GEOSPATIAL MODELLING AND MAPPING TRAINING WORKSHOP - AMMNet 2024 HOHOE, GHANA

Introduction to Geospatial Modelling and Mapping with R

Facilitator/Trainer: Prof. Justice Moses K. Aheto

MAY to AUGUST 2024

Learning objectives: This module will help you understand spatial data types and will show you how to develop geospatial models and to create geospatial maps for both observed and predicted health outcomes with focus on Malaria using the R free software and its associated packages. We will be working with possible scenarios, understanding the data, maps and why this is critical to support health policy and intervention strategies related to malaria control and elimination programs.

By the end of this session, you will:

1. Understand spatial data types
2. Be able to identify and download various sources of spatial data, especially shapefiles.
3. Be able to load and map spatial data (e.g., shapefiles, raster files)
4. Be able to load and map spatial data in CSV file formats

5. Be aware of the different types of mapping procedures and their appropriateness.
6. Know how to produce maps for different types of spatial data using the mapping functions such as plot, spplot, tmap, leaflet, and their appropriateness.
7. Know how to produce both static and web-based interactive maps for disease surveillance spatial data, especially malaria.
8. Know how to correctly interpret results from the Geospatial maps to support policy decisions in the presence of limited public health resources.
9. Be able to develop geospatial models to analyse and map geostatistical and lattice spatial datasets to support malaria control and elimination programs.

Install the required packages

#install.packages("raster") #install.packages("maptools") #install.packages("gtools") #install.packages("sp") #install.packages("spdep") #install.packages("rgdal") #install.packages("ggplot2") #install.packages("tiff") #install.packages("leaflet") #install.packages("tmap") #install.packages("RColorBrewer") #install.packages("dplyr") #install.packages("OpenStreetMap") #install.packages("maptiles")

```
# Loading required packages to read in the data and for spatial data manipulation, modelling and mapping
library(raster)
```

```
## Loading required package: sp
```

```
library(maptools)
```

```
## Please note that 'maptools' will be retired during October 2023,
## plan transition at your earliest convenience (see
## https://r-spatial.org/r/2023/05/15/evolution4.html and earlier blogs
## for guidance);some functionality will be moved to 'sp'.
##  Checking rgeos availability: TRUE
```

```
##
## Attaching package: 'maptools'
```

```
## The following object is masked from 'package:sp':
##
##     sp2Mondrian
```

```
library(gtools)
library(sp)
library(spdep)
```

```
## 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')`
```

```
## Loading required package: sf
```

```
## Linking to GEOS 3.11.2, GDAL 3.8.2, PROJ 9.3.1; sf_use_s2() is TRUE
```

```
library(rgdal)
```

```
## Please note that rgdal will be retired during October 2023,
## plan transition to sf/stars/terra functions using GDAL and PROJ
## at your earliest convenience.
## See https://r-spatial.org/r/2023/05/15/evolution4.html and https://github.com/r-spatial/evolution
## rgdal: version: 1.6-7, (SVN revision 1203)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 3.7.2, released 2023/09/05
## Path to GDAL shared files: C:/Users/jmkaheto/AppData/Local/R/win-library/4.3/rgdal/gdal
##  GDAL does not use iconv for recoding strings.
## GDAL binary built with GEOS: TRUE
## Loaded PROJ runtime: Rel. 9.3.0, September 1st, 2023, [PJ_VERSION: 930]
## Path to PROJ shared files: C:/Users/jmkaheto/AppData/Local/R/win-library/4.3/rgdal/proj
## PROJ CDN enabled: FALSE
## Linking to sp version:2.1-1
## To mute warnings of possible GDAL/OSR exportToProj4() degradation,
## use options("rgdal_show_exportToProj4_warnings"="none") before loading sp or rgdal.
```

```
library(ggplot2)
library(tiff)
library(leaflet)
library(tmap)
```

```
##
## Attaching package: 'tmap'
```

```
## The following object is masked from 'package:datasets':
##
##     rivers
```

```
library(tmaptools)
library(RColorBrewer)
library(dplyr)
```

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

```
## The following objects are masked from 'package:raster':
##
##     intersect, select, union
```

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

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

```
library(spatsurv)
```

```
##
##  Welcome to 'spatsurv': Spatial Survival Analysis
##  B. M. Taylor & B. S. Rowlingson.
##
## Type 'spatsurvVignette()' to view the package vignette.
##
## Type 'citation("spatsurv")' to view the citation for this package.
##
## Please see the spatsurv package NEWS file for latest additions, changes and bug fixes.
```

```
##
## Attaching package: 'spatsurv'
```

```
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
library(maptiles)
library(tinytex)
library(webshot2)



# Finding the working directory
#getwd()

# Setting the working directory

setwd("C:/Users/jmkaheto/Documents/Dropbox/Consult/Prof. Afrane Yaw/Ghana Workshop MAY 2024")
```

Link to download some spatial data for various countries

https://spatialdata.dhsprogram.com/data/#/single/surveys (https://spatialdata.dhsprogram.com/data/#/single/surveys)

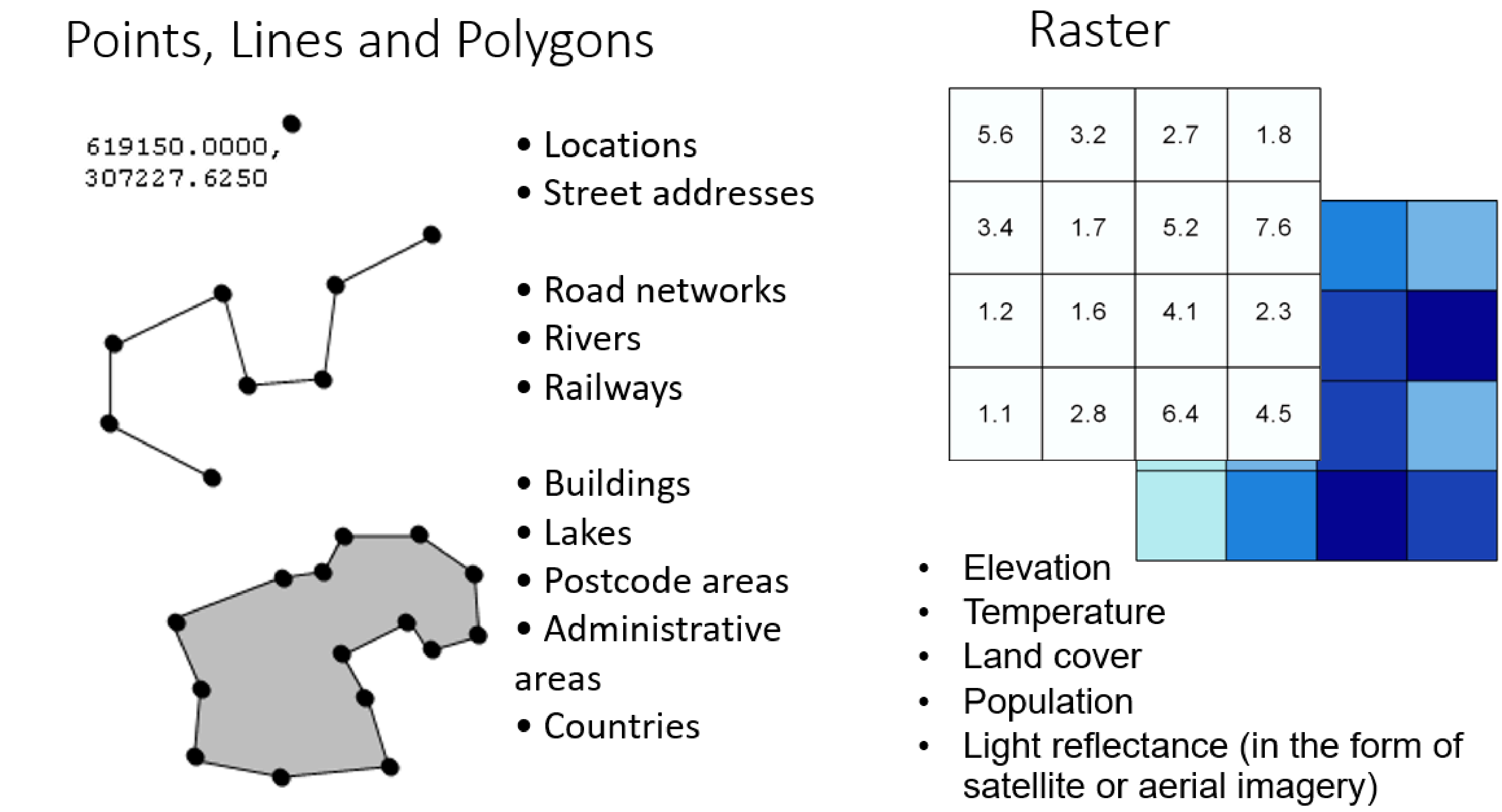
For Ghana specifically

https://spatialdata.dhsprogram.com/boundaries/#view=table&countryId=GH (https://spatialdata.dhsprogram.com/boundaries/#view=table&countryId=GH)

DAY 2: 18th April 2023

Module 1: Geospatial mapping of malaria in R (lattice data)

Understanding spatial data types



Some spatial data types.

We need to understand the Raster format, and transform it to data frames for

analysis as below:

- We have to separate the values in the data from the data structure:

5.6	3.2	2.7	1.8
3.4	1.7	5.2	7.6
1.2	1.6	4.1	2.3
1.1	2.8	6.4	4.5

5.6 3.2 2.7 1.8

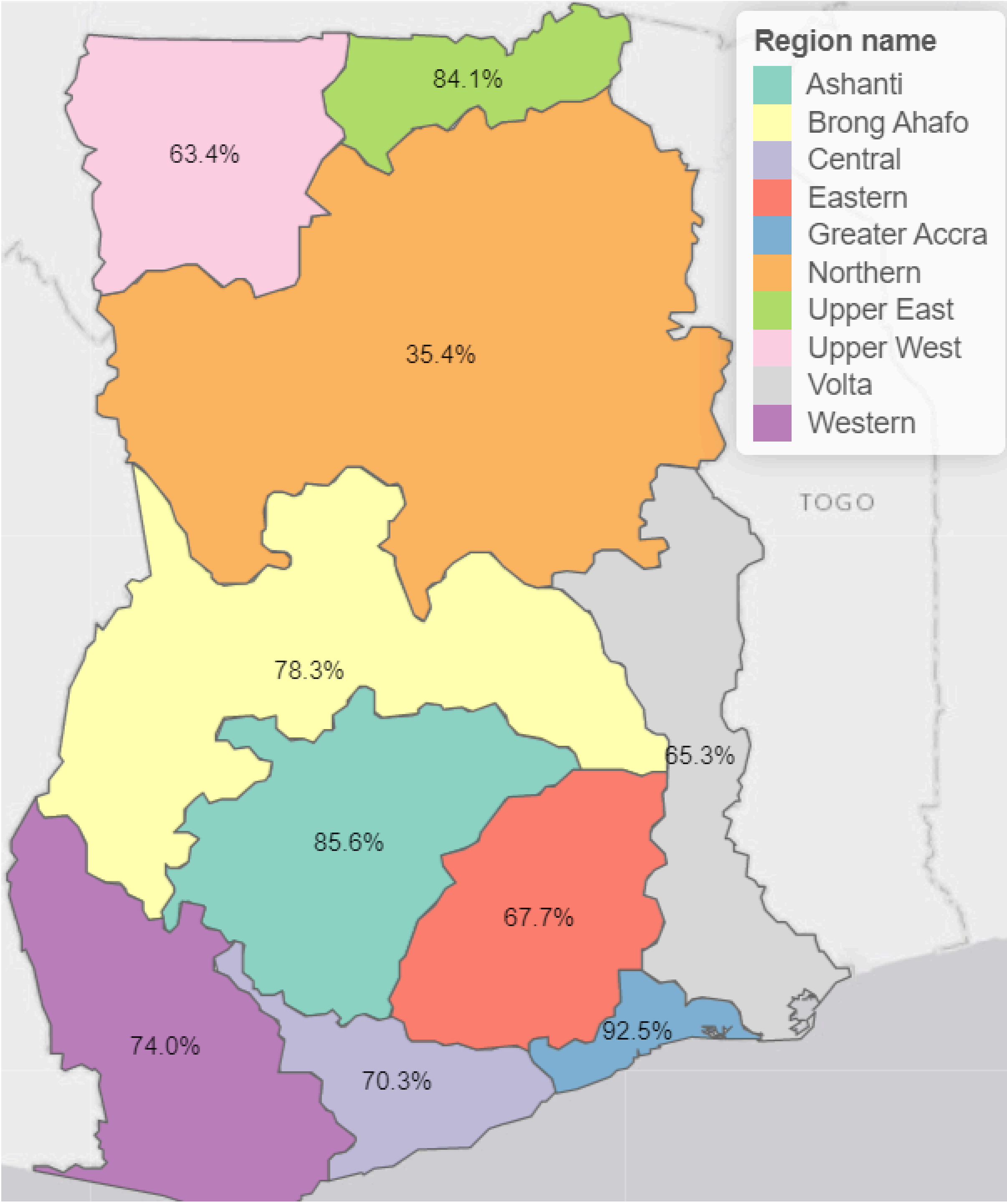
3.4 1.7 5.2 7.6

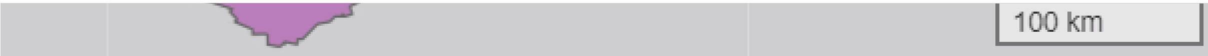
1.2 1.6 4.1 2.3

1.1 2.8 6.4 4.5

Understanding spatial data types.

