BIOS6640:R Project

Ana Babinec

October 26, 2018

```
setwd("~/BIOS 6640/Final R Project")
library(RColorBrewer)
library(sp)
library(maptools)
```

```
## Checking rgeos availability: FALSE
##      Note: when rgeos is not available, polygon geometry     computations in maptools depend on gpcli
##      which has a restricted licence. It is disabled by default;
##      to enable gpclib, type gpclibPermit()
```

```
library(lattice)
library(latticeExtra) # For layer()
library(rgdal)
```

```
## rgdal: version: 1.3-6, (SVN revision 773)
##  Geospatial Data Abstraction Library extensions to R successfully loaded
##  Loaded GDAL runtime: GDAL 2.2.3, released 2017/11/20
##  Path to GDAL shared files: C:/Users/aebabinec/Documents/R/win-library/3.5/rgdal/gdal
##  GDAL binary built with GEOS: TRUE
##  Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ_VERSION: 493]
##  Path to PROJ.4 shared files: C:/Users/aebabinec/Documents/R/win-library/3.5/rgdal/proj
##  Linking to sp version: 1.3-1
```

```
##Load the data##
getwd()
```

```
## [1] "C:/Users/aebabinec/Documents/BIOS 6640/Final R Project"
```

```
alldata<-read.csv("MozSyntheticMalaria.csv")
```

```
alldata2<-subset(alldata, Epiyear< 2017)

##create under 5 cases variable##
alldata2$under5<-(alldata2$malaria/(alldata2$u5weight*alldata2$Population_UN)*1000)
#gathers together weeks
