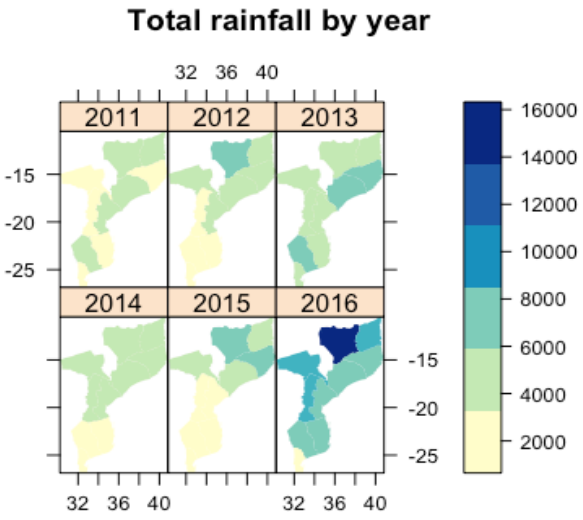
HW 4 R

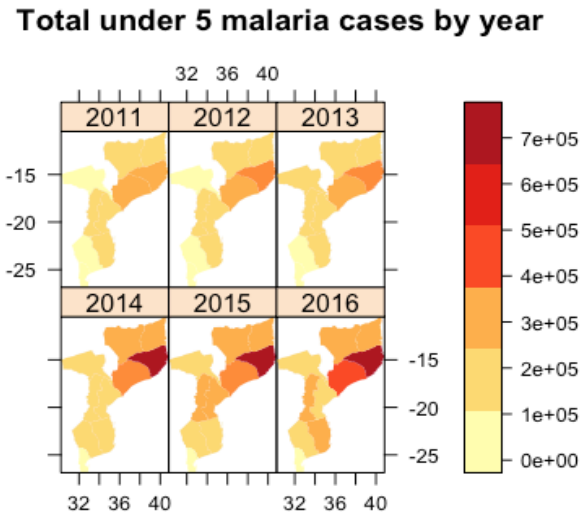
Melissa Lowe with assistance from Emma Jones, Kaci Pickett, and Jammay Winslow

10/3/2018

Maps:

This code is partially from class, partially from Jimmy and partially from Emma: (Thanks peeps)





CODE:

Maps

```{r, echo=F, message=F, warning=F}

op <- par()

options(width=80)

emptyenv()

rm(list=ls())

### MAPPING PACKAGES ###

#install.packages(c("RColorBrewer", "sp", "maptools", "lattice", "latticeExtra", "rgdal"))

library(RColorBrewer)

library(sp)

library(maptools)

library(lattice)

library(latticeExtra) # For layer()

library(rgdal)

library(RColorBrewer)

library(classInt)

library(gridExtra)

library(grid)

```

Now we need to aggregate the malaria and climatic data over the districts to create a set of statistics for each district. Conveniently, this aggregation I perform results in the rownames of the dataframe being the district codes, which is required for creating what is called a spatial polygons dataframe (we'll create this below).

```{r, echo=FALSE, message=T, warning=T}

# Note: because I started an Rproject, I can use relative files paths

# I just need to use './...' to access other folders within this project now

all2 <- read.csv("~/Desktop/MS YEAR 2/S1/R Class/R Class Notes/R-Class/MozSyntheticMalaria.csv")

all2$cpt <- (all2$malaria/(all2$Population_UN*all2$u5weight))*1000

all2.2 <- subset(all2, Epiyear < 2017)


```


```{r, echo=FALSE}

table(all2.2$Province)


all2.2 <- subset(all2.2, all2.2$Province != "MAPUTO CIDADE")


all2.2$admin_1 <- all2.2$Province


# Create means by province and epiyear

malProv <- tapply(all2.2$malaria, list(all2.2$Province, all2.2$Epiyear), mean)

rainProv <- tapply(all2.2$rainTot, list(all2.2$Province, all2.2$Epiyear), mean)


```


```{r, echo=FALSE}
```

#want to know statistics for district code and year, rows are distcode and columns are years

cpt <- as.data.frame(tapply(all2.2$malaria, list(all2.2$admin_1, all2.2$Epiyear), sum))

colnames(cpt) <- c("cpt10", "cpt11", "cpt12", "cpt13", "cpt14", "cpt15", "cpt16")

rainTot <- as.data.frame(tapply(all2.2$rainTot, list(all2.2$admin_1, all2.2$Epiyear), sum))

colnames(rainTot) <- c("rain10", "rain11", "rain12", "rain13", "rain14", "rain15", "rain16")

tavg <- as.data.frame(tapply(all2.2$tavg, list(all2.2$admin_1, all2.2$Epiyear), mean))

colnames(tavg) <- c("t10", "t11", "t12", "t13", "t14", "t15", "t16")

allStats <- as.data.frame(cbind(cpt, rainTot, tavg))

#rownames(allStats) #notice that the row names are the province names

#colnames(allStats)

allStats <- allStats[-6,]

```

```{r, echo=FALSE}

Let's load the district-level administrative shape file for Mozambique.

 # read in the Moz shape file for districts

poly1 <- readShapePoly('~/Desktop/MS YEAR 2/S1/R Class/R Class Notes/R-Class/Moz_admin2.shp', IDvar="DISTCODE")

#this is the polygon file. it's been developed with district codes. RStudio helps recall how to get things out of lists. They're shape files as R stores them

poly2 <- readShapePoly('~/Desktop/MS YEAR 2/S1/R Class/R Class Notes/R-Class/mozambique_admin1.shp', IDvar="admin_1")

#this is the polygon file. it's been developed with province codes. RStudio helps recall how to get things out of lists. They're shape files as R stores them

row.names(allStats) <- c("Cabo Delgado", "Gaza", "Inhambane", "Manica", "Maputo", "Nampula", "Nassa", "Sofala", "Tete", "Zambezia")

Now let's combine the `allStats` dataframe we created above with the `poly2` shapefile so that we can plot the statistics on the map.

polydat <- SpatialPolygonsDataFrame(poly2, allStats)

```

this code is paritally from class, partially from Jimmy and partially from Emma: (Thanks peeps)

```{r, echo=F, message=T, warning=T, fig.height=8, fig.width=8}

par(mfrow=c(2,1))

 # MULTIPLE #

tempPal <- brewer.pal(n = 7, name = "YlOrRd")

rainPal <- brewer.pal(n = 7, name = "YlGnBu")

my.palette <- brewer.pal(n = 7, name = "OrRd")

```
trellis.par.set(sp.theme(regions=list(col = rainPal)))

p2 <- spplot(polydat, c("cpt11", "cpt12", "cpt13", "cpt14", "cpt15", "cpt16"),
     names.attr = c("2011", "2012", "2013", "2014", "2015", "2016"),
     colorkey=list(space="right"), scales = list(draw = TRUE),
     main = "Total under 5 malaria cases by year",
     as.table = TRUE, col.regions = tempPal, col='transparent', cuts=5)
#invisible(dev.off())



p1 <- spplot(polydat, c("rain11", "rain12", "rain13", "rain14", "rain15", "rain16"),
     names.attr = c("2011", "2012", "2013", "2014", "2015", "2016"),
     colorkey=list(space="right"), scales = list(draw = TRUE),
     main = "Total rainfall by year",
     as.table = TRUE, col.regions = rainPal, col='transparent',  cuts=5)
#invisible(dev.off())

grid.arrange(p1, p2, ncol = 2)
```