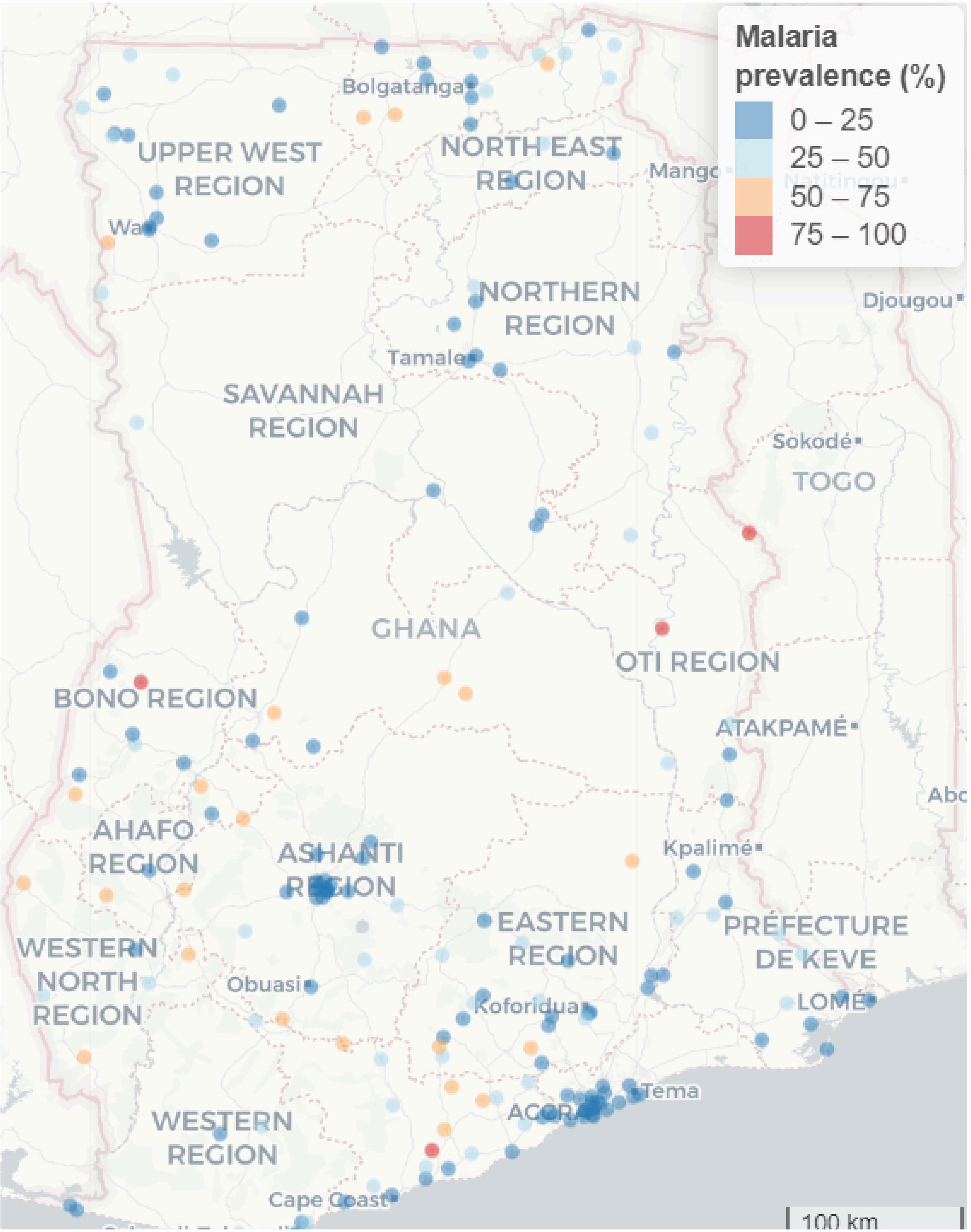
Here is an example lattice data

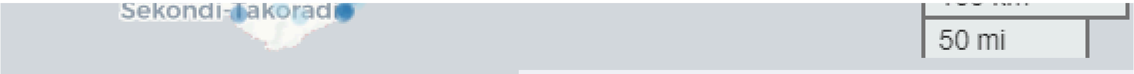

Region name
Ashanti
Brong Ahafo
Central
Eastern
Greater Accra
Northern
Upper East
Upper West
Volta
Western
84.1%
63.4%
35.4%
TOGO
78.3%
65.3%
85.6%
67.7%
92.5%
74.0%
70.3%



Lattice data.

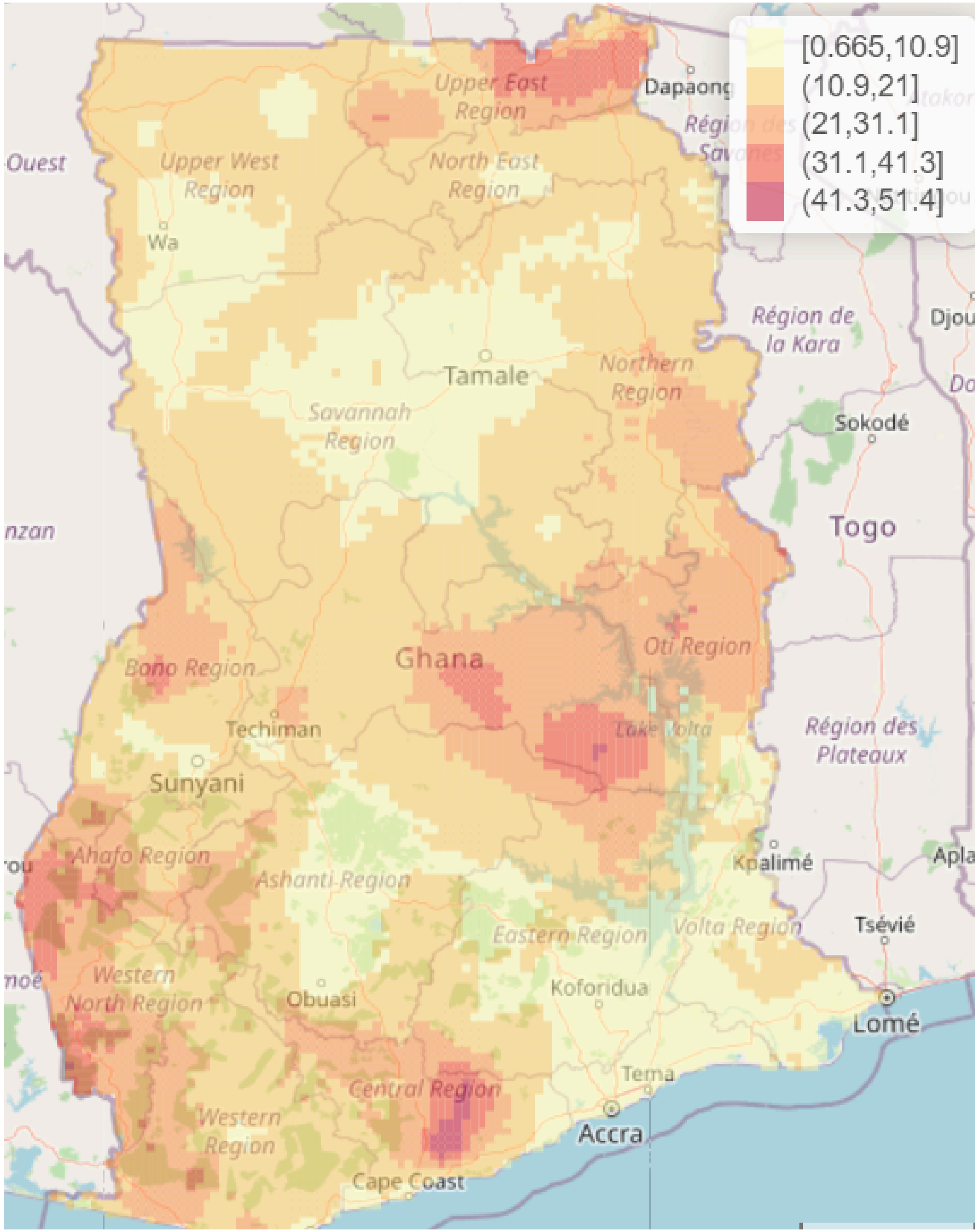
Here is an example Geostatistical data


Malaria prevalence (%)
0 – 25
25 – 50
50 – 75
75 – 100
Bolgatanga
UPPER WEST REGION
NORTH EAST REGION
Mango
Wa
NORTHERN REGION
Djougou
Tamale
SAVANNAH REGION
Sokodé
TOGO
GHANA
OTI REGION
BONO REGION
ATAKPAMÉ
Kpalimé
AHAFO REGION
ASHANTI REGION
EASTERN REGION
PRÉFECTURE DE KEVE
WESTERN NORTH REGION
Obuasi
Koforidua
LOMÉ
Tema
ACCRA
WESTERN REGION
Cape Coast
100 km



Geostatistical data.

Here is an example Raster data


[0.665,10.9]
(10.9,21]
(21,31.1]
(31.1,41.3]
(41.3,51.4]
Upper East Region
Upper West Region
North East Region
Northern Region
Savannah Region
Tamale
Wa
Dapaong
Région de la Kara
Sokodé
Togo
Ghana
Oti Region
Bono Region
Techiman
Sunyani
Lake Volta
Région des Plateaux
Ahafo Region
Ashanti Region
Kpalimé
Eastern Region
Volta Region
Tsévié
Western North Region
Obuasi
Koforidua
Lomé
Tema
Central Region
Accra
Western Region
Cape Coast

Raster data.

Geospatial mapping of malaria in R (lattice data)

Mapping lattice data

Mapping with SpatialPolygonsDataFrame object

Malaria data: 2019 Ghana Malaria Indicator Survey (GMIS) data

Starting from scratch (Look for malaria report

Read in the regional shapefile for Ghana (downloaded from the link below): https://gadm.org/download_country40.html (https://gadm.org/download_country40.html)

```
reg<- readOGR("gadm40_GHA_shp","gadm40_GHA_1")
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\jmkaheto\Documents\Dropbox\Consult\Dr.Jaline Gerardin Northwestern Univ USA\AMMNet_Workshop_
Ghana_Aheto\gadm40_GHA_shp", layer: "gadm40_GHA_1"
## with 10 features
## It has 11 fields
```

```
class(reg)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
plot(reg)

head(reg@data) # Retrieve the data frame only
```

```
#save(reg,file="reg.RData")
#load("reg.RData")

### Extract only the dataframe
data<-reg@data
class(data)
```

```
## [1] "data.frame"
```

```
names(data)
```

```
##  [1] "ID_0"      "COUNTRY"   "ID_1"      "NAME_1"    "VARNAME_1" "NL_NAME_1"
##  [7] "TYPE_1"    "ENGTYPE_1" "CC_1"      "HASC_1"    "ISO_1"
```

```
# We need only the region ID (ID_1) and name (NAME_1) variables so we keep only
# that as below and save the data as csv file for further processing
var<-c("ID_1","NAME_1")
data<-data[,var]
names(data)
```

```
## [1] "ID_1"   "NAME_1"
```

```
write.csv(data,"data.csv")

# Now, we have two ways to capture the malaria data into the regional shapefile from
# the 2019 GMIS data

# Steps:

# 1. Open the csv file data manually to enter the figures for each region from the
# 2019 GMIS report and save it. See page xiv (i.e., page 16 of 159)

# 2. Create a new data in R and add it to the regional shape file. Be sure to
# assigned the malaria prevalence (for RDT) figures to their respective regions
# We have done this procedure under lattice example earlier so will focus on step 1)


# Continue from here after step 1 operation above

regc<-read.csv("data2.csv") # Using csv
summary(regc)
```

```
##      ID_1               NAME_1            malaria_p
##  Length:10          Length:10          Min.   : 1.00
##  Class :character   Class :character   1st Qu.:19.68
##  Mode  :character   Mode  :character   Median :28.15
##                                        Mean   :24.49
##                                        3rd Qu.:31.05
##                                        Max.   :35.40
```

```
## Merge region data with the shapefile
names(reg)
```

```
##  [1] "ID_0"      "COUNTRY"   "ID_1"      "NAME_1"    "VARNAME_1" "NL_NAME_1"
##  [7] "TYPE_1"    "ENGTYPE_1" "CC_1"      "HASC_1"    "ISO_1"
```

```
names(regc)
```

```
## [1] "ID_1"      "NAME_1"    "malaria_p"
```

```
reg_data<-merge(reg, regc,by="NAME_1")
summary(reg_data)
```

```
## Object of class SpatialPolygonsDataFrame
## Coordinates:
##         min      max
## x -3.255419  1.19177
## y  4.738751 11.17330
## Is projected: FALSE
## proj4string : [+proj=longlat +datum=WGS84 +no_defs]
## Data attributes:
##     NAME_1              ID_0              COUNTRY             ID_1.x
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   VARNAME_1          NL_NAME_1            TYPE_1           ENGTYPE_1
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      CC_1              HASC_1              ISO_1              ID_1.y
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    malaria_p
##  Min.   : 1.00
##  1st Qu.:19.68
##  Median :28.15
##  Mean   :24.49
##  3rd Qu.:31.05
##  Max.   :35.40
```

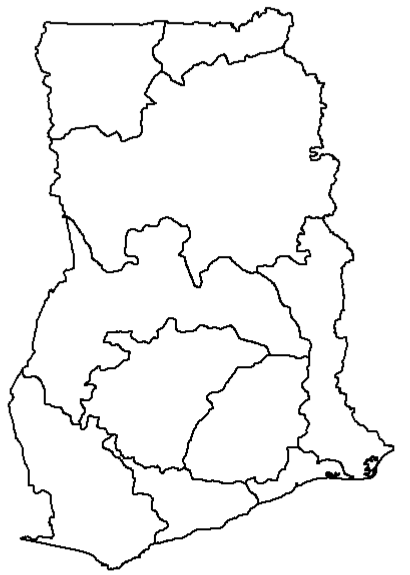
```
save(reg_data,file="reg_data.RData")


names(reg_data)
```

```
##  [1] "NAME_1"    "ID_0"      "COUNTRY"   "ID_1.x"    "VARNAME_1" "NL_NAME_1"
##  [7] "TYPE_1"    "ENGTYPE_1" "CC_1"      "HASC_1"    "ISO_1"     "ID_1.y"
## [13] "malaria_p"
```

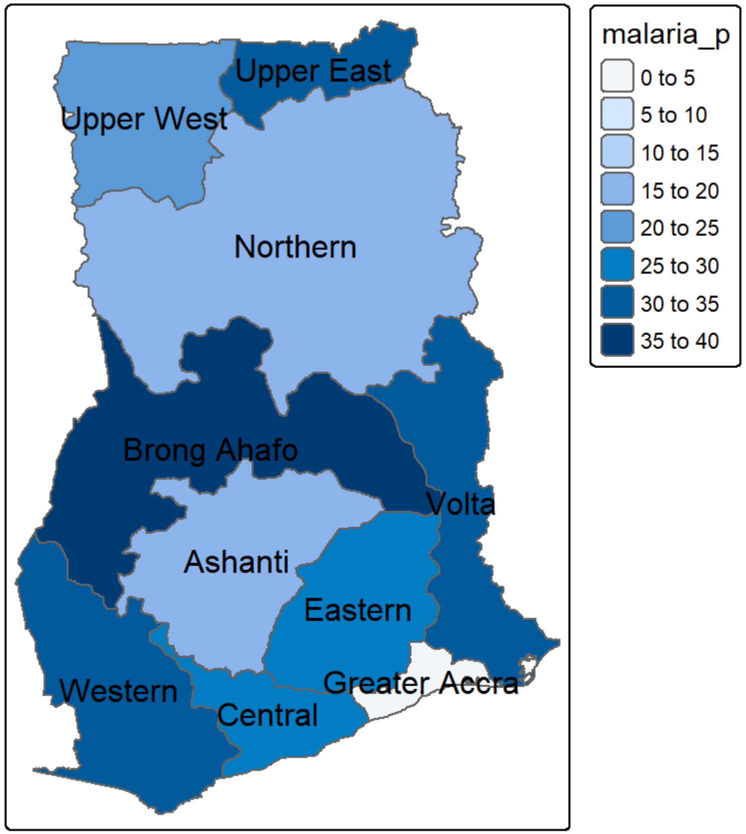
```
summary(reg_data)
```

```
## Object of class SpatialPolygonsDataFrame
## Coordinates:
##         min      max
## x -3.255419  1.19177
## y  4.738751 11.17330
## Is projected: FALSE
## proj4string : [+proj=longlat +datum=WGS84 +no_defs]
## Data attributes:
##     NAME_1              ID_0              COUNTRY             ID_1.x
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   VARNAME_1          NL_NAME_1            TYPE_1           ENGTYPE_1
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      CC_1              HASC_1             ISO_1              ID_1.y
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    malaria_p
##  Min.   : 1.00
##  1st Qu.:19.68
##  Median :28.15
##  Mean   :24.49
##  3rd Qu.:31.05
##  Max.   :35.40
```

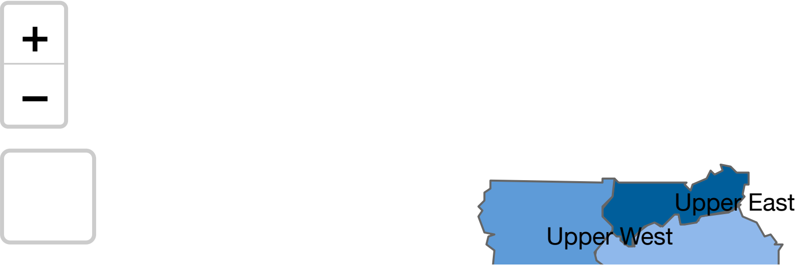
```r
plot(reg_data) # plotting only the regional polygons
```