cpt <- as.data.frame(tapply(alldata2$under5, list(alldata2$Province, alldata2$Epiyear), sum))
colnames(cpt) <- c("cpt10", "cpt11", "cpt12", "cpt13", "cpt14", "cpt15", "cpt16")

rainTot <- as.data.frame(tapply(alldata2$rainTot, list(alldata2$Province, alldata2$Epiyear), sum))
colnames(rainTot) <- c("rain10", "rain11", "rain12", "rain13", "rain14", "rain15", "rain16")

tavg <- as.data.frame(tapply(alldata2$tavg, list(alldata2$Province, alldata2$Epiyear), mean))
colnames(tavg) <- c("t10", "t11", "t12", "t13", "t14", "t15", "t16")

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


##take out City province thats duplicate##
allStats2<-allStats[-6,]
```

```
rownames(allStats2)<-c("Cabo Delgado", "Gaza", "Inhambane", "Manica", "Maputo", "Nampula", "Nassa", "So
```

```
#library(rgdal)
#poly1<-readOGR( "mozambique_admin1.shp")

poly1<-readShapePoly('mozambique_admin1.shp', IDvar="NAME1")
plot(poly1)
```

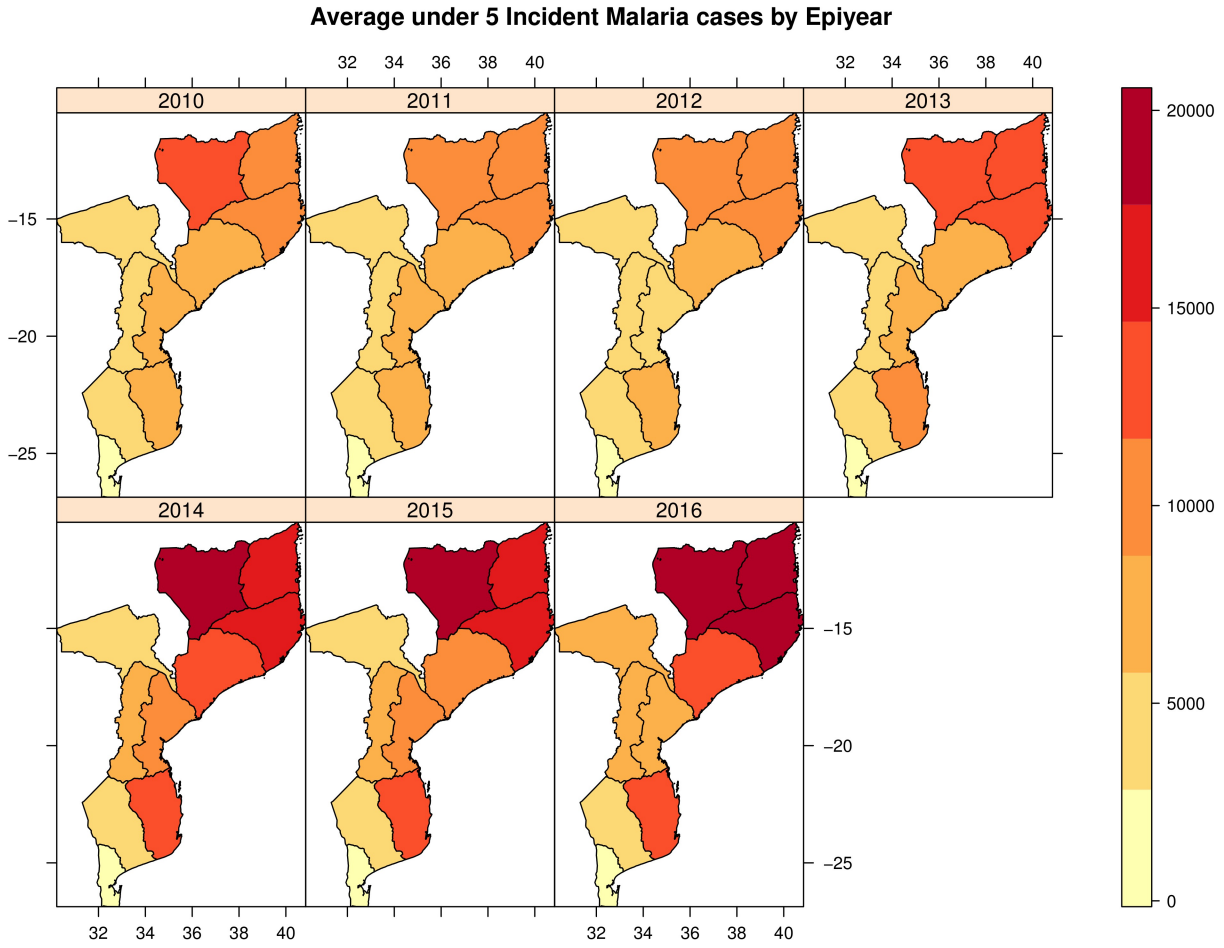
```
polydat <- SpatialPolygonsDataFrame(poly1, allStats2)


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

library(classInt)
```

```
## Loading required package: spData
```

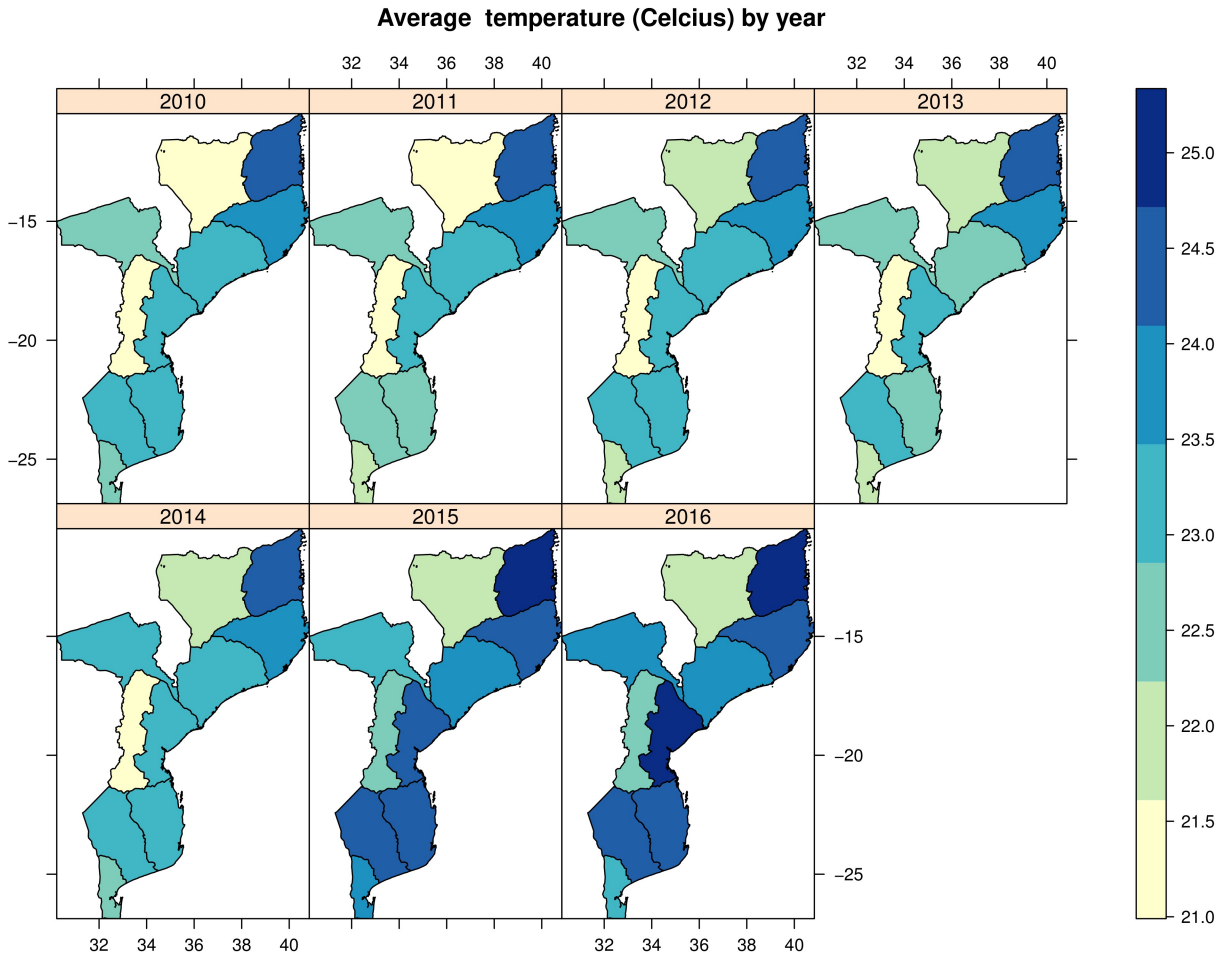
```
## To access larger datasets in this package, install the spDataLarge
## package with: `install.packages('spDataLarge',
## repos='https://nowosad.github.io/drat/', type='source'))`
```

