INTRODUCTION TO R

Gulilat Tefera Diro

diro@sca.uqam.ca

ESCER Centre, Department of Earth and Atmospheric Sciences, UQAM

Université du Québec à Montréal
Département des sciences de la Terre et de l'atmosphère

Outline:

- Introduction
- Reading/Writing NetCDF data with R
- How to manipulate data with R
  - Exploratory data analysis
  - Compute climatology
- Basic Graphics with R
  - Contour map eg. JJA mean rainfall over Africa.
  - Line plot/Boxplot/histogram for a point or over a given area

Introduction

- R is a software for analyzing data and graphical display
- It's freely downloadable and is available for Linux, Mac and Windows
- The main website for R is http://www.r-project.org
- It's well documented, and supported by a large community
- R comes complete with its base libraries and often some recommended packages. But you can extend your version of R by installing additional packages that are relevant for your analysis eg. Library for Netcdf interface.

Installing Packages

- In Linux
  - From the command line:
    - R CMD INSTALL package_name.tgz
  - Or with in R invironment use the install.packages() command

    R> install.packages('package_name')
    - N.B. R is Case sensitive
- In Windows
  - From the **Packages** Menu, choose **Install packages**
    - From the list of sources, choose the nearest place and then select the package name from the list of packages

Opening and Quiting: in Linux

- To open R, just type R I.e.

  $>R

then it starting with a screen that might look like

R version 2.13.1 (2011-07-08)

Copyright (C) 2011 The R Foundation for Statistical Computing

ISBN 3-900051-07-0

Platform: x86_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

……

>

- To quit from R environment, just type q() i.e.

$> q()

$>Save workspace image? [y/n/c] : n

Libraries we need

- One of the library packages that's useful for reading netcdf data is 'ncdf'
- More often we want to display the coastal map when we plot the contour map and for this we need R libraries such as 'maps' and 'mapproj'
- The default version of R does not have the above libraries. To install packages use this command

  >install.packages("ncdf")

- And once installed, we have to load them by using the command:

  > library("ncdf")

Some useful R commands

- Assignment statement in R

  a<-b or a=b

- Create a vector eg.

  A <- c(1,2,3,4,5,6,7,8,9,10) or A <-1:10

  B <- seq(1,10,2)

- Create an empty 3X4 matrix

  ab <-array(NA,c(3,4))

- Looping

  for (i in c(1:n)) { }

- Branching

  if (a ==b) { }

- apply function

  eg. ab <-apply(dat,c(1,2),mean)

Reading from file

To open a netcdf file, use *open.ncdf* function but first let's link the data (in the unix command line)I.e.

```
$ ln -sf /skynet3_exec2/diro/data/cru_1990_2009_pre.nc
```

nd in R session try this

```
$> crudata = open.ncdf("cru_1990_2009_pre.nc")
```

- You can see the fields inside crudata by typing:

- $> print(crudata)

- To get fields like longitude,latitude,rainfall use get.var.ncdf function i.e.

```
lon <- get.var.ncdf(crudata,"lon")
lat <- get.var.ncdf(crudata,"lat")
timearr <- get.var.ncdf(crudata,"time")
crutpr <-  get.var.ncdf(crudata,"pre")
```

- It's good to see the dimension of the variable you loaded, e.g.

To compute seasonal mean

- To create a vector

  tind = (1:dim(timearr))

- Create an empty 3-dimensional matrix for longitude, latitude, and time

  jjapr <-array(NA,c(length(lon),length(lat),length(tind)/12))

- Take the JJA average for every longitude and latitude points and store it to the empty matrix that we prepare

```
i=1;sts=6
for (j in seq(sts,length(tind),12)){
  jjapr[,,i] <-apply(crutpr[,,j:(j+2)],c(1,2),mean)
  i<-i+1
}
```

- The new matrix is having a time dimension of length(tind)/12.