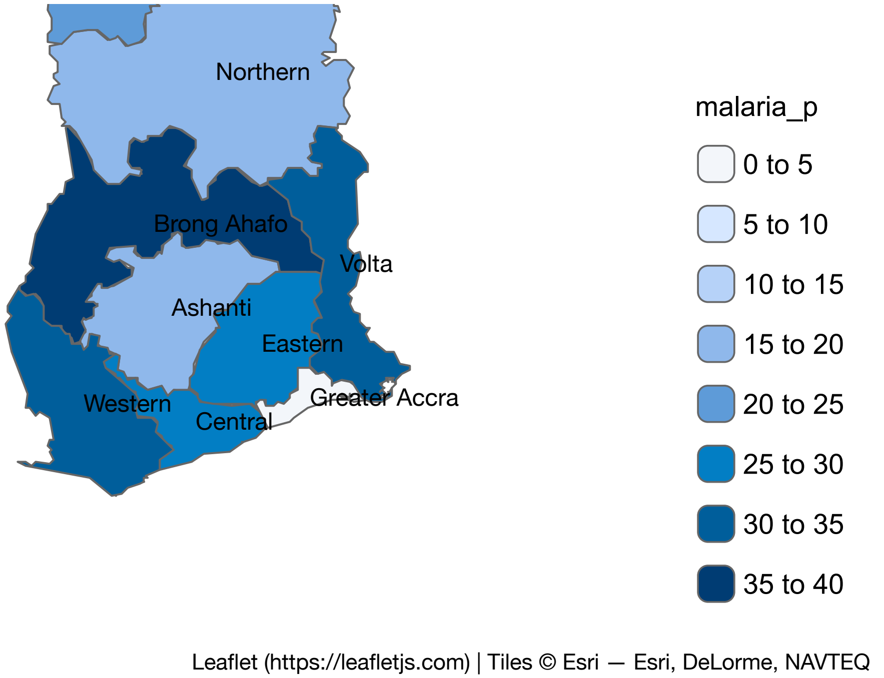


```r
# install.packages("tmap")

library(tmap)

# Convert to sf object for mapping
reg_data<-st_as_sf(reg_data)

# Mapping using tmap
wmap<-tm_shape(reg_data) +
    tm_polygons("malaria_p")+
    tm_text("NAME_1")

wmap
```




```
# Produce the web based version in html
wm <- tmap_leaflet(wmap)
wm
```





Improve the map

```
summary(reg_data)
```

```
##     NAME_1              ID_0              COUNTRY              ID_1.x         
##  Length:10          Length:10          Length:10          Length:10         
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##   VARNAME_1          NL_NAME_1             TYPE_1            ENGTYPE_1        
##  Length:10          Length:10          Length:10          Length:10         
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##      CC_1              HASC_1              ISO_1              ID_1.y         
##  Length:10          Length:10          Length:10          Length:10         
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##    malaria_p               geometry 
##  Min.   : 1.00   MULTIPOLYGON :10  
##  1st Qu.:19.68   epsg:4326    : 0  
##  Median :28.15   +proj=long...: 0  
##  Mean   :24.49                     
##  3rd Qu.:31.05                     
##  Max.   :35.40                     
```

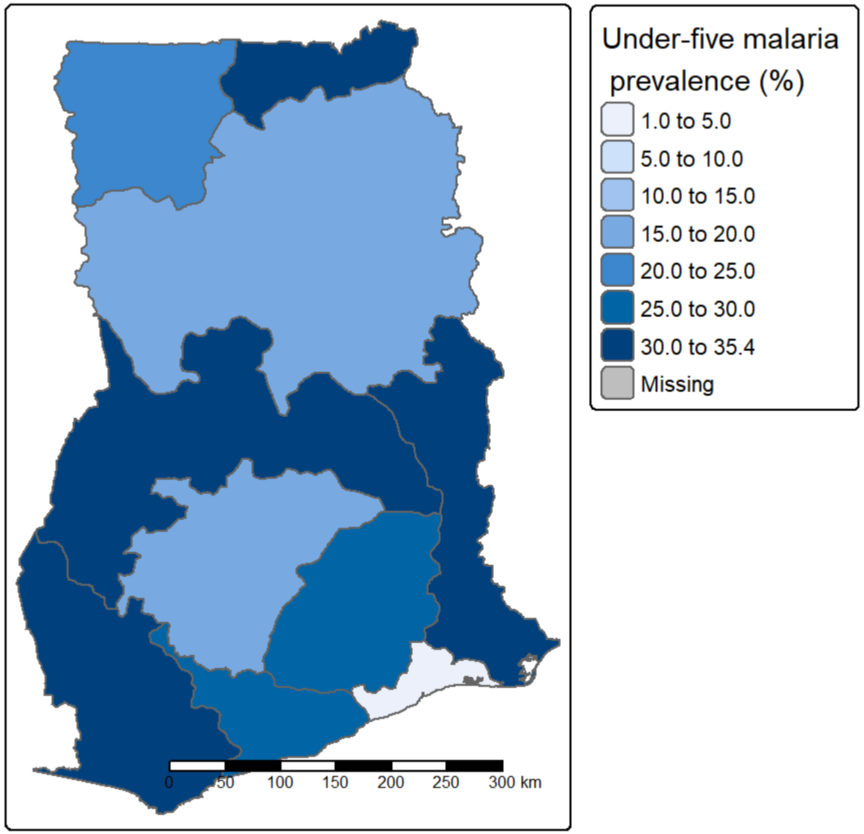
```
mal_map<-tm_shape(reg_data)+
tm_polygons("malaria_p",title="Under-five malaria \n prevalence (%)",
              style = "fixed",
              breaks = c(1,5,10,15,20,25,30,35.4),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

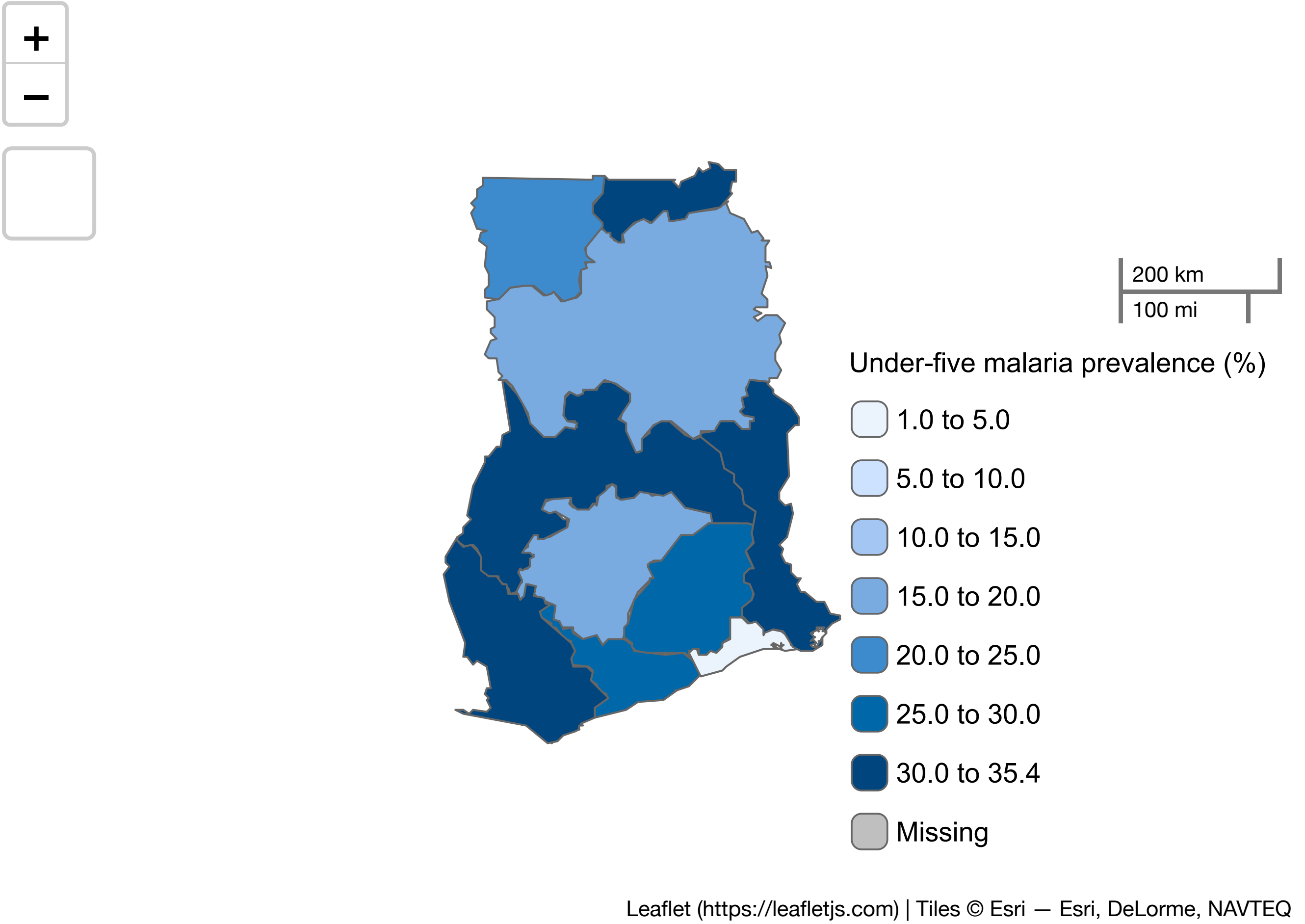
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks' to 'tm_scale_intervals(<HERE>)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_map
```



```
mal_map_web <- tmap_leaflet(mal_map)
mal_map_web
```



Using Open Street Map basemap tiles and removing NAs from the legend

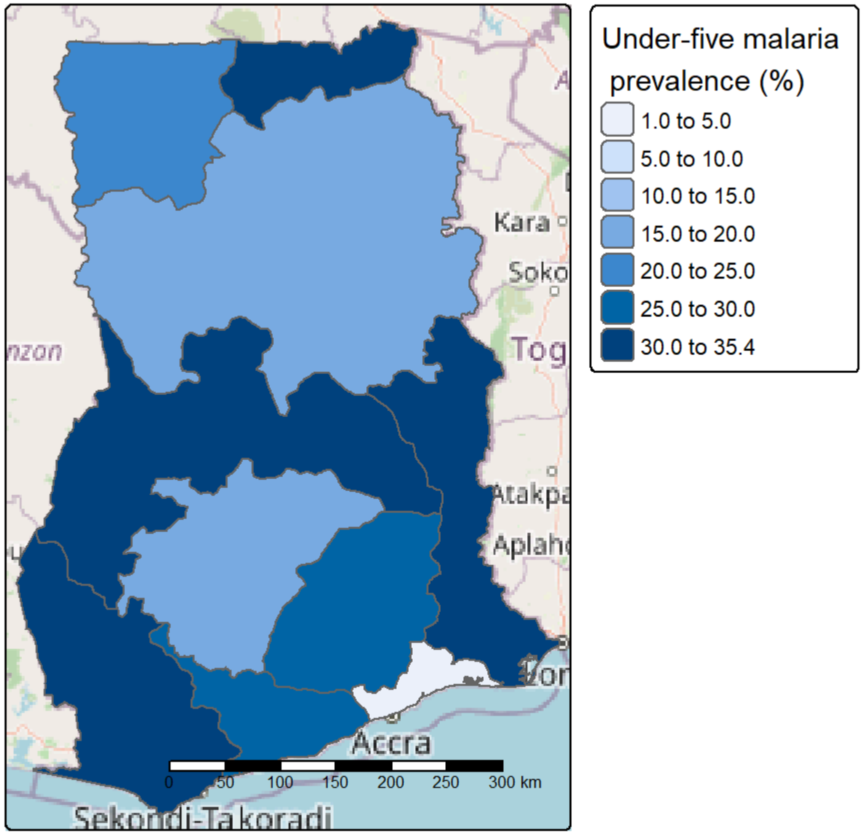
```
mal_map<-tm_basemap(leaflet::providers$OpenStreetMap) +
tm_shape(reg_data)+
tm_polygons("malaria_p",title="Under-five malaria \n prevalence (%)",
              style = "fixed",textNA="",
              breaks = c(1,5,10,15,20,25,30,35.4),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

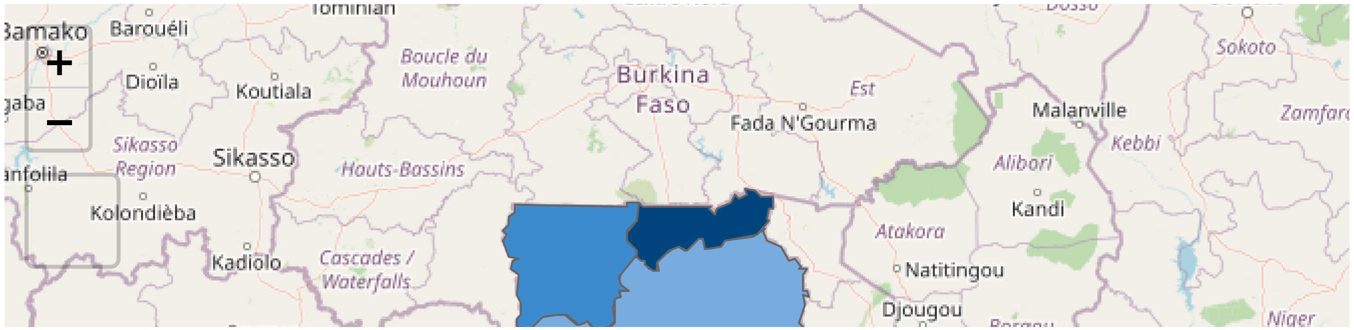
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples,
specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

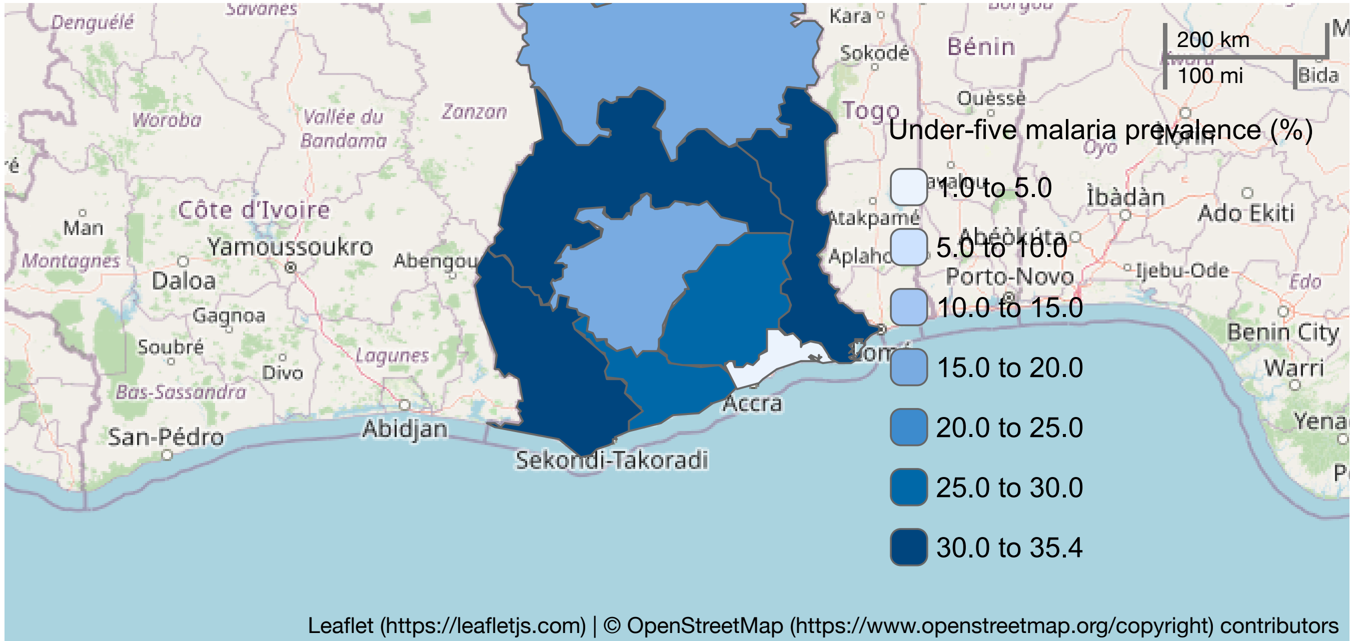
```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_map
```




