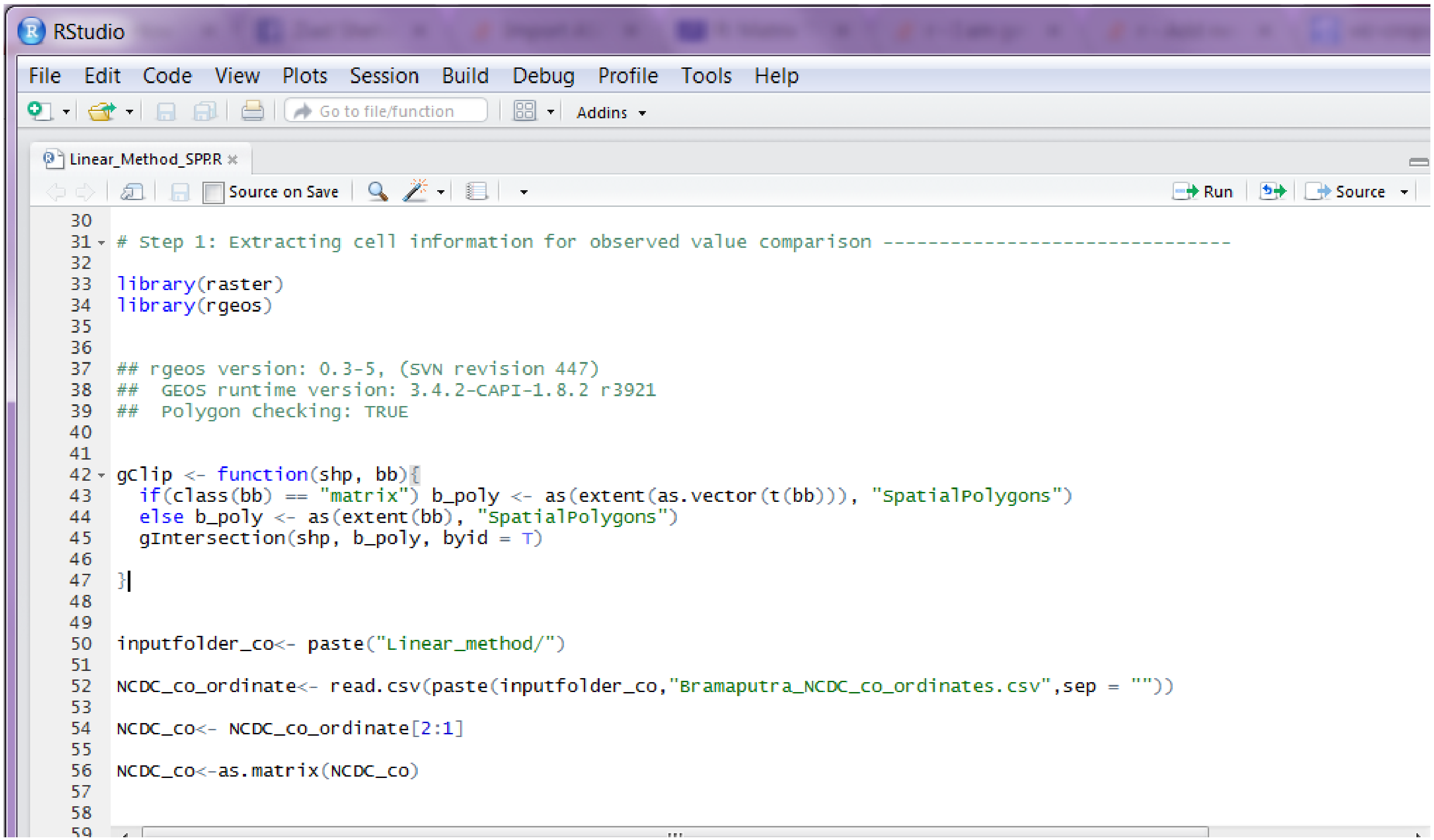
$$P_{\mathrm{cor,m,d}} = P_{\mathrm{raw,m,d}} \times \frac{\mu\left(P_{\mathrm{obs,m}}\right)}{\mu\left(P_{\mathrm{raw,m}}\right)},$$

where $P_{cor,m,d}$ is corrected precipitation at given month m on dth day. $P_{raw,m,d}$ is satellite precipitation at given month m on dth day. $\mu\ (P_{obs,m})$ mean value of CHIRPS precipitation at given month *m and* $\mu\ (P_{raw,m})$ mean value of satellite precipitation at given month *m*