Computing JJA Climatology

- Take the JJA time series and apply the mean function over every grid points i.e.

jjapr_cl = apply(jjapr,c(1,2),mean)

- To extract data over a certain region

alon = lon[lon > 0 & lon <55]
alat = lat[lat > -35 & lat <35]
atpr = jjapr_cl[lon> 0 & lon <55,lat > -35 & lat <35]

- *to extract a time series (from the jja time series) over a sub-region*

gntpr = jjapr[lon> 5 & lon <15,lat > -5 & lat <10,]

Writing/Saving in NetCDF

- Define the Coordinates

```
dim1=dim.def.ncdf(”Longitude","Degrees",as.double(alon))
dim2=dim.def.ncdf(”Latitude","Degrees",as.double(alat))
```

- Define the Empty NetCDF Variable

```
varz=var.def.ncdf("CRU precipitation","mm",list
(dim1,dim2),-999,longname=" JJA precipitation climatology")
```

- Define the file name and put the data in

```
nccru=create.ncdf("examplecru.nc",varz)
put.var.ncdf(nccru,varz,atpr)
```

- Close the NetCDF variable

```
close.ncdf(nccru)
```

Contour map with R

Define your colorbar

require(RColorBrewer)

```
rgtc=colorRamp(c("white","yellow","lightgreen","green","lightblue","blue"))
```

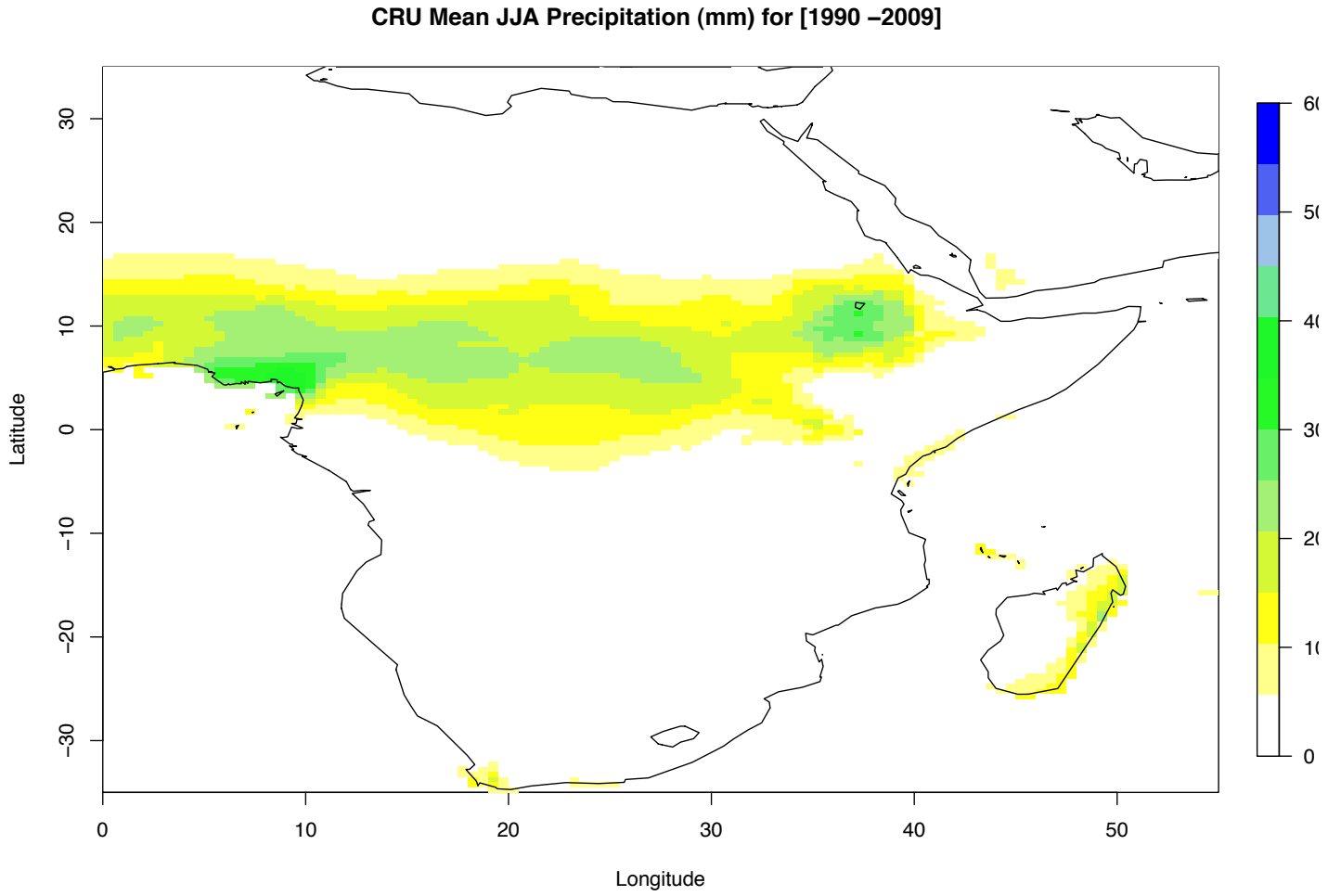
If you want to save the figure in file set

```
postscript(paste("jjacruprp.eps"))
```