```
spplot(polydat, c("cpt10","cpt11", "cpt12", "cpt13", "cpt14", "cpt15", "cpt16"),
       names.attr = c("2010","2011", "2012", "2013", "2014", "2015", "2016"),
       colorkey=list(space="right"), scales = list(draw = TRUE),
       main = "Average under 5 Incident Malaria cases by Epiyear",
```

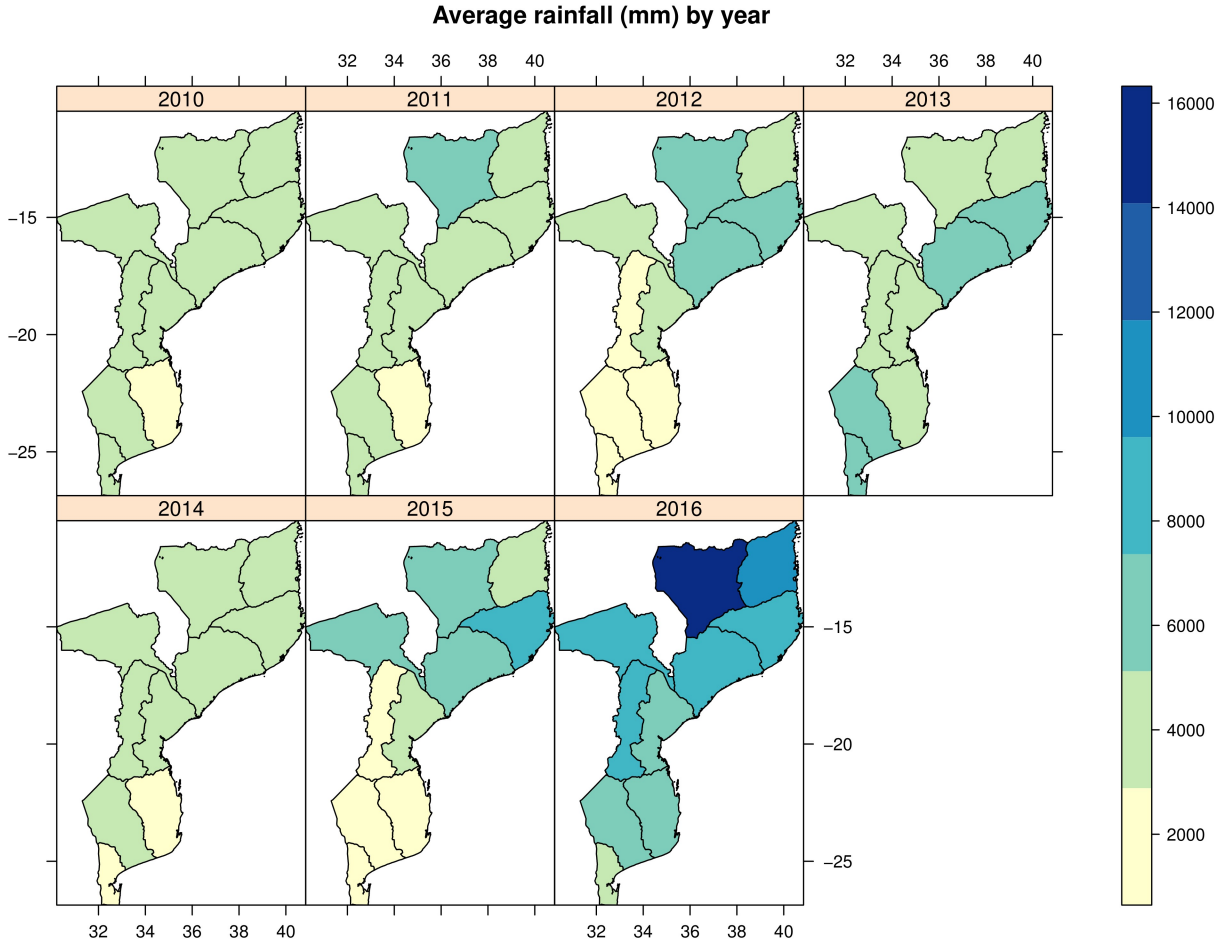
```
       as.table = TRUE, col.regions = tempPal, col="black", cuts=6)
```




```
spplot(polydat, c("t10","t11", "t12", "t13", "t14", "t15", "t16"),
       names.attr = c("2010","2011", "2012", "2013", "2014", "2015", "2016"),
       colorkey=list(space="right"), scales = list(draw = TRUE),
       main = "Average  temperature (Celcius) by year",
       as.table = TRUE, col.regions = rainPal, col="black", cuts=6)
```




```
spplot(polydat, c("rain10", "rain11", "rain12", "rain13", "rain14", "rain15", "rain16"),
       names.attr = c("2010","2011", "2012", "2013", "2014", "2015", "2016"),
       colorkey=list(space="right"), scales = list(draw = TRUE),
       main = "Average rainfall (mm) by year",
       as.table = TRUE, col.regions = rainPal, col="black", cuts=6)
```




```
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:latticeExtra':
##
##     layer
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
```

```
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(stats)