```
mal_map_web <- tmap_leaflet(mal_map)
mal_map_web
```





Hurrayyyyyyyyyyyyyyyyyyyyyyyyyyyy you are now experts in geospatial mapping using lattice data. Wait a minute, more to work on in the next sections for more improvement in the maps!

Example 2: download malaria data directly from DHS Program website below:

https://spatialdata.dhsprogram.com/data/#/ (https://spatialdata.dhsprogram.com/data/#/)

Now, load the data

```
mal_dat<-readOGR("shps","sdr_subnational_data_mis_2019")
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\jmkaheto\Documents\Dropbox\Consult\Dr.Jaline Gerardin Northwestern Univ USA\AMMNet_Workshop_
Ghana_Aheto\shps", layer: "sdr_subnational_data_mis_2019"
## with 10 features
## It has 48 fields
```

```
names(mal_dat) # see the names in the data (see the note file for description)
```

```
##  [1] "ISO"        "FIPS"       "DHSCC"      "SVYTYPE"    "SVYYEAR"
##  [6] "CNTRYNAMEE" "CNTRYNAMEF" "CNTRYNAMES" "DHSREGEN"   "DHSREGFR"
## [11] "DHSREGSP"   "SVYID"      "REG_ID"     "Svy_Map"    "MULTLEVEL"
## [16] "LEVELRNK"   "REGVAR"     "REGCODE"    "REGNAME"    "OTHREGVAR"
## [21] "OTHREGCO"   "OTHREGNA"   "LEVELCO"    "LEVELNA"    "REPALLIND"
## [26] "REGNOTES"   "SVYNOTES"   "MLCMLTCANM" "MLCMLTCRDT" "MLCMLTCMSY"
## [31] "MLCMLTCNUM" "MLCMLTCUNW" "MLPMALCRDT" "MLPMALCRDE" "MLPMALCRDR"
## [36] "MLPMALCRDL" "MLPMALCRDU" "MLPMALCNMR" "MLPMALCUNR" "MLPMALCUER"
## [41] "MLPMALCMSY" "MLPMALCMSE" "MLPMALCMSR" "MLPMALCMSL" "MLPMALCMSU"
## [46] "MLPMALCNMM" "MLPMALCUNM" "MLPMALCUEM"
```

```
summary(mal_dat)
```

```
## Object of class SpatialPolygonsDataFrame
## Coordinates:
##         min       max
## x -3.249167  1.199572
## y  4.735417 11.166668
## Is projected: FALSE
## proj4string : [+proj=longlat +datum=WGS84 +no_defs]
## Data attributes:
##      ISO                FIPS               DHSCC              SVYTYPE
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     SVYYEAR      CNTRYNAMEE         CNTRYNAMEF         CNTRYNAMES
##  Min.   :2019   Length:10          Length:10          Length:10
##  1st Qu.:2019   Class :character   Class :character   Class :character
##  Median :2019   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2019
##  3rd Qu.:2019
##  Max.   :2019
##    DHSREGEN           DHSREGFR           DHSREGSP             SVYID
##  Length:10          Length:10          Length:10          Min.   :557
##  Class :character   Class :character   Class :character   1st Qu.:557
##  Mode  :character   Mode  :character   Mode  :character   Median :557
##                                                           Mean   :557
##                                                           3rd Qu.:557
##                                                           Max.   :557
##     REG_ID            Svy_Map           MULTLEVEL            LEVELRNK
##  Length:10          Length:10          Length:10          Min.   :1
##  Class :character   Class :character   Class :character   1st Qu.:1
##  Mode  :character   Mode  :character   Mode  :character   Median :1
##                                                           Mean   :1
##                                                           3rd Qu.:1
##                                                           Max.   :1
##     REGVAR             REGCODE        REGNAME           OTHREGVAR
##  Length:10          Min.   : 1.00   Length:10          Length:10
```

```
##  Class :character   1st Qu.: 3.25   Class :character   Class :character  
##  Mode  :character   Median : 5.50   Mode  :character   Mode  :character  
##                     Mean   : 5.50                                        
##                     3rd Qu.: 7.75                                        
##                     Max.   :10.00                                        
##     OTHREGCO      OTHREGNA            LEVELCO            LEVELNA         
##  Min.   :9999   Length:10          Length:10          Length:10          
##  1st Qu.:9999   Class :character   Class :character   Class :character   
##  Median :9999   Mode  :character   Mode  :character   Mode  :character   
##  Mean   :9999                                                            
##  3rd Qu.:9999                                                            
##  Max.   :9999                                                            
##   REPALLIND           REGNOTES           SVYNOTES           MLCMLTCANM    
##  Length:10          Length:10          Length:10          Min.   : 92.10  
##  Class :character   Class :character   Class :character   1st Qu.: 95.72  
##  Mode  :character   Mode  :character   Mode  :character   Median : 98.30  
##                                                           Mean   : 97.33  
##                                                           3rd Qu.: 99.70  
##                                                           Max.   :100.00  
##    MLCMLTCRDT       MLCMLTCMSY       MLCMLTCNUM      MLCMLTCUNW  
##  Min.   : 91.60   Min.   : 92.10   Min.   :203.0   Min.   :9999  
##  1st Qu.: 95.12   1st Qu.: 95.47   1st Qu.:256.0   1st Qu.:9999  
##  Median : 98.30   Median : 98.15   Median :274.5   Median :9999  
##  Mean   : 97.08   Mean   : 97.21   Mean   :292.0   Mean   :9999  
##  3rd Qu.: 99.00   3rd Qu.: 99.58   3rd Qu.:300.0   3rd Qu.:9999  
##  Max.   :100.00   Max.   :100.00   Max.   :518.0   Max.   :9999  
##    MLPMALCRDT      MLPMALCRDE      MLPMALCRDR      MLPMALCRDL   
##  Min.   : 1.00   Min.   :0.800   Min.   :0.200   Min.   : 0.00  
##  1st Qu.:19.68   1st Qu.:4.400   1st Qu.:0.200   1st Qu.:12.12  
##  Median :28.15   Median :4.750   Median :0.200   Median :15.75  
##  Mean   :24.49   Mean   :4.800   Mean   :0.280   Mean   :14.94  
##  3rd Qu.:31.05   3rd Qu.:5.175   3rd Qu.:0.275   3rd Qu.:20.75  
##  Max.   :35.40   Max.   :9.000   Max.   :0.800   Max.   :23.70  
##    MLPMALCRDU      MLPMALCNMR      MLPMALCUNR      MLPMALCUER   
##  Min.   : 2.60   Min.   : 83.0   Min.   :186.0   Min.   :203.0  
##  1st Qu.:27.25   1st Qu.:197.2   1st Qu.:250.2   1st Qu.:256.0  
##  Median :37.95   Median :275.0   Median :270.5   Median :274.5  
##  Mean   :34.11   Mean   :261.1   Mean   :284.3   Mean   :292.0  
```

```
##  3rd Qu.:41.27   3rd Qu.:303.2   3rd Qu.:294.2   3rd Qu.:300.0  
##  Max.   :51.20   Max.   :421.0   Max.   :509.0   Max.   :518.0  
##    MLPMALCMSY       MLPMALCMSE       MLPMALCMSR      MLPMALCMSL       MLPMALCMSU   
##  Min.   : 2.40   Min.   :1.300   Min.   :0.20   Min.   : 0.00   Min.   : 5.00  
##  1st Qu.:10.43   1st Qu.:2.750   1st Qu.:0.20   1st Qu.: 4.30   1st Qu.:16.05  
##  Median :12.65   Median :3.400   Median :0.25   Median : 5.45   Median :21.10  
##  Mean   :14.07   Mean   :3.260   Mean   :0.29   Mean   : 7.58   Mean   :20.62  
##  3rd Qu.:17.52   3rd Qu.:4.075   3rd Qu.:0.30   3rd Qu.:10.78   3rd Qu.:24.32  
##  Max.   :27.00   Max.   :4.300   Max.   :0.60   Max.   :18.40   Max.   :35.60  
##    MLPMALCNMM       MLPMALCUNM       MLPMALCUEM   
##  Min.   : 83.0   Min.   :187.0   Min.   :203.0  
##  1st Qu.:197.0   1st Qu.:249.2   1st Qu.:256.0  
##  Median :276.5   Median :271.5   Median :274.5  
##  Mean   :261.8   Mean   :284.6   Mean   :292.0  
##  3rd Qu.:305.5   3rd Qu.:294.2   3rd Qu.:300.0  
##  Max.   :421.0   Max.   :509.0   Max.   :518.0
```

```
head(mal_dat)
```

```
# Percentage of children age 6-59 months tested for anemia

summary(mal_dat$MLCMLTCANM) # some summary statistics
```

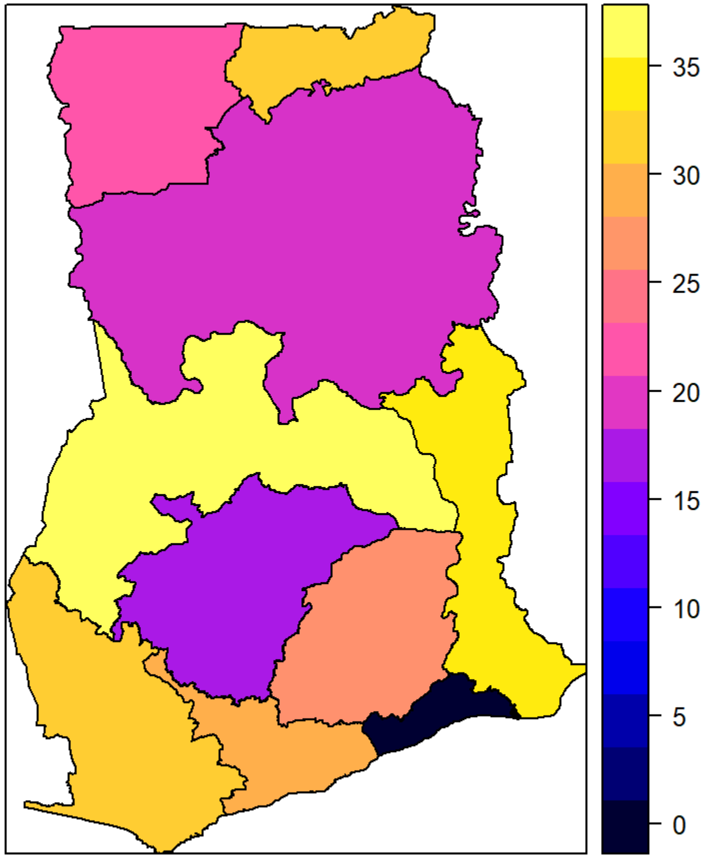
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   92.10   95.72   98.30   97.33   99.70  100.00
```

Mapping using spplot funtion from sp/raster package (Not the focus here though)

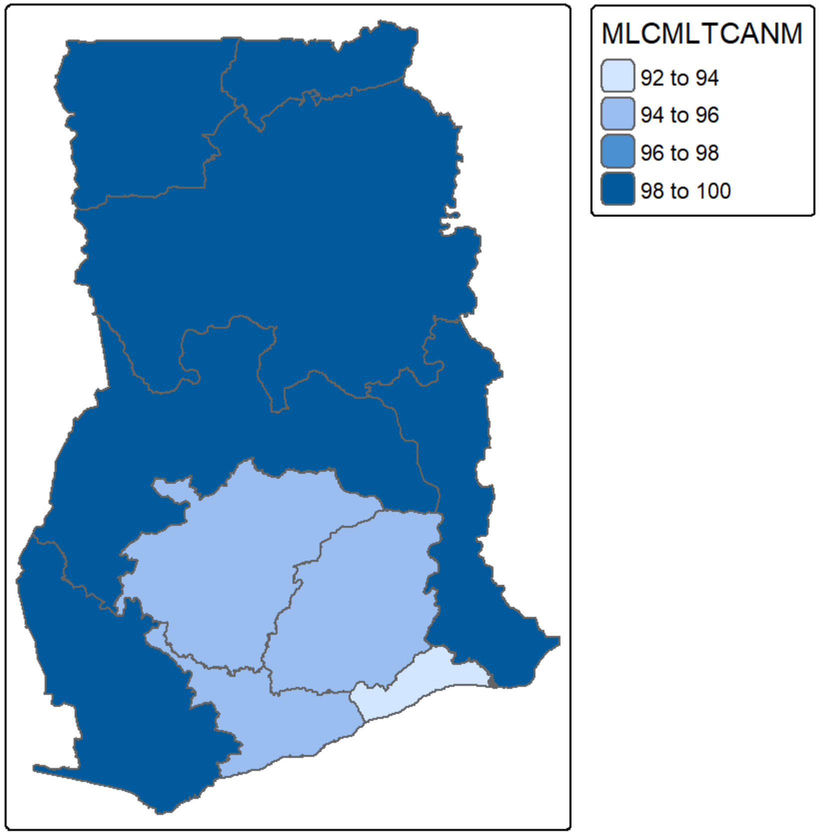
Mapping Percentage of children aged 6-59 months tested for

anemia: 2019 GMIS

```
spplot(mal_dat,"MLPMALCRDT")
```



```
## Mapping using tmap package (our focus)

## Mapping Percentage of children aged 6-59 months tested for anemia

mal_dat<-st_as_sf(mal_dat)

tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM")
```



```
# Create a beautiful map for Ghana using tmap package

summary(mal_dat$MLCMLTCANM)
```