then use image.plot and plot.axes i.e.

```
mage.plot(alon,alat,atpr,zlim=c(0,450),col=rgb(rgtc(seq
(0,1,length=12)),max=255),xlab='Longitude',ylab='Latitude')
plot.axes=map('world',interior = F,add=TRUE)
title("CRU Mean JJA Precipitation (mm) for [1990 -2009] ")
dev.off()
```

CRU JJA climatology (1990-2009)



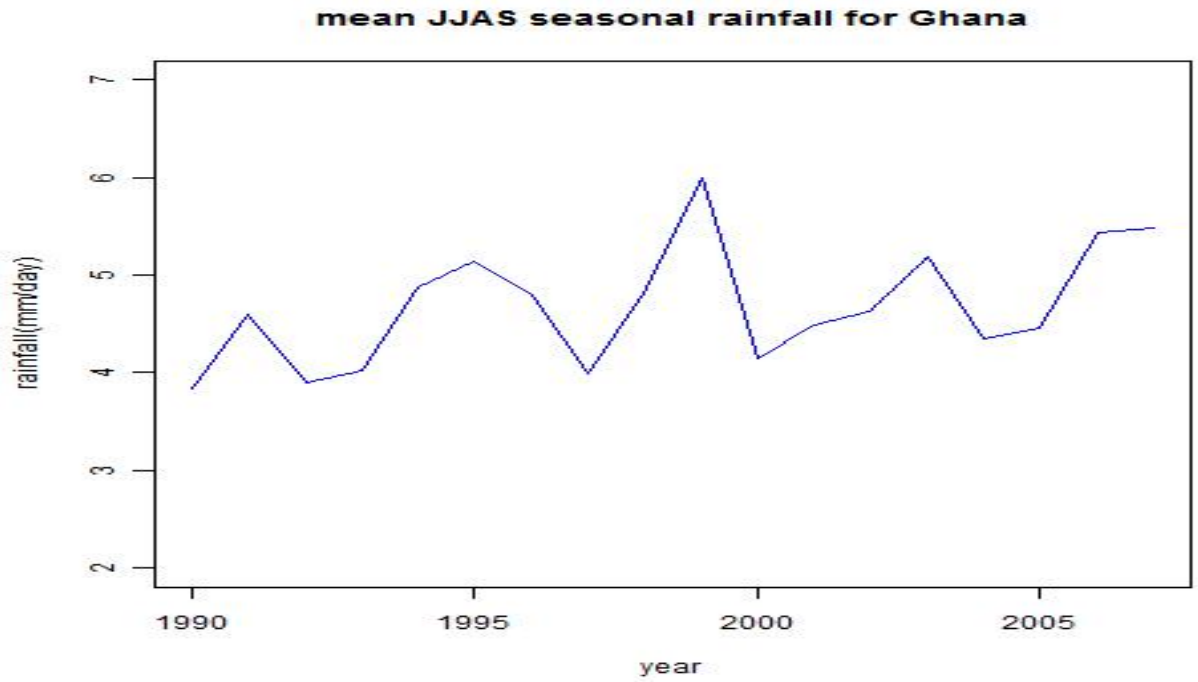
Summary: Exploratory data analysis

```
gnlnavpr=apply(gntpr,c(2,3),mean)
gnavtpr = apply(gnlnavpr,c(2),mean)
mean(gnavtpr[,1])
sd(gnavtpr[,1])
var(gnavtpr[,1])
summary(gnavtpr[,1])
```

```
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 3.833   4.201   4.614   4.677   5.077   5.994
```