data.lag <- alldata2 %>%
  group_by(DISTCODE) %>%
  mutate(raintot2 = lag(rainTot, 2), ## 2 week lag
         raintot4 = lag(rainTot, 4), ## 4 week lag
         raintot8 = lag(rainTot, 8), ## 8 week lag
         tavg2= lag(tavg, 2), ## 2 week lag
         tavg4 = lag(tavg, 4), ## 4 week lag
         tavg8 = lag(tavg, 8)) ## 8 week lag
```

```
ggplot(data = data.lag) +
  geom_smooth(mapping = (aes(x = Epiweek, y = under5, colour="Under 5 malaria")), linetype="longdash") +
  geom_smooth(mapping = aes(x = Epiweek, y = rainTot, colour="No lag"))+
  geom_smooth(mapping = aes(x = Epiweek, y = raintot2, colour="Lag 2 weeks")) +
  geom_smooth(mapping = aes(x = Epiweek, y = raintot4, colour="Lag 4 weeks")) +
  geom_smooth(mapping = aes(x = Epiweek, y = raintot8, colour="Lag 8 weeks")) +
  scale_colour_manual("", values = c("Under 5 malaria" ="black", "No lag" ="orange","Lag 2 weeks" ="purp
   facet_wrap(~ Region, nrow=2) +
  scale_y_continuous(sec.axis = sec_axis(~.*2, name = "Total rain (mm) with various lag-effect")) +
  labs(x = "Epidemiology week", y = "Cases under 5 per 1,000")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

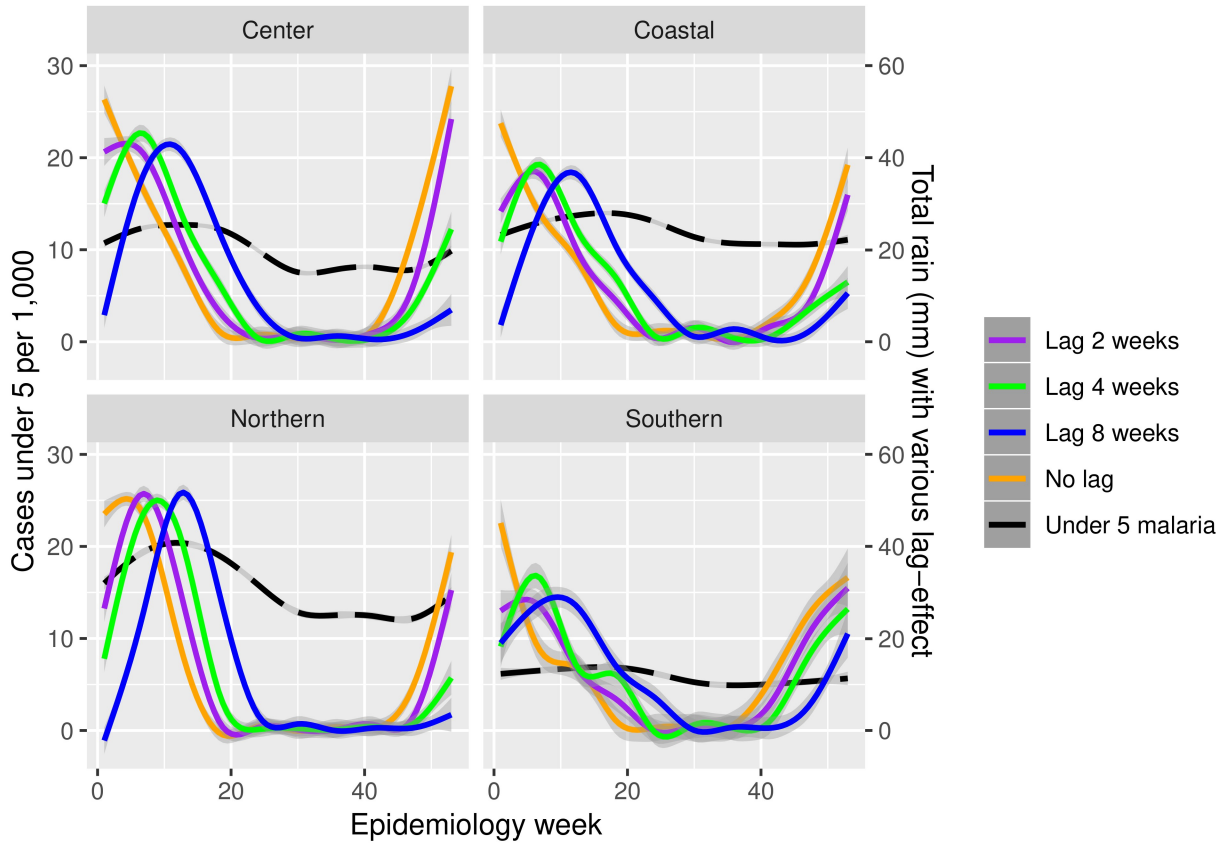
```
## Warning: Removed 284 rows containing non-finite values (stat_smooth).
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 568 rows containing non-finite values (stat_smooth).