Linear Method: Mean Calculation

- For entire time period, we will get 12 different mean values for each month; for example from 2005 to 2012, we will get 12 mean precipitation raster/matrix for Jan-Dec.
- Each cell will have 12 mean values for each month. If CHIRPS raster is 30x60; for each month, we have to calculate a 30x60 raster.
- We will have 12 raster files for each SPP. Extent and pixel size should be same as CHIRPS.
- Final output would be 30x60 raster

Correction of Extracting Gauge Station Information Area for the River Basin:




```r
# Step 1: Extracting cell information for observed value comparison -------------------------------

library(raster)
library(rgeos)


## rgeos version: 0.3-5, (SVN revision 447)
##  GEOS runtime version: 3.4.2-CAPI-1.8.2 r3921
##  Polygon checking: TRUE


gClip <- function(shp, bb){
  if(class(bb) == "matrix") b_poly <- as(extent(as.vector(t(bb))), "SpatialPolygons")
  else b_poly <- as(extent(bb), "SpatialPolygons")
  gIntersection(shp, b_poly, byid = T)

}


inputfolder_co<- paste("Linear_method/")

NCDC_co_ordinate<- read.csv(paste(inputfolder_co,"Bramaputra_NCDC_co_ordinates.csv",sep = ""))

NCDC_co<- NCDC_co_ordinate[2:1]

NCDC_co<-as.matrix(NCDC_co)
```

Area Extent and Time Extent

```
Year<- c("2015")

Basin<- c("Brahmaputra")
##"Ganges","Indus","Meghna")



inputfolder<- paste(Year,"/chirps-v2.0.",sep = "")
if(!file.exists(inputfolder))dir.create(inputfolder)

## first days of years
date1<- seq(as.Date("2015-1-1"), as.Date("2015-12-31"), "days")
date_seq<- gsub("-",".",date1)
head(date_seq)

date_seq<- as.matrix(date_seq)

no_days<-length(date_seq)
```

Step 2: Extracting cell information for observed value comparison

```
crop_folder<-paste("chirps/test/Brahmaputra/chirps/cropped_chirps/",Year[y],"/",sep="")

if(!file.exists(crop_folder))dir.create(crop_folder)

crop_chirps_file<-paste(crop_folder,date_seq[d],".tif",sep="")

writeRaster(chirps_global_crop, filename=crop_chirps_file, format="GTiff", overwrite=TRUE)

# NCDC_point<-matrix(0,1,2)
# NCDC_point[,1]<-as.numeric(NCDC_co[p,3])
# NCDC_point[,2]<-as.numeric(NCDC_co[p,2])


st_chirps[,2:4]<-data.frame(coordinates(NCDC_co),extract(chirps_global_crop,NCDC_co))
```

Step 3: Mean Monthly Component for CHIRPS and Gauge Data

```
for (m in 1:length(mon)) {
  myList <- list()


  for (y in 1:length(Year)) {


    inputfolder<- paste("C:/chirps/test/Brahmaputra/chirps_obs_daily_comparison/",Year[y],sep = "")
    setwd(inputfolder)

    files <- list.files(path=".",pattern = paste(Year[y],".",sprintf("%02d", mon[m]),".*",sep="")) #path=inputfolder,

    for(i in files) { myList[[length(myList)+1]]<-as.matrix(read.csv(i)) }



  }

  MonthlyMean<-Reduce("+", myList)/length(myList)

  mean_monthly<-as.data.frame(MonthlyMean)

  mean_monthly$Ratio<- mean_monthly$obs_prcp/mean_monthly$chirps_prcp


  mean_monthly_garbage_removal <- mean_monthly[!is.infinite(mean_monthly$Ratio),]

  mean_monthly_final <- mean_monthly_garbage_removal[!is.na(mean_monthly_garbage_removal$Ratio),]
```

Chirps Correction

```
for (y in 1:length(Year)) {

  for (m in 1:length(mon)) {

    inputfolder_chirps<- paste("C:/chirps/test/Brahmaputra/chirps/cropped_chirps/",Year[y],"/",sep = "")
    setwd(inputfolder_chirps)


    files_chirps <- list.files(path=".",pattern = paste(Year[y],".",sprintf("%02d", mon[m]),".*",sep="")) #path=inputfo


     for(i in files_chirps) {

       require(raster)

       directory<- paste(inputfolder_chirps,i,sep = "")

       chirps_daily<- as.matrix(raster(directory))


      monthly_factor<- as.numeric(lapply(paste(mon[m],"monthly_obs_bias_factor", sep = "_"),get))

      corrected_chirps<-data.matrix(chirps_daily*monthly_factor,rownames.force = T)