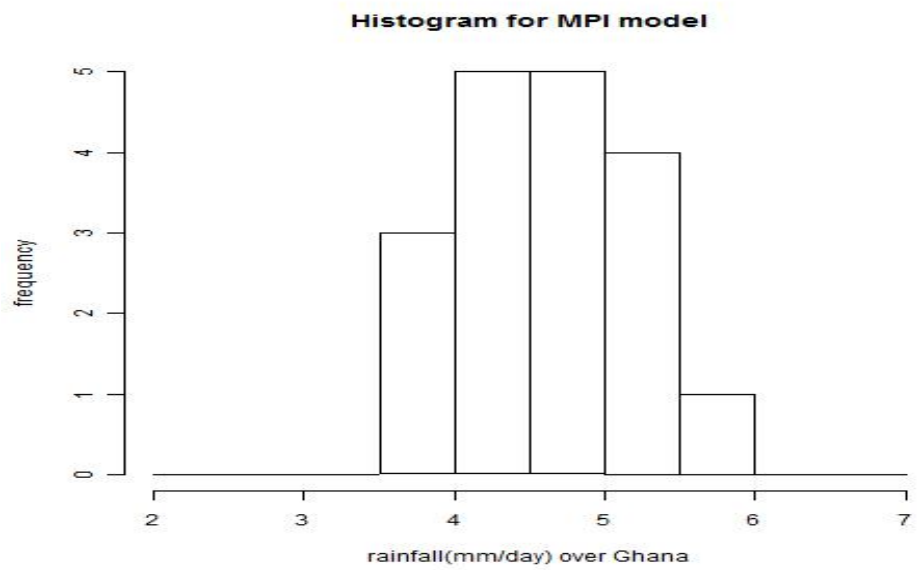
Basic Graphics with R: Line plot

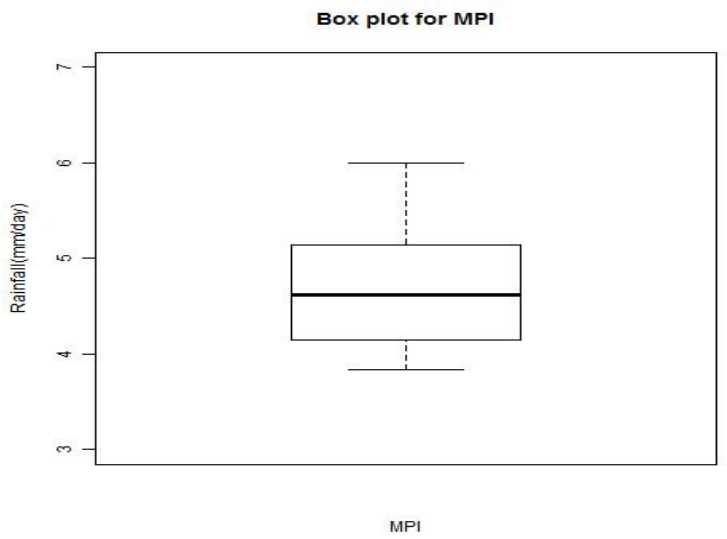
```
plot(yr,gnavtpr[,1],ylim=c
  (2,7),'l',col='blue',xlab='year',ylab='rainfall(mm/
  day)',main='mean JJAS seasonal rainfall ')
```