```

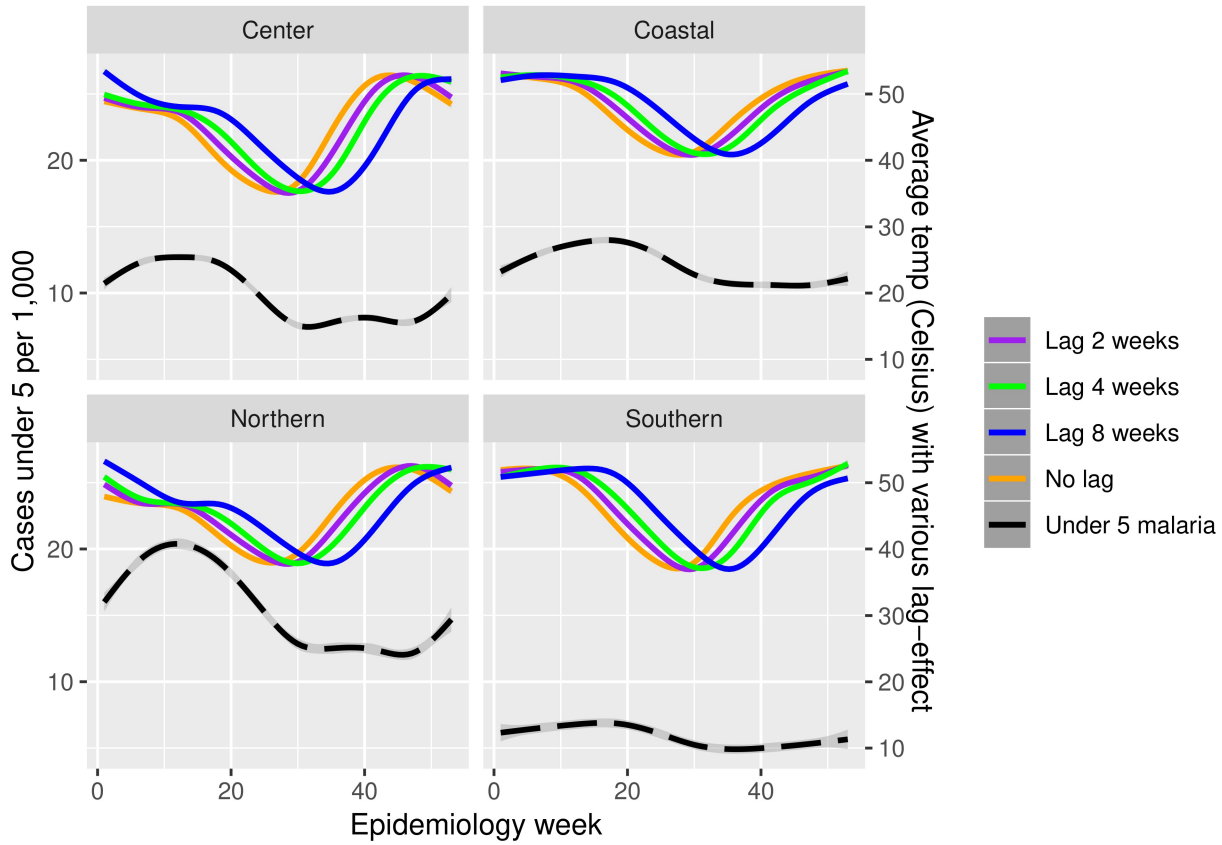
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 1136 rows containing non-finite values (stat_smooth).
```




```
ggplot(data = data.lag) +
  geom_smooth(mapping = (aes(x = Epiweek, y = under5, colour="Under 5 malaria")), linetype="longdash") +
  geom_smooth(mapping = aes(x = Epiweek, y = tavg, colour="No lag"))+
  geom_smooth(mapping = aes(x = Epiweek, y = tavg2, colour="Lag 2 weeks")) +
  geom_smooth(mapping = aes(x = Epiweek, y = tavg4, colour="Lag 4 weeks")) +
  geom_smooth(mapping = aes(x = Epiweek, y = tavg8, colour="Lag 8 weeks")) +
  scale_colour_manual("", values = c("Under 5 malaria" ="black", "No lag" ="orange","Lag 2 weeks" ="pur
  facet_wrap(~ Region, nrow=2) +
  scale_y_continuous(sec.axis = sec_axis(~.*2, name = "Average temp (Celsius) with various lag-effect"))
  labs(x = "Epidemiology week", y = "Cases under 5 per 1,000")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## Warning: Removed 284 rows containing non-finite values (stat_smooth).
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## Warning: Removed 568 rows containing non-finite values (stat_smooth).
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## Warning: Removed 1136 rows containing non-finite values (stat_smooth).
```




```
##Correlation plot of lagged variables with under5##
limdat <- data.lag[c(27,28,29,30,31,32,33,12,13)]
cormat <- round(cor(limdat, use="complete.obs"),2)

# Get lower triangle of the correlation matrix
get_lower_tri<-function(cormat){
  cormat[upper.tri(cormat)] <- NA
  return(cormat)
}
# Get upper triangle of the correlation matrix
get_upper_tri <- function(cormat){
  cormat[lower.tri(cormat)]<- NA
  return(cormat)
}

library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
melted_cormat <- melt(cormat)
head(melted_cormat)
```

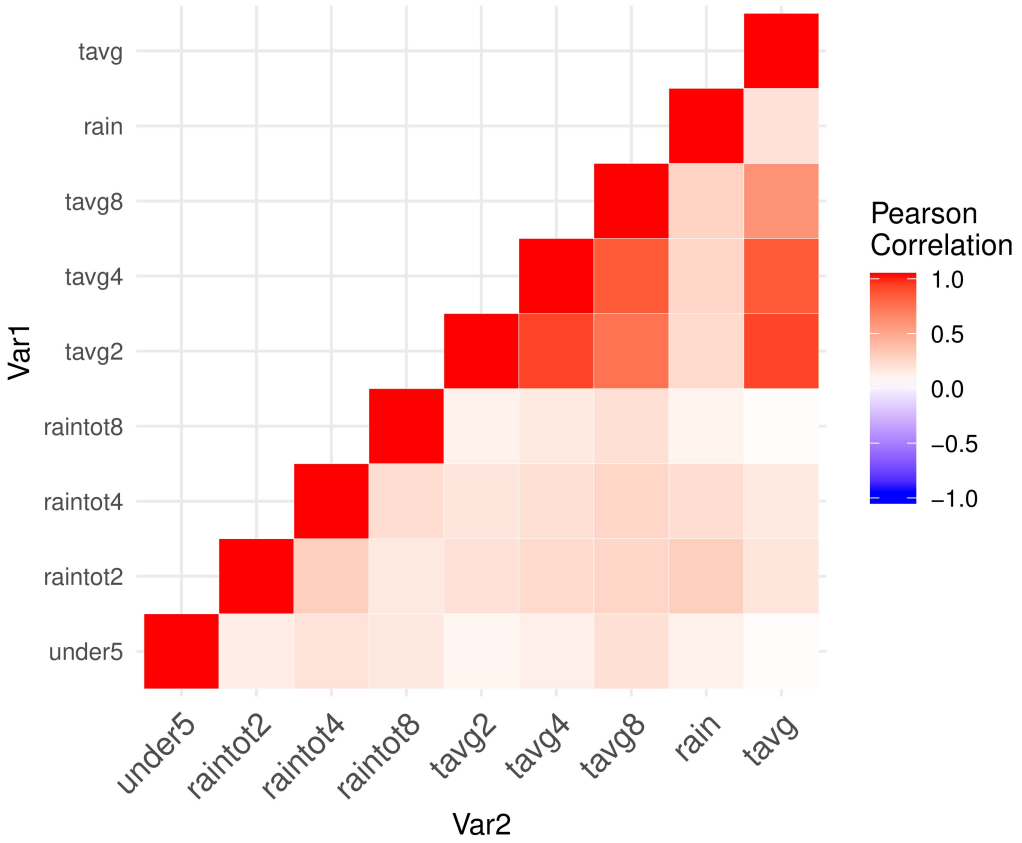
```
##       Var1   Var2 value
```

```
## 1   under5 under5  1.00
## 2 raintot2 under5  0.09
## 3 raintot4 under5  0.14
## 4 raintot8 under5  0.12
## 5    tavg2 under5  0.05
## 6    tavg4 under5  0.08
```