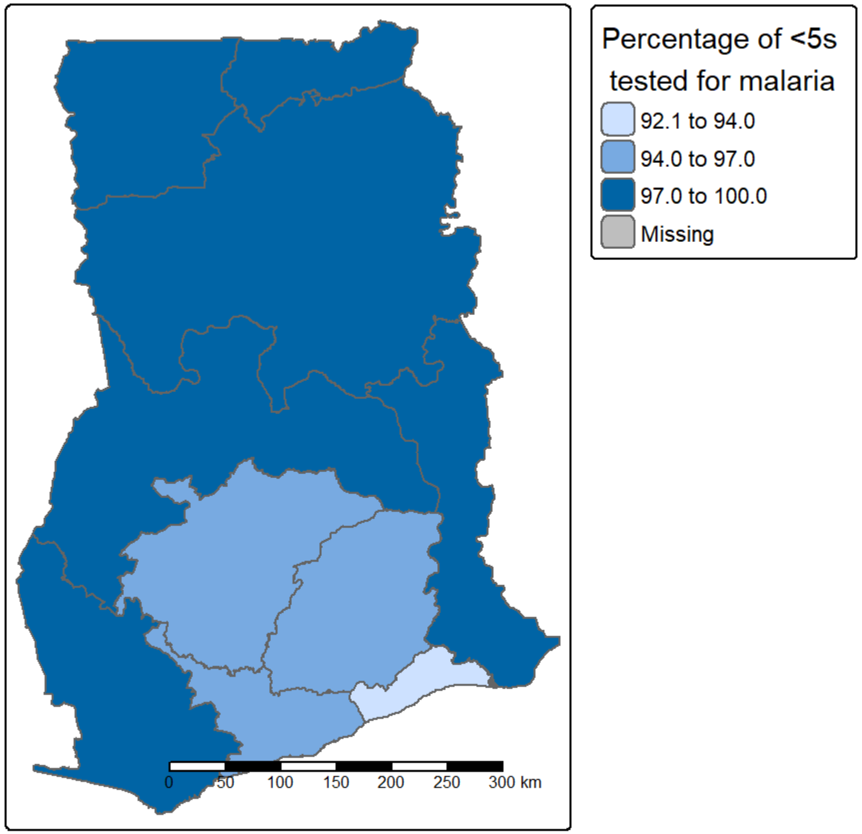
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   92.10   95.72   98.30   97.33   99.70  100.00
```

```
tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",title="Percentage of <5s \n tested for malaria",
              style = "fixed",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks' to 'tm_scale_intervals(<HERE>)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

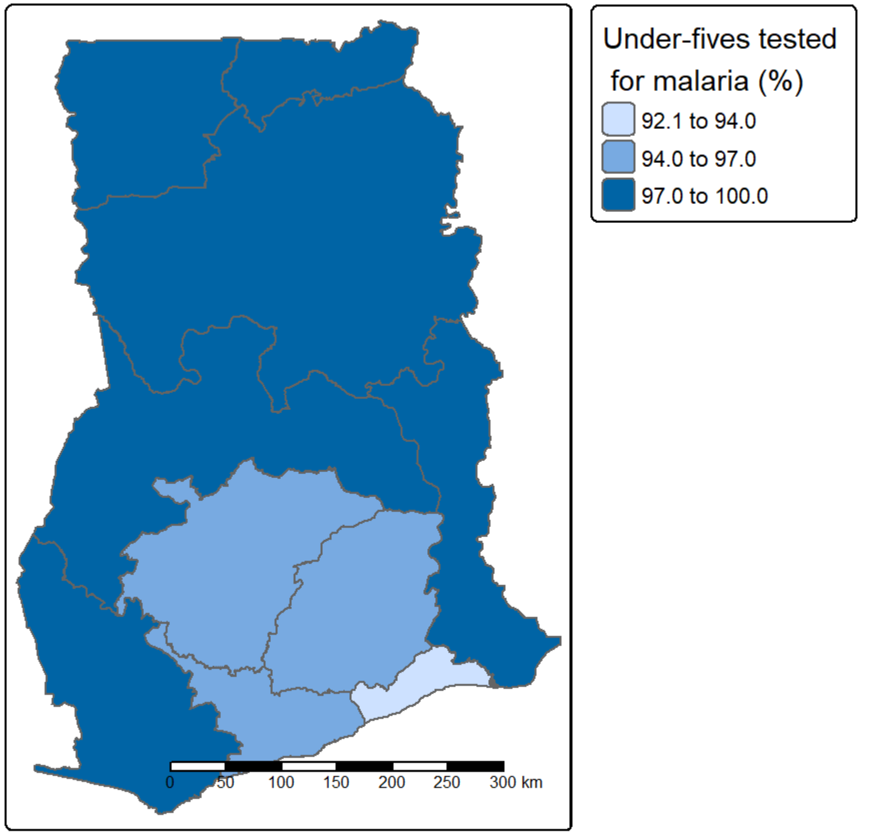



```
# Same but change the title and remove NAs in legend

tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",title="Under-fives tested \n for malaria (%)",
              style = "fixed",textNA="",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples,
specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```



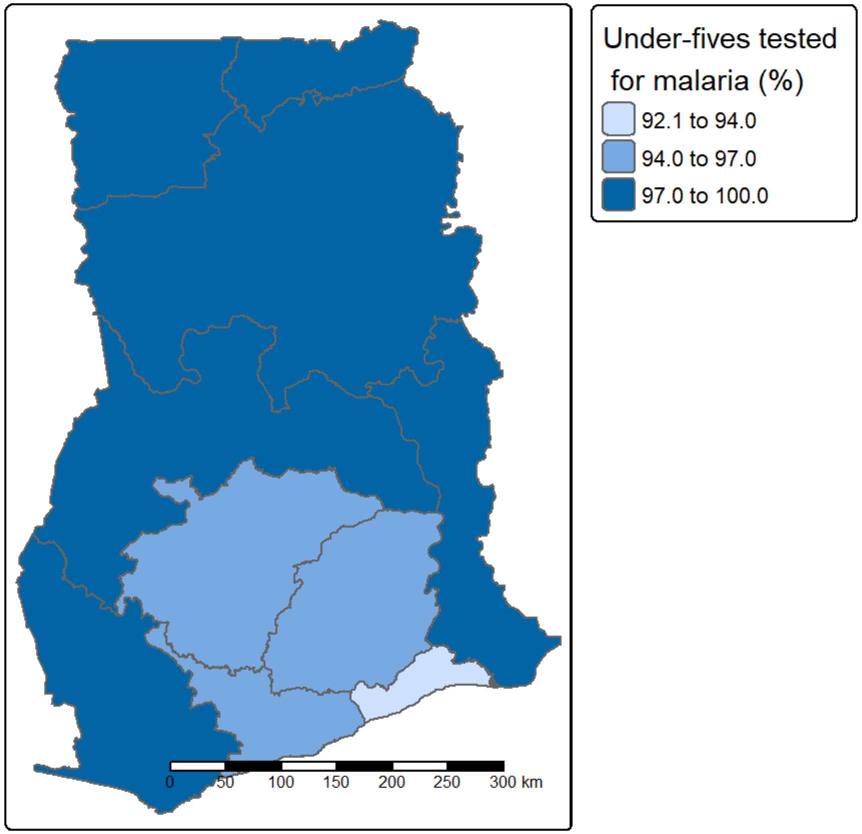
```
# Same but assign object name

mal_test<-tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",title="Under-fives tested \n for malaria (%)",
              style = "fixed",textNA="",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples,
specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_test
```



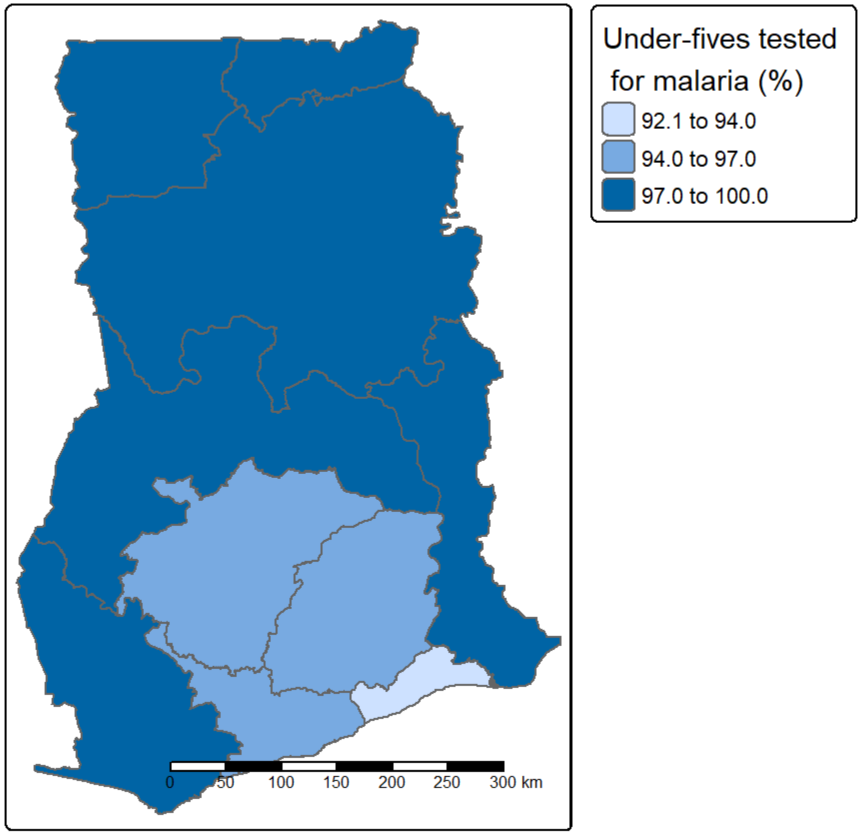

```
# Same but break the title and produce interactive web-based map

mal_test2<-tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",title="Under-fives tested \n for malaria (%)",
              style = "fixed",textNA="",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

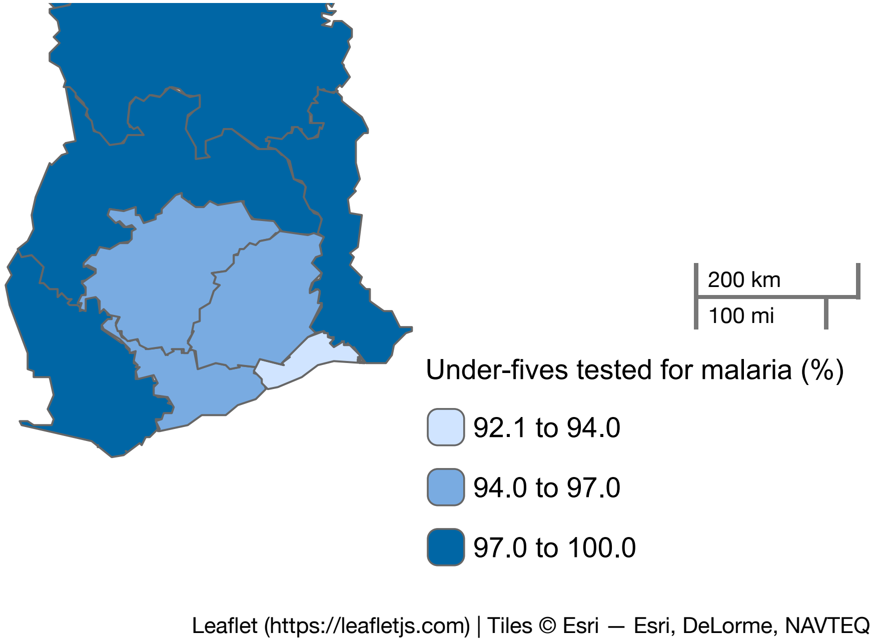
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples,
specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_test2
```




```
mal_test_web <- tmap_leaflet(mal_test2)
mal_test_web
```



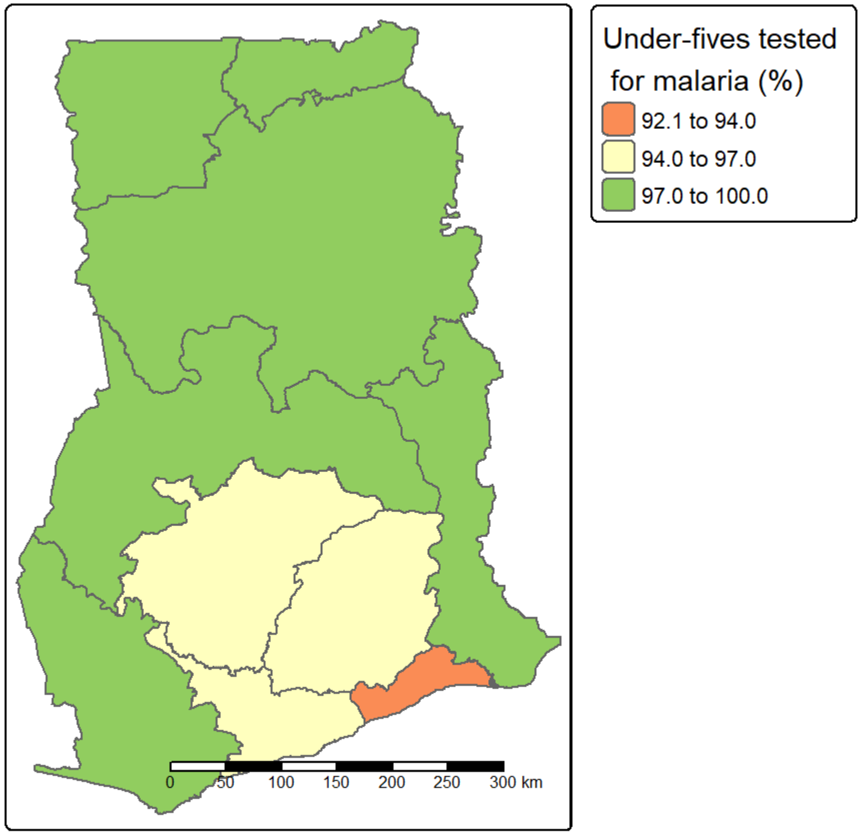

```
# Same but change color to suit the occasion
pal=brewer.pal(3,"RdYlGn")

mal_test2<-tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",palette=pal, title="Under-fives tested \n for malaria (%)",
              style = "fixed",textNA="",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

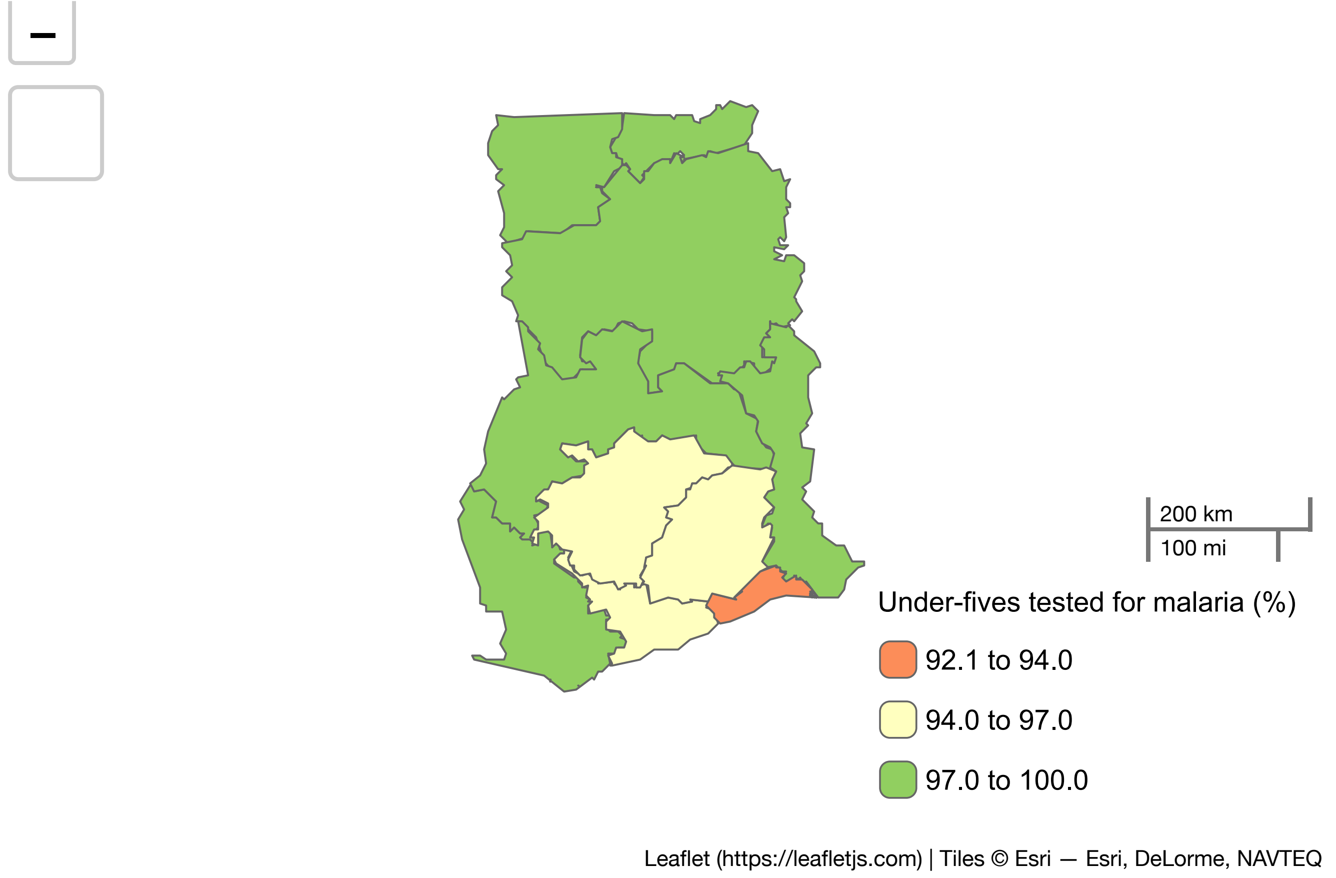
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'palette' (rename to 'values'), 'textNA' (rename to 'label.na') to 'tm_scale_interval
s(<HERE>)'. For small multiples, specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = l
ist(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_test2
```