Distributions (Histogram and Boxplot)

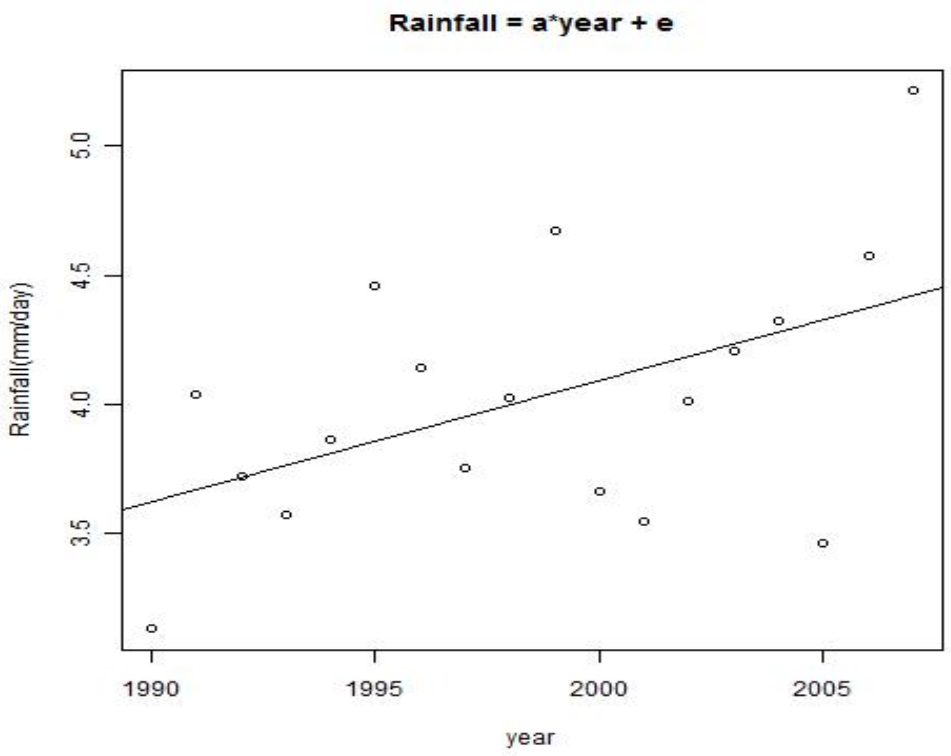
```
hist(gnavtpr[,1],breaks=seq(2,7,0.5),xlab="rainfall(mm/
  day)",ylab="frequency", main="Histogram for MPI model")
boxplot(gnavtpr[,1],ylim=c(3,7),xlab='MPI',ylab='Rainfall
  (mm/day),main='Box plot for MPI')
```





Linear trend: Regression with R

```
plot(gnavtpr[,4]~yr,xlab="year",ylab="Rainfall(mm/day)",main="Rainfall = a*year + e")
abline(lm(gnavtpr[,4]~yr))
```



summary(lm(gnavtpr[,4]~yr))

```
Call:
lm(formula = gnavtpr[, 4] ~ yr)

Residuals:
     Min       1Q   Median       3Q      Max
-0.86864 -0.19305  0.01748  0.22959  0.79346

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -90.11490   41.02004  -2.197   0.0431 *
yr            0.04710    0.02053   2.295   0.0356 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4518 on 16 degrees of freedom
Multiple R-squared: 0.2476,     Adjusted R-squared: 0.2006
F-statistic: 5.266 on 1 and 16 DF,  p-value: 0.0356
```