```
upper_tri <- get_upper_tri(cormat)
upper_tri
```

```
##          under5 raintot2 raintot4 raintot8 tavg2 tavg4 tavg8 rain tavg
## under5        1     0.09     0.14     0.12  0.05  0.08  0.16 0.07 0.01
## raintot2     NA     1.00     0.25     0.11  0.15  0.19  0.21 0.25 0.13
## raintot4     NA       NA     1.00     0.18  0.13  0.16  0.21 0.18 0.11
## raintot8     NA       NA       NA     1.00  0.07  0.11  0.16 0.06 0.01
## tavg2        NA       NA       NA       NA  1.00  0.88  0.70 0.19 0.88
## tavg4        NA       NA       NA       NA    NA  1.00  0.80 0.21 0.80
## tavg8        NA       NA       NA       NA    NA    NA  1.00 0.22 0.55
## rain         NA       NA       NA       NA    NA    NA    NA 1.00 0.15
## tavg         NA       NA       NA       NA    NA    NA    NA   NA 1.00
```

```
# Melt the correlation matrix
melted_cormat <- melt(upper_tri, na.rm = TRUE)

# Heatmap
library(ggplot2)
ggplot(data = melted_cormat, aes(Var2, Var1, fill = value))+
  geom_tile(color = "white")+
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1), space = "Lab",
                       name="Pearson\nCorrelation") +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1))+
  coord_fixed()
```




```
melted_cormat
```

```
##         Var1     Var2 value
## 1    under5   under5  1.00
## 10   under5 raintot2  0.09
## 11 raintot2 raintot2  1.00
## 19   under5 raintot4  0.14
## 20 raintot2 raintot4  0.25
## 21 raintot4 raintot4  1.00
## 28   under5 raintot8  0.12
## 29 raintot2 raintot8  0.11
## 30 raintot4 raintot8  0.18
## 31 raintot8 raintot8  1.00
## 37   under5    tavg2  0.05
## 38 raintot2    tavg2  0.15
## 39 raintot4    tavg2  0.13
## 40 raintot8    tavg2  0.07
## 41    tavg2    tavg2  1.00
## 46   under5    tavg4  0.08
## 47 raintot2    tavg4  0.19
## 48 raintot4    tavg4  0.16
## 49 raintot8    tavg4  0.11
## 50    tavg2    tavg4  0.88
## 51    tavg4    tavg4  1.00
## 55   under5    tavg8  0.16
## 56 raintot2    tavg8  0.21
```

```