```
mal_test2_web <- tmap_leaflet(mal_test2)
mal_test2_web
```



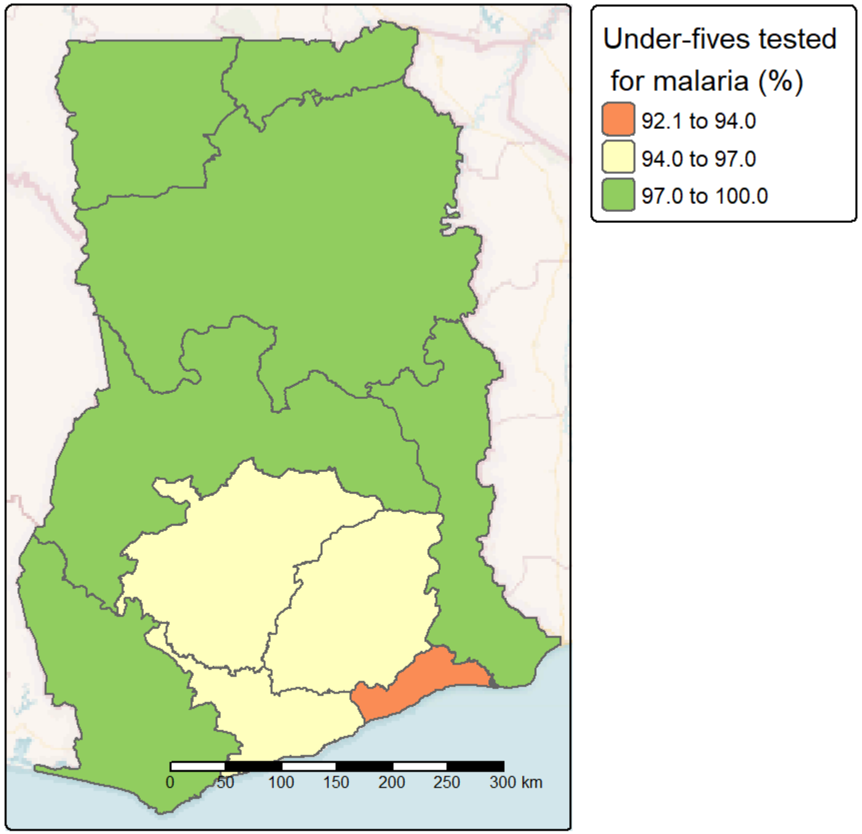

```
# Mapping without the basemap tiles

mal_test2<-tm_basemap(leaflet::providers$CartoDB.VoyagerNoLabels) +
tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",palette=pal, title="Under-fives tested \n for malaria (%)",
              style = "fixed",textNA="",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

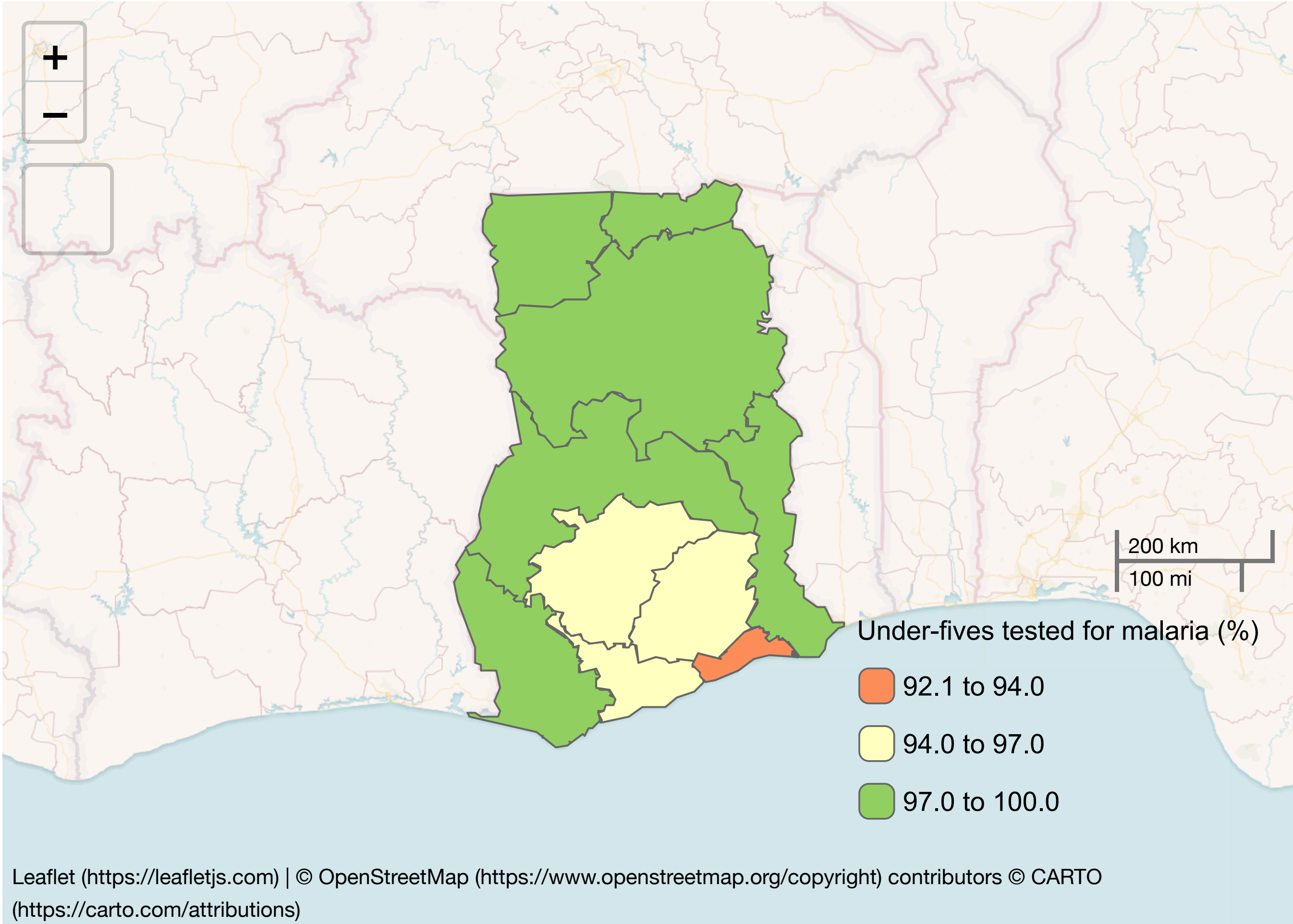
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'palette' (rename to 'values'), 'textNA' (rename to 'label.na') to 'tm_scale_interval
s(<HERE>)'. For small multiples, specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = l
ist(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_test2
```



```
mal_test2_web <- tmap_leaflet(mal_test2)
mal_test2_web
```



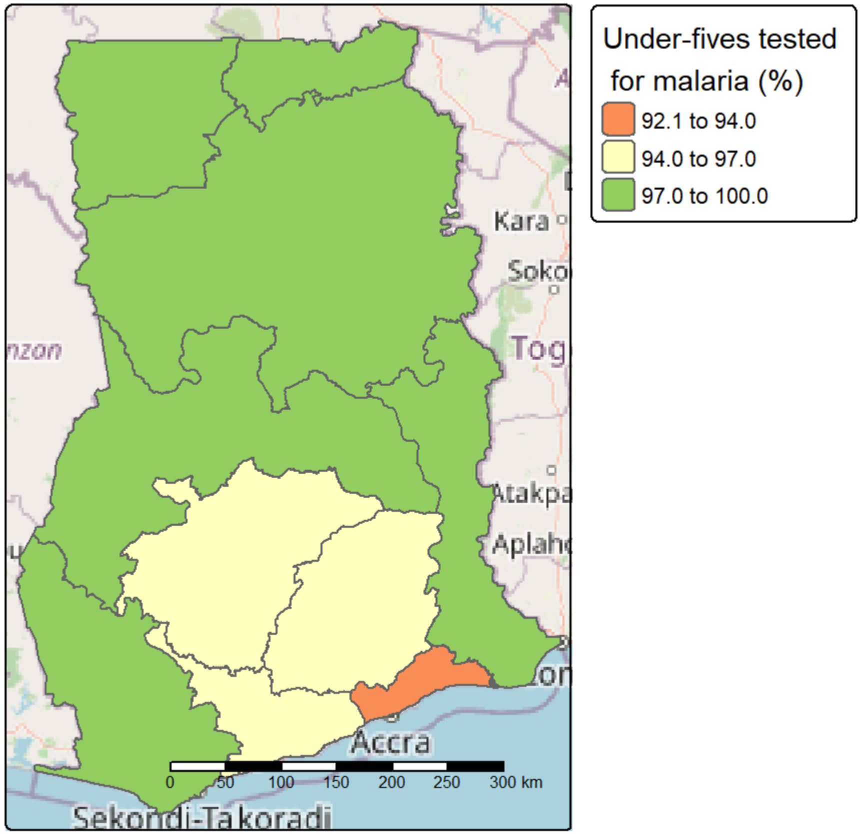
```
# Mapping with Open Street Map basemap tiles

mal_test3<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
tm_polygons("MLCMLTCANM",palette=pal, title="Under-fives tested \n for malaria (%)",
              style = "fixed",textNA="",
              breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

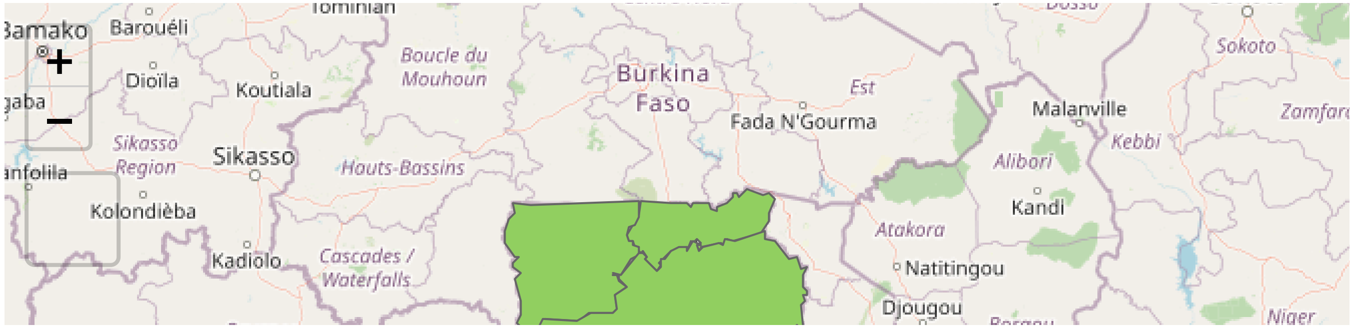
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'palette' (rename to 'values'), 'textNA' (rename to 'label.na') to 'tm_scale_interval
s(<HERE>)'. For small multiples, specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = l
ist(<scale1>, <scale2>, ...)'
```

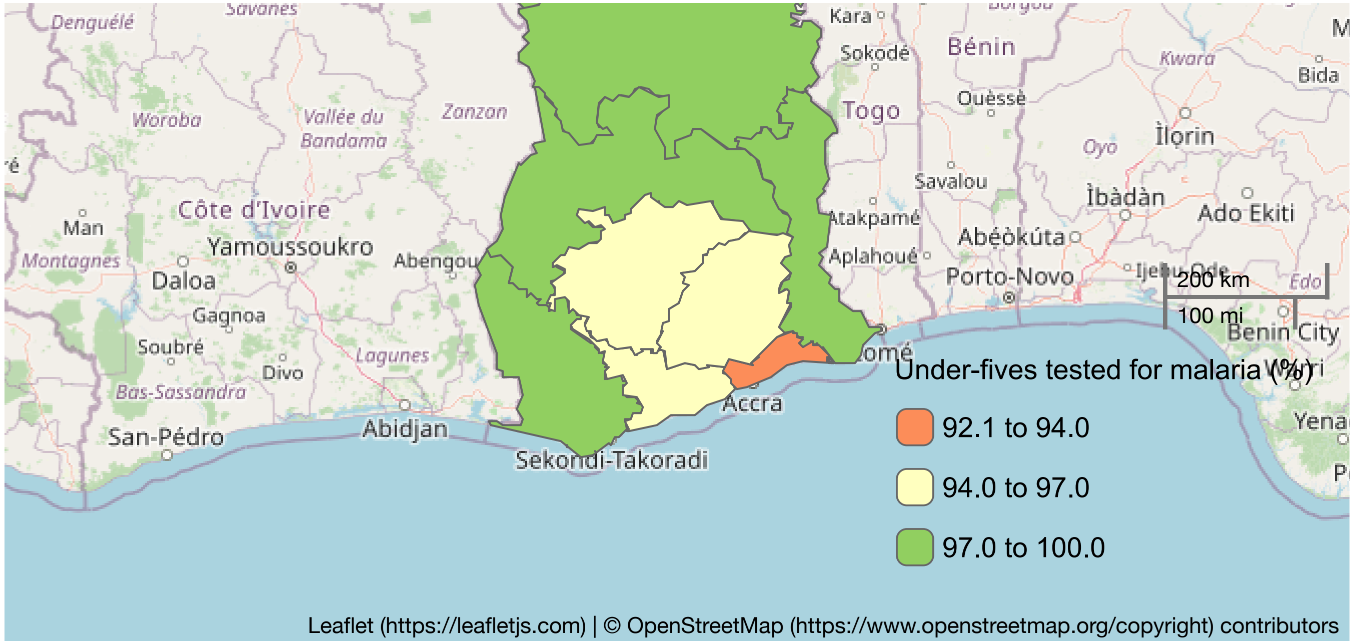
```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_test3
```