```

Corrected CHIRPS and SPP: Mean Monthly Matrix

```r
    outputfolder<- paste(Year,"/",Basin[b],"/",sep = "")
    if(!file.exists(outputfolder))dir.create(outputfolder)

    outputfile<- paste(outputfolder,date_seq[d],".precip.chirps.",Basin[b],".txt",sep = "")
    writeRaster(PRcrop25,outputfile,format="ascii",NAflag=-9999,overwrite=TRUE)


   }

   }

# Step 2: Load Data (CHIRPS and SPP) ---------------------------

data_names<-c("chirps","persian")

no_data<-c(1:2)

# Step 3: Make mean monthly matrix for chirps and SPP -------------------------


for (d in 1:length(data_names)) {

for (m in 1:length(mon)){
     myList <- list()
    for (y in 1:length(years)) {


inputfolder<- paste("C:/chirps/test/Brahmaputra/",data_names[d],"/",years[y],"/",sep = "")
```

```r

for(i in files) { myList[[length(myList)+1]]<-as.matrix(raster(i)) }



     }
      MonthlyMean<-Reduce("+", myList) / length(myList)
      mean_monthly<-as.matrix(MonthlyMean)

      rb <- raster(mean_monthly)
      class(rb)

      # replace with correct coordinates
      extent(rb) <- c(82,98,23.75,31.5)


      outputfile<- paste("C:/chirps/test/Brahmaputra/",data_names[d],"/",sprintf("%02d", mon[m]),".mean.monthly.chirps

      writeRaster(rb,outputfile,format="ascii",NAflag=-9999,overwrite=TRUE)

   }


}
```

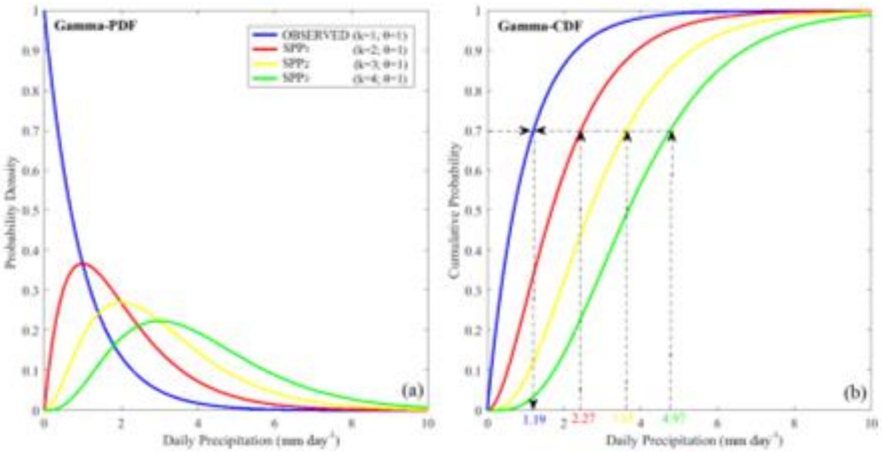
SPP Correction

```
monthly_chirps_factor<-raster(paste("C:/chirps/test/Brahmaputra/","corrected_chirps","/",mon[m],"_corrected_chirps_m
monthly_spp_factor<-raster(paste("C:/chirps/test/Brahmaputra/","persian","/",mon[m],"_persian_monthly_bias_factor.ti

monthly_bias_factor<-overlay(monthly_chirps_factor, monthly_spp_factor, fun=function(x, y){ x/y} )

corrected_spp<-overlay(spp_daily, monthly_bias_factor,fun=function(x, y){ x*y})
```

Quantile Mapping

- Satellite and CHIRPS data covering the same period of record are used to create a "quantile map" of each population
- Gamma Probability Density Function (Gamma-PDF) is considered for precipitation distribution
- The Gamma-PDF is fitted for CHIRPS and satellite products at every grid-point and for all 12 months separately
- The Gamma Cumulative Distribution Function (Gamma-CDF) is used for determining probability associated with precipitation

Quantile Mapping



a)The Gamma Probability Density Function (Gamma-PDF) is applied assuming different shapes
 (k parameter) for each dataset.
b) The respective Gamma Cumulative Distribution Function (Gamma-CDF) for CHIRPS and SPPs is matched
for a Probability (P=0.7) using the Inverse Gamma Function, which is finally used to calculate the bias-corrected
daily satellite precipitation estimates.
From Valdés-Pineda et al. (2016), Open Access Article in Hydrology and Earth System Sciences Discussions