## 57 raintot4    tavg8  0.21
## 58 raintot8    tavg8  0.16
## 59    tavg2    tavg8  0.70
## 60    tavg4    tavg8  0.80
## 61    tavg8    tavg8  1.00
## 64   under5     rain  0.07
## 65 raintot2     rain  0.25
## 66 raintot4     rain  0.18
## 67 raintot8     rain  0.06
## 68    tavg2     rain  0.19
## 69    tavg4     rain  0.21
## 70    tavg8     rain  0.22
## 71     rain     rain  1.00
## 73   under5     tavg  0.01
## 74 raintot2     tavg  0.13
## 75 raintot4     tavg  0.11
## 76 raintot8     tavg  0.01
## 77    tavg2     tavg  0.88
## 78    tavg4     tavg  0.80
## 79    tavg8     tavg  0.55
## 80     rain     tavg  0.15
## 81     tavg     tavg  1.00
```

```
library(stats)
cpt<-aggregate(data.lag$under5, by=list(Category=data.lag$Epiyear), FUN=sum)
colnames(cpt) <-c("Year", "cpt")
temp<-aggregate(data.lag$tavg8, na.rm=TRUE, by=list(Category=data.lag$Epiyear), FUN=sum)
colnames(temp)<-c("Year", "temp8")
rain<-aggregate(data.lag$raintot4, na.rm=TRUE, by=list(Category=data.lag$Epiyear), FUN=sum)
colnames(rain)<-c("Year", "rain4")
rain2<-rain[-c(1)]
temp2<-temp[-c(1)]
cpt2<-cpt[-c(1)]
prediff<-ts(as.data.frame(cbind(cpt2, temp2, rain2)),start=c(2010, 1), frequency=1)
prediff
```

```
## Time Series:
## Start = 2010
## End = 2016
## Frequency = 1
##            cpt    temp8    rain4
## 2010  67476.82 128073.2 30289.90
## 2011  65208.96 164582.2 38556.74
## 2012  64718.87 168384.7 38472.19
## 2013  76233.95 166853.9 51836.29
## 2014 104023.70 168602.6 38240.62
## 2015 105568.03 175623.8 42006.10
## 2016 110910.02 169938.9 70894.90
```

```
getdiff<-diff(prediff) /prediff[-nrow(prediff),]*100
head(getdiff)
```

```
##             cpt      temp8       rain4
## [1,] -3.3609468 28.5064033  27.2923978
## [2,] -0.7515639  2.3103586  -0.2192872
```

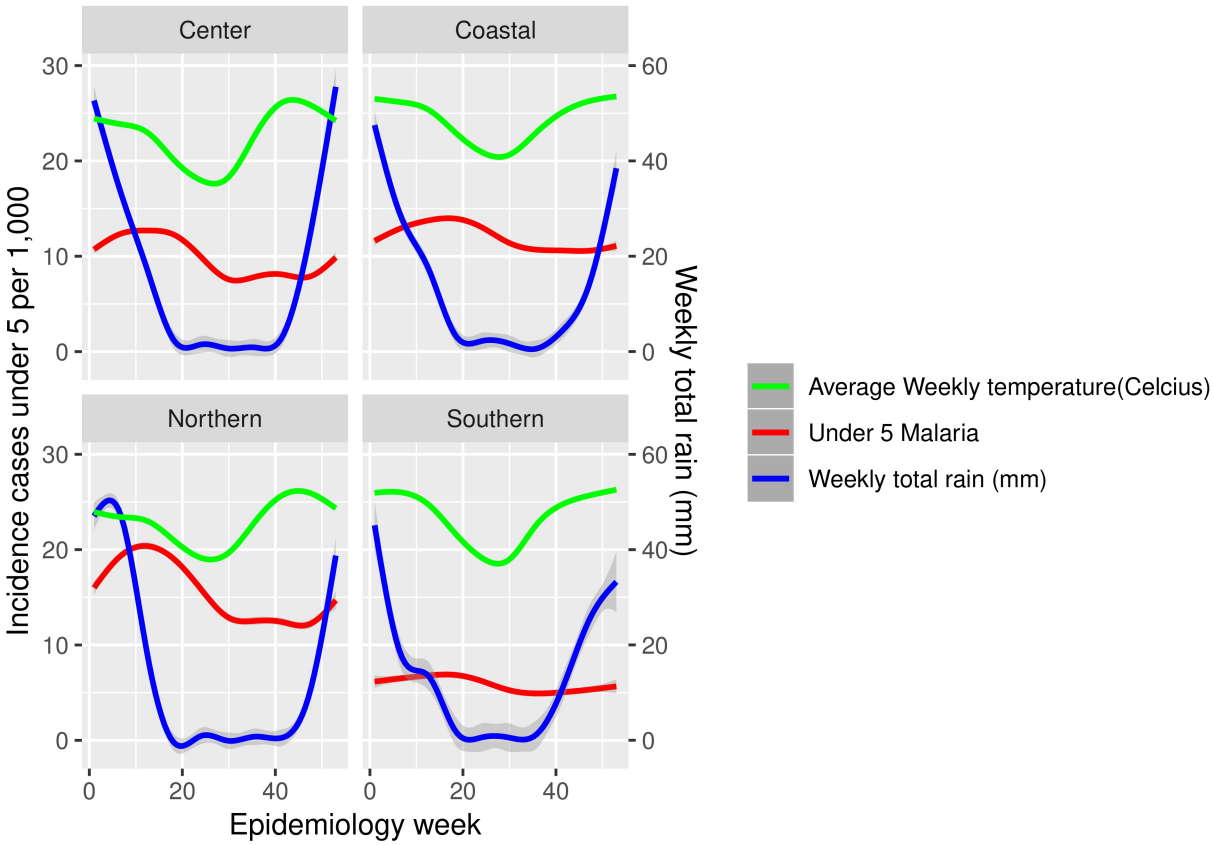
```
## [3,] 17.7924554 -0.9090901  34.7370399
## [4,] 36.4532513  1.0480375 -26.2280923
## [5,]  1.4845895  4.1643600   9.8468069
## [6,]  5.0602396 -3.2369660  68.7728687
```

```
#Add variable for Epiyear (xaxis)#
Epiyear<-c("2011", "2012", "2013", "2014", "2015", "2016")
graphdiff<-as.data.frame(cbind(getdiff,Epiyear))