```
mal_test3_web <- tmap_leaflet(mal_test3)
mal_test3_web
```






```
## We can trick the system to use the actual percentages (see malaria prevalence example next)

## Mapping malaria prevalence (i.e., MLPMALCRDT) among children aged 6-59 months according to RDT

summary(mal_dat$MLPMALCRDT)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    1.00   19.68   28.15   24.49   31.05   35.40 
```

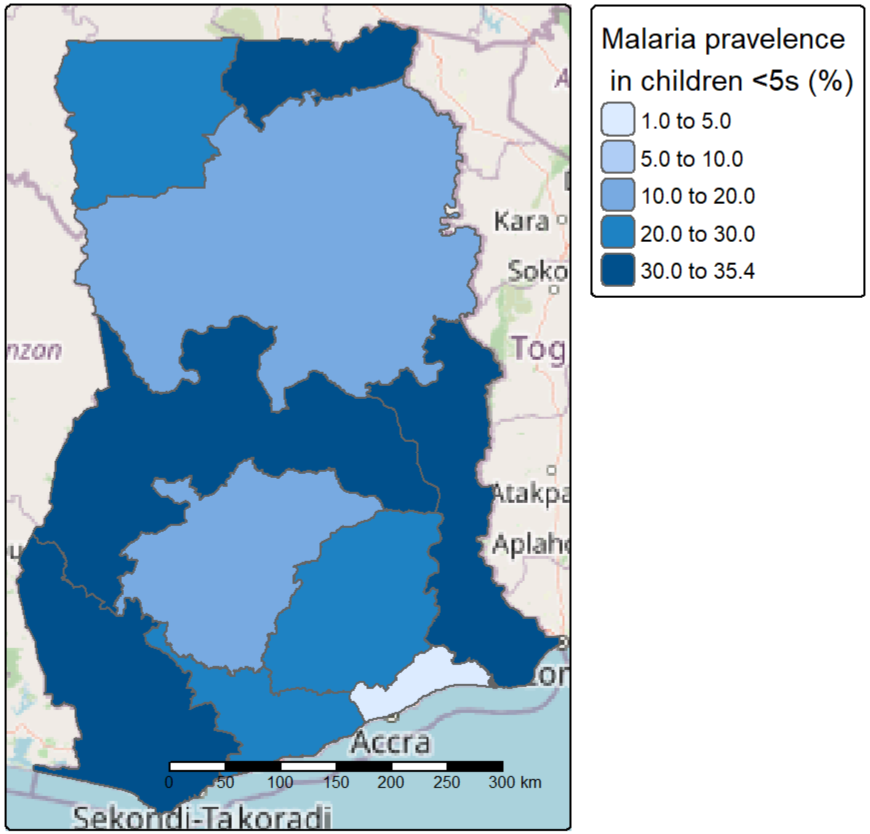
```
mal_prev1<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("MLPMALCRDT",title="Malaria pravelence \n in children <5s (%)",
             style = "fixed",textNA="",
             breaks = c(1,5.0,10.0,20.0,30.0,35.4),
             legend.hist = F) +
             tm_layout(legend.outside = TRUE)+
             tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

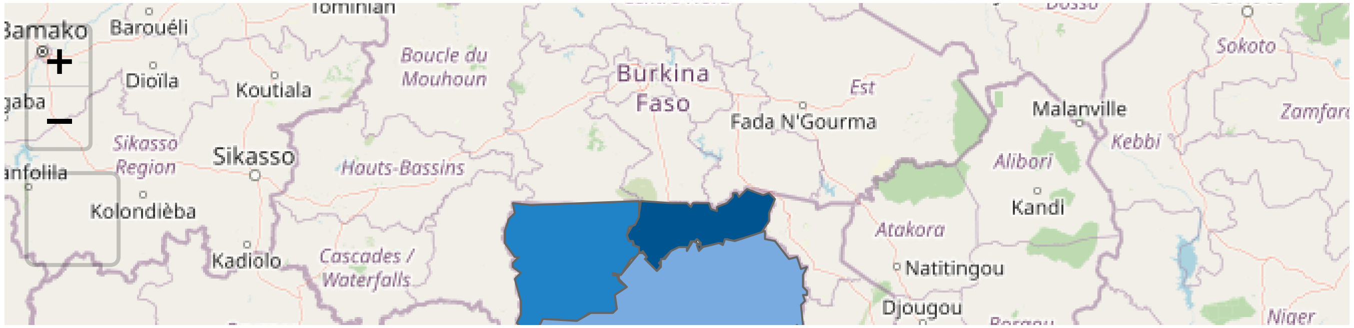
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples,
specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

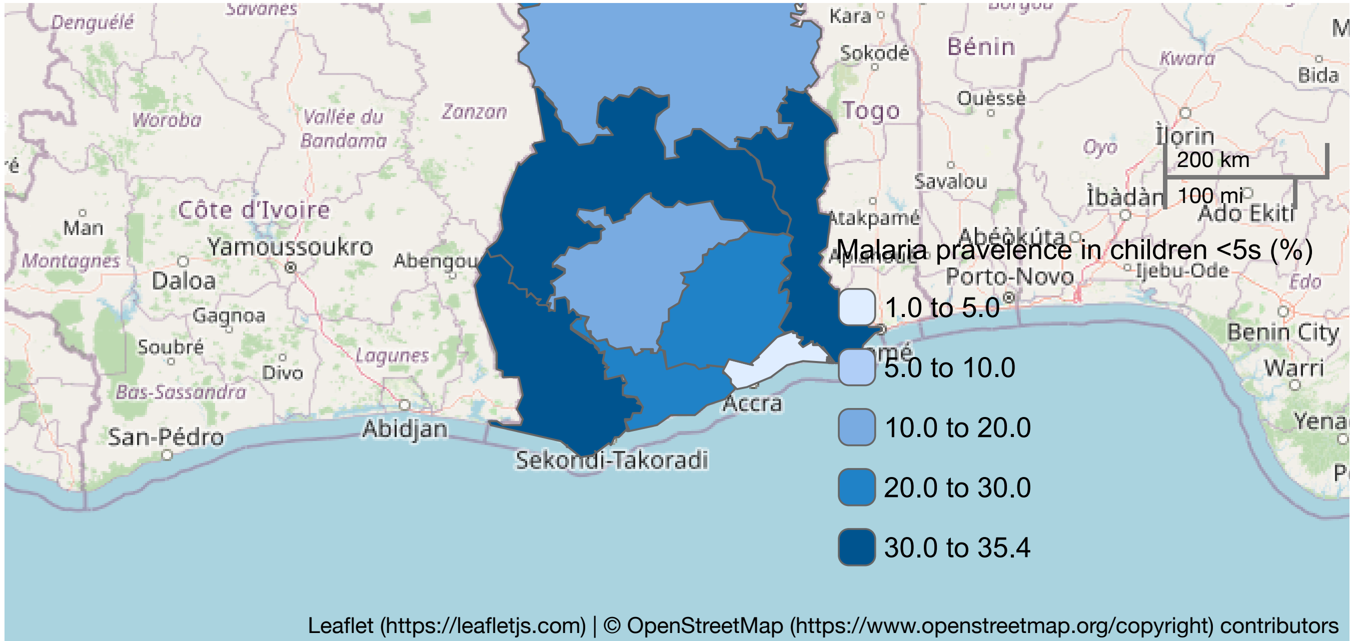
```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev1
```




```
mal_prev1_web <- tmap_leaflet(mal_prev1)
mal_prev1_web
```






```
# Same but change color to suit the occasion

pal2=brewer.pal(5,"RdYlGn")

summary(mal_dat$MLPMALCRDT)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   19.68   28.15   24.49   31.05   35.40
```

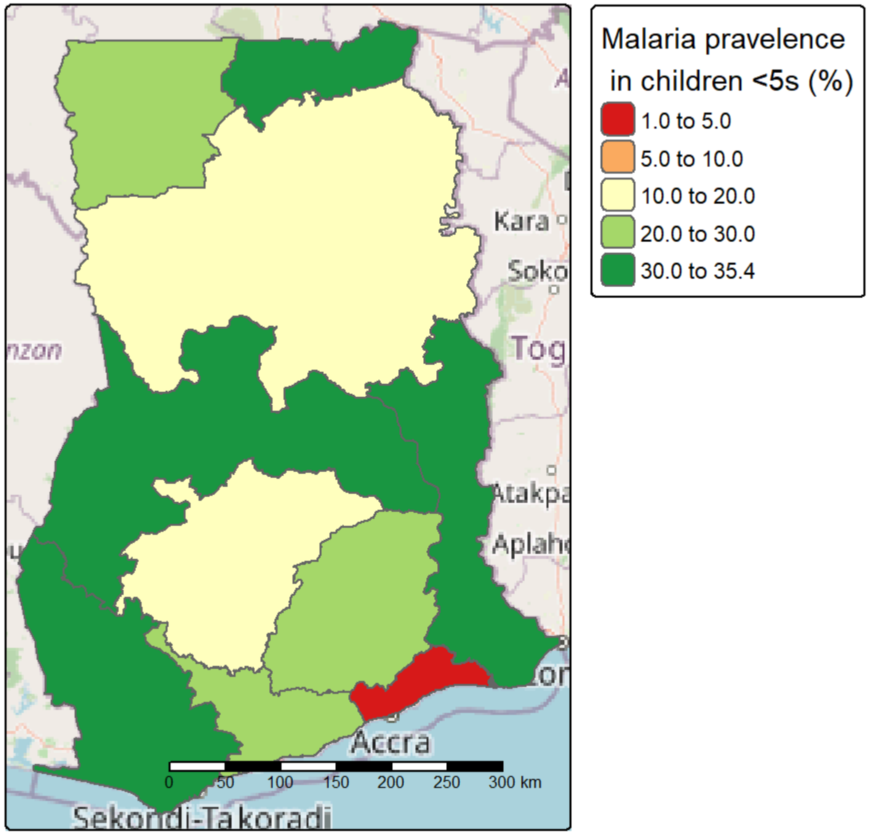
```
mal_prev1<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("MLPMALCRDT",palette=pal2,title="Malaria pravelence \n in children <5s (%)",
             style = "fixed",textNA="",
             breaks = c(1,5.0,10.0,20.0,30.0,35.4),
             legend.hist = F) +
             tm_layout(legend.outside = TRUE)+
             tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

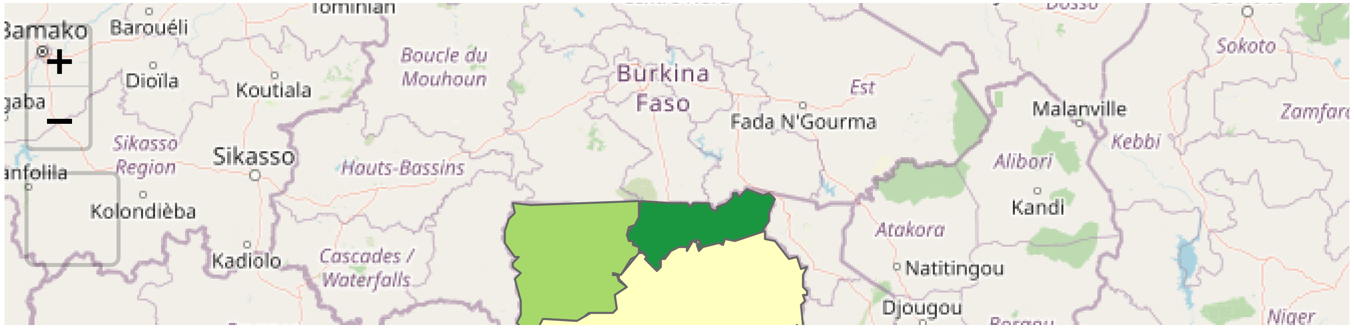
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'palette' (rename to 'values'), 'textNA' (rename to 'label.na') to 'tm_scale_interval
s(<HERE>)'. For small multiples, specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = l
ist(<scale1>, <scale2>, ...)'
```

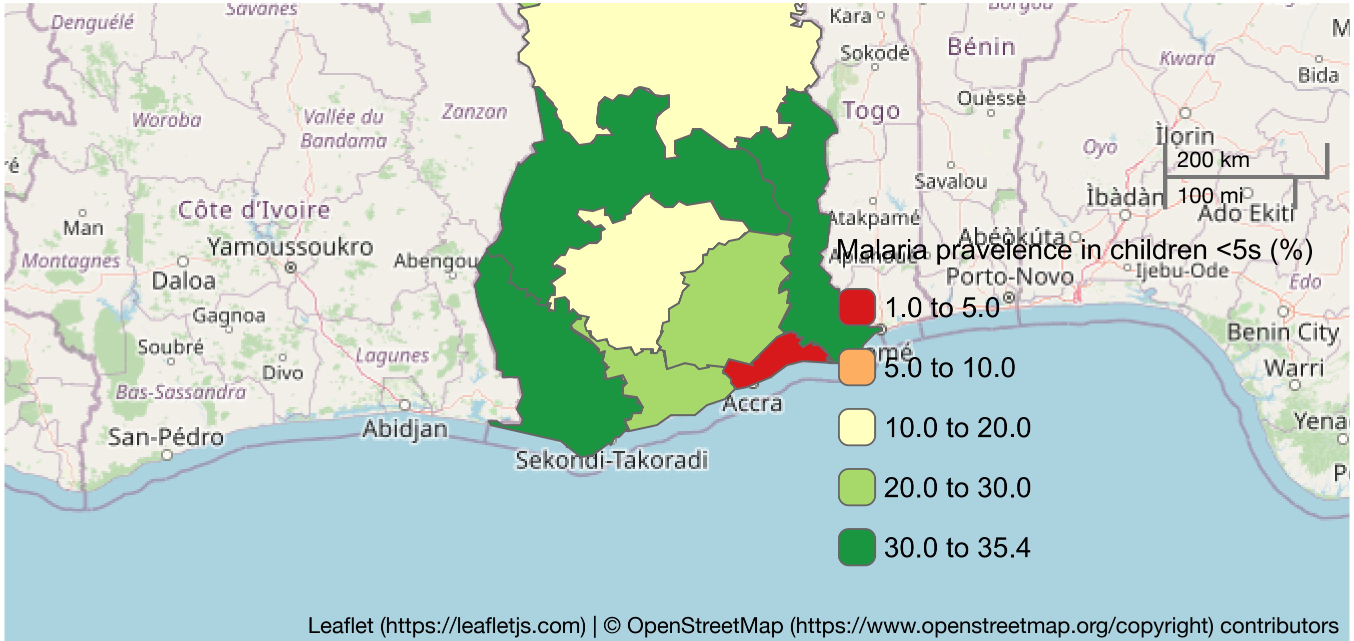
```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev1
```




```
mal_prev1_web <- tmap_leaflet(mal_prev1)
mal_prev1_web
```