Loading Corrected CHIRPS and SPPs

```r
for (p in 1:length(mon)){
        myList <- list()
        counter<-0
        counter2<-0
        for (y in 1:length(years)) {
 # Loading CHIRPS and SPP --------------------------------------------------
ObsInputfolder<- paste("C:/chirps/quantile/Brahmaputra/chirps/",years[y],"/",sep = "")
SatInputfolder<- paste("C:/chirps/quantile/Brahmaputra/persian/",years[y],"/",sep = "")
files_chirps <- list.files(path=ObsInputfolder,pattern = paste("2015",".",sprintf("%02d", mon[p]),".*",sep="")) #path=
files_sat <- list.files(path=SatInputfolder,pattern = paste("2015",".",sprintf("%02d", mon[p]),".*",sep="")) #path=inp
for(i in files_chirps) {
  counter<-counter+1
  print(counter)
  ObsDirectory<- paste(ObsInputfolder,i,sep = "")
  if (counter==1){
```

Making a 3-D Matrix for Each Month:

```
# Making 3-D Matrix for Each Month Considering All Years ------------------


dim(chirps) <- c(dim(raster(SatDirectory))[1], dim(raster(SatDirectory))[2], counter) # 3-D arrray formation

#dim(chirps) <- c(dim(raster(SatDirectory))[1], dim(raster(SatDirectory))[2], length(files_chirps)) # converting it to
Drizzle<-1 # less than 1 mm rain is considered drizzle
chirps[which(chirps<Drizzle)]<-0

chirps

dim(sat_prcp) <- c(dim(raster(paste(SatInputfolder,j,sep = "")))[1], dim(raster(paste(SatInputfolder,j,sep = "")))[2],
Drizzle<-1 # less than 1 mm rain is considered drizzle
sat_prcp[which(sat_prcp<Drizzle)]<-0

#CHIRPS<-array(3:63, dim=c(3,4,5))


#z <- array(1:60, dim=c(3,4,5))
#z[1,1,1]<-0 ## for testing

#GammaCDF <- array(0, dim=c(3,4,5))
#BCz<- array(0, dim=c(3,4,5)) # for storing bias corrected data

GammaCDF_chirps <- array(0, dim=c(dim(raster(ObsDirectory))[1], dim(raster(ObsDirectory))[2],counter))
GammaCDF_sat<- array(0, dim=c(dim(raster(SatDirectory))[1], dim(raster(SatDirectory))[2], counter2)) # for storing bia:

```

Parameter Identification of Gamma-PDF and Quantile Estimation

```r
if (length(IndexNonZeroCHIRPS)>4 & length(IndexNonZeroSat)>4 & length(unique(NonZeroCHIRPS)) >4 & length(unique(NonZ
  { ### at least more than 2 points are required for curve fitting and more that 3 unique values
#CHIRPSParmsLambda[i,j]<-fitdistr(CHIRPS[i,j,], "gamma")$estimate[1] #lambda OR SHAPE
#CHIRPSParmsTheta[i,j]<-fitdistr(CHIRPS[i,j,], "gamma")$estimate[2] #theta or rate
CHIRPSParmsLambda[m,n]<-fitdistr(NonZeroCHIRPS, "gamma")$estimate[1] #lambda OR SHAPE
CHIRPSParmsTheta[m,n]<-fitdistr(NonZeroCHIRPS, "gamma")$estimate[2] #theta or rate



#GammaParmsLambda[i,j]<-fitdistr(z[i,j,], "gamma")$estimate[1] #lambda
#GammaParmsTheta[i,j]<-fitdistr(z[i,j,], "gamma")$estimate[2] #theta
GammaParmsLambda[m,n]<-fitdistr(NonZeroSat, "gamma")$estimate[1] #lambda
GammaParmsTheta[m,n]<-fitdistr(NonZeroSat, "gamma")$estimate[2] #theta


#GammaCDF[i,j,]<-pgamma(z[i,j,], GammaParmsLambda[i,j], rate = GammaParmsTheta[i,j], log = FALSE)
NonZeroGammaCDF<-pgamma(NonZeroSat, GammaParmsLambda[m,n], rate = GammaParmsTheta[m,n], log = FALSE)
#print(NonZeroGammaCDF)

#BCz[i,j,] <-qgamma(GammaCDF[i,j,],CHIRPSParmsLambda[i,j], CHIRPSParmsTheta[i,j])
GammaCDF_sat[m,n,IndexNonZeroSat]<-qgamma(NonZeroGammaCDF,CHIRPSParmsLambda[m,n], CHIRPSParmsTheta[m,n]) #inverse
}else {
  print(NonZeroSat)
  GammaCDF_sat[m,n,IndexNonZeroSat]<- NonZeroSat ## no bias correction is done if only 2 points are available


}

```

Acknowledgement:

- Professor Juan Valdes and members of his research group
- Emily, Africa and Lee