graphdata<-as.data.frame(graphdiff)
graphdata$temp8<-as.numeric(levels(graphdata$getdiff.temp8))[graphdata$getdiff.temp8]
graphdata$rain4<-as.numeric(levels(graphdata$getdiff.rain4))[graphdata$getdiff.rain4]
graphdata$cpt<-as.numeric(levels(graphdata$getdiff.cpt))[graphdata$getdiff.cpt]
graphdata$Year<-as.numeric(levels(graphdata$Epiyear))[graphdata$Epiyear]
```

```
library(ggplot2)
ggplot(data=graphdata) +
    geom_smooth(mapping = aes(x=Epiyear, y=cpt), color="black")+
    geom_smooth(mapping = aes(x=Epiyear, y =temp8),color= "red")+
    geom_smooth(mapping= aes(x=Epiyear, y =rain4), color="blue")+
  labs(x = "Epidemiology year", y = "Percent change from prior year")
```

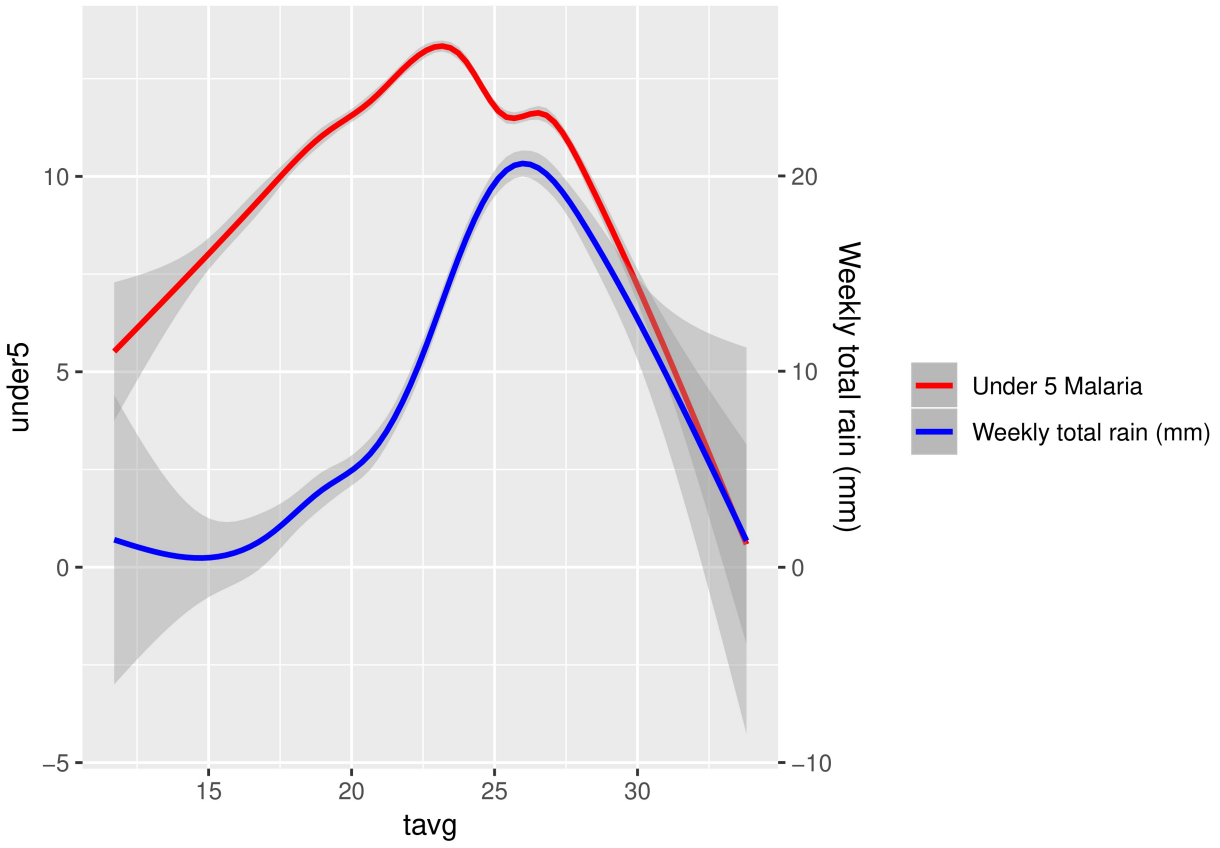
```
##Plot of rain TEMP epi week cpt and region ##
library(ggplot2)
ggplot(data = alldata2) +
  geom_smooth(mapping = aes(x = Epiweek, y = under5, color="Under 5 Malaria")) +
  geom_smooth(mapping = aes(x = Epiweek, y = rainTot, color="Weekly total rain (mm)")) +
  geom_smooth(mapping =aes(x=Epiweek, y=tavg, color = "Average Weekly temperature(Celcius)"))+
  scale_colour_manual("", values = c("Under 5 Malaria" ="red", "Weekly total rain (mm)" ="blue", "Averag
  facet_wrap(~ Region, nrow=2) +
  scale_y_continuous(sec.axis = sec_axis(~.*2, name = "Weekly total rain (mm)")) +
  labs(x = "Epidemiology week", y = "Incidence cases under 5 per 1,000")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
#PLOT OF Under5 vs rain vs tavg
ggplot(data = alldata2) +
  geom_smooth(mapping = aes(x = tavg, y = under5, color="Under 5 Malaria")) +
  geom_smooth(mapping = aes(x = tavg, y = rainTot, color="Weekly total rain (mm)")) +
  #geom_smooth(mapping =aes(x=Epiweek, y=tavg, color = "Average Weekly temperature(Celcius)"))+
  scale_colour_manual("", values = c("Under 5 Malaria" ="red", "Weekly total rain (mm)" ="blue", "Averag
  scale_y_continuous(sec.axis = sec_axis(~.*2, name = "Weekly total rain (mm)"))
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
#labs(x = "Epidemiology week", y = "Incidence cases under 5 per 1,000")
```