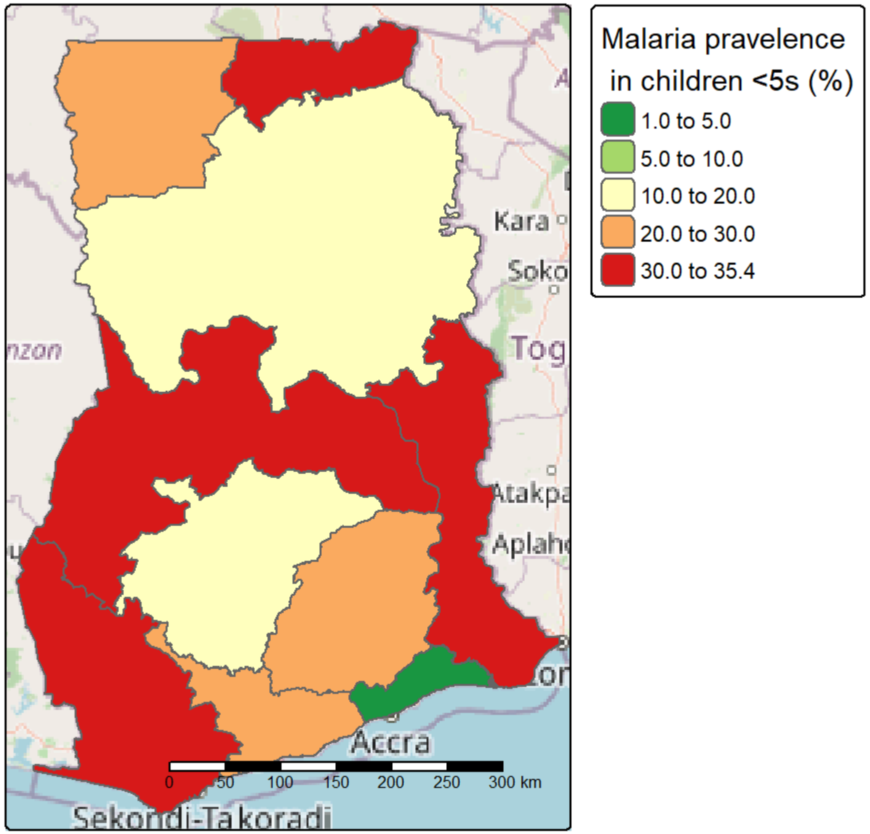

```
# reverse color to make red represents worst case scenario
pal2=rev(brewer.pal(5,"RdYlGn")) # note reversing color here

# x11()# divide the result screen into 2 for the second map to aid comparison

mal_prev2<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("MLPMALCRDT",palette=pal2,title="Malaria pravelence \n in children <5s (%)",
             style = "fixed",textNA="",
             breaks = c(1,5.0,10.0,20.0,30.0,35.4),
             legend.hist = F) +
             tm_layout(legend.outside = TRUE)+
             tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

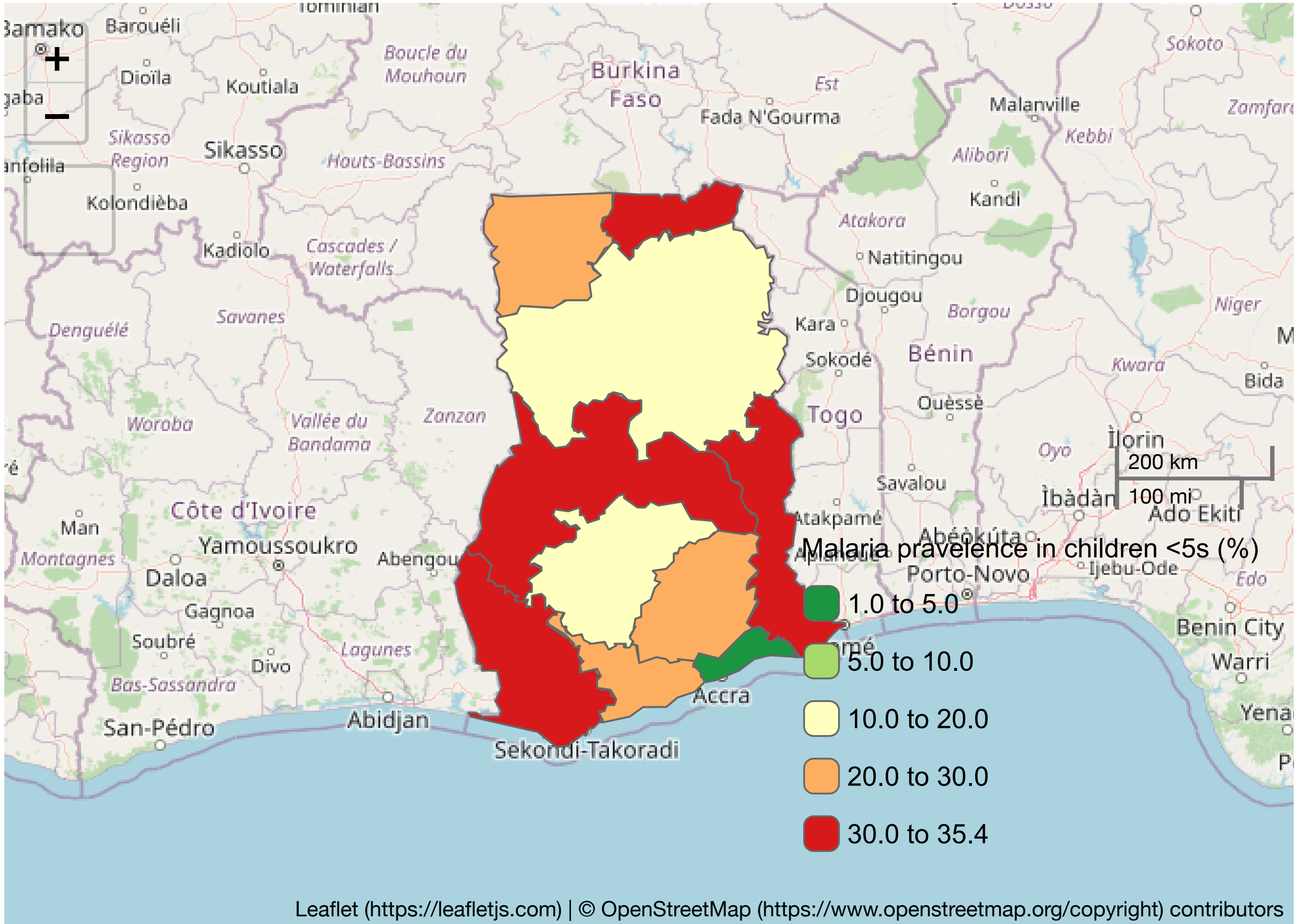
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'palette' (rename to 'values'), 'textNA' (rename to 'label.na') to 'tm_scale_interval
s(<HERE>)'. For small multiples, specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = l
ist(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev2
```



```
mal_prev2_web <- tmap_leaflet(mal_prev2)
mal_prev2_web
```



Same as above but without tiles (some journal requirement for publication)

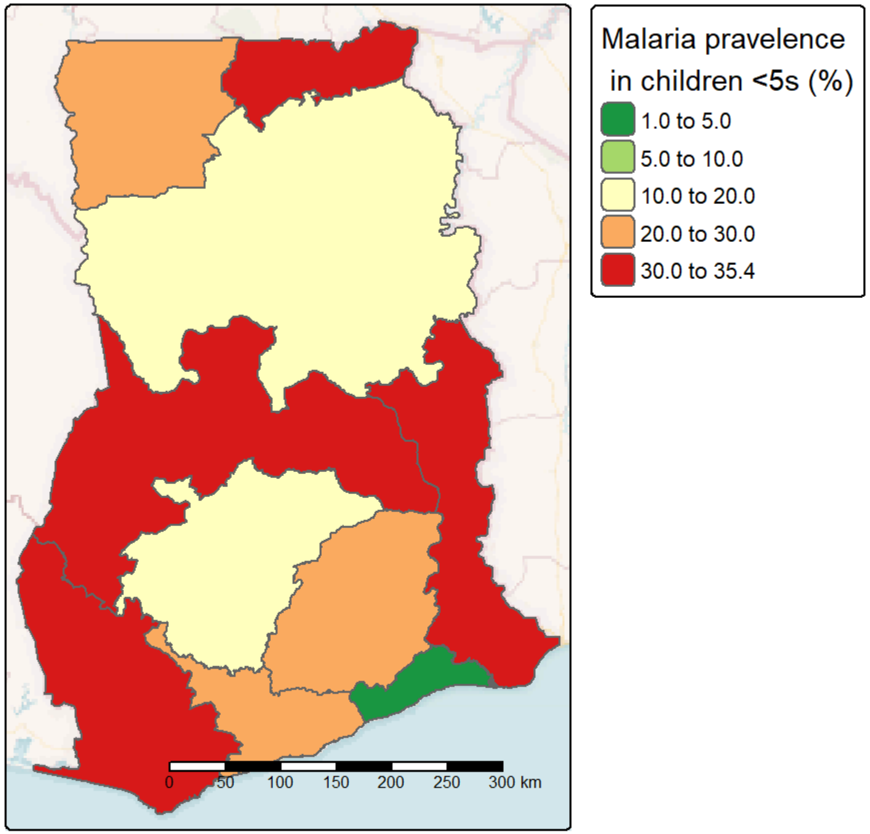
```
mal_prev3<-tm_basemap(leaflet::providers$CartoDB.VoyagerNoLabels) +
tm_shape(mal_dat)+
tm_polygons("MLPMALCRDT",palette=pal2,title="Malaria pravelence \n in children <5s (%)",
             style = "fixed",textNA="",
             breaks = c(1,5.0,10.0,20.0,30.0,35.4),
             legend.hist = F) +
             tm_layout(legend.outside = TRUE)+
             tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

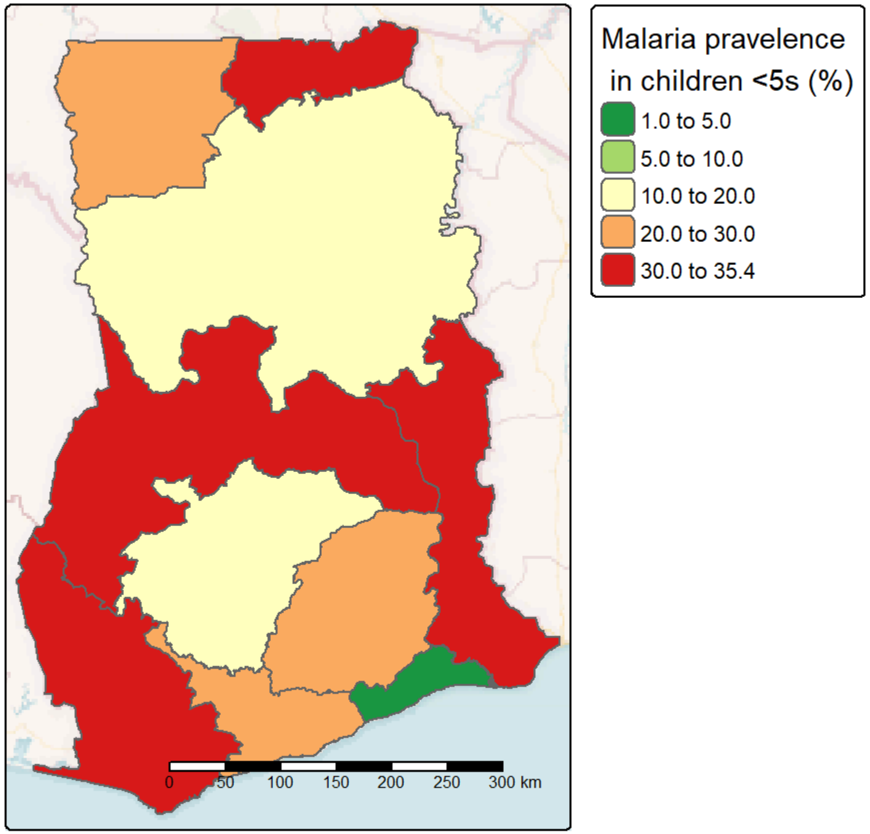
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks', 'palette' (rename to 'values'), 'textNA' (rename to 'label.na') to 'tm_scale_interval
s(<HERE>)'. For small multiples, specify a 'tm_scale_' for each multiple, and put them in a list: 'fill.scale = l
ist(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev3
```




```
mal_prev3_web <- tmap_leaflet(mal_prev3)
mal_prev3
```



Same as above but insert the exact percentages inside the respective regions

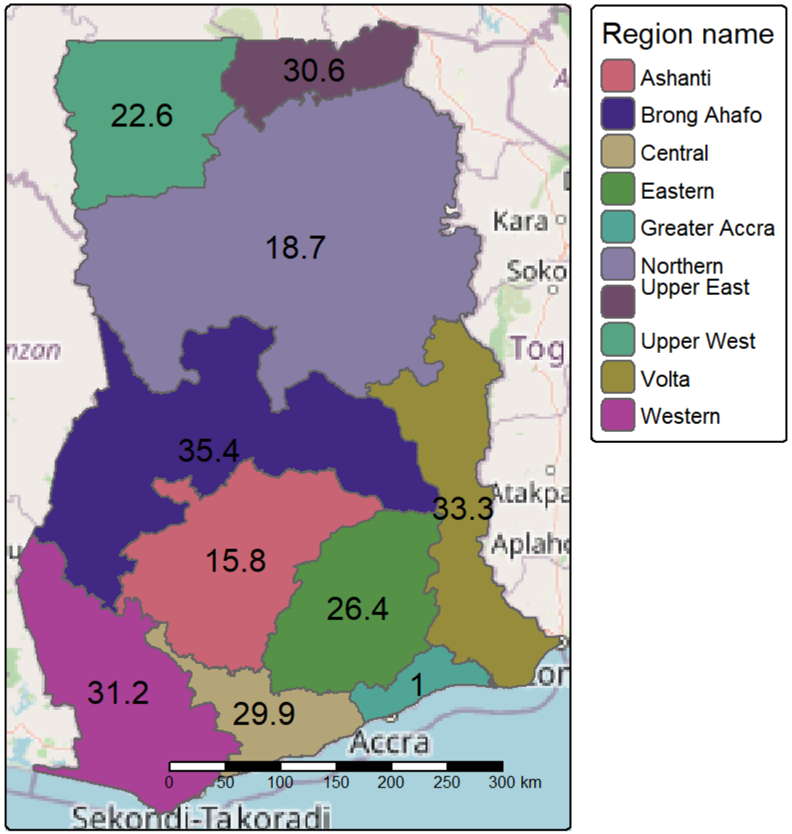
```
mal_prev4<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("REGNAME",title="Region name",
              style = "fixed",textNA="",
              #breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_text("MLPMALCRDT",size=1)+
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

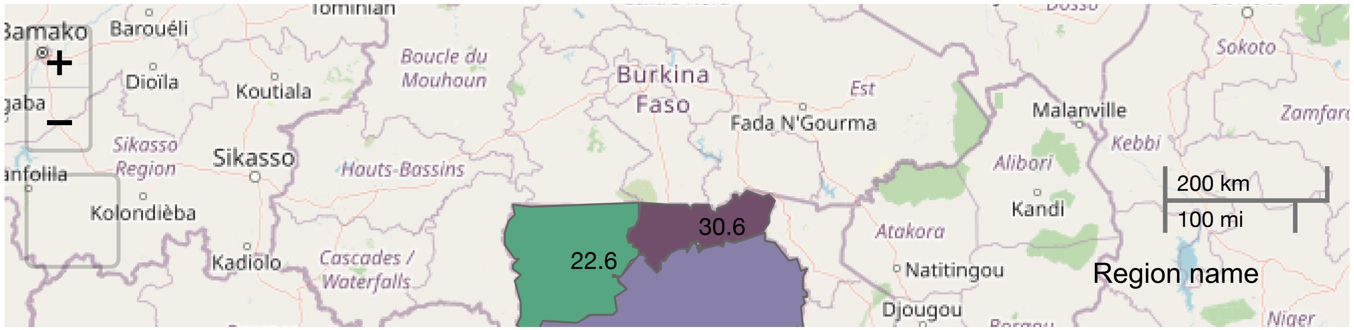
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples, specify a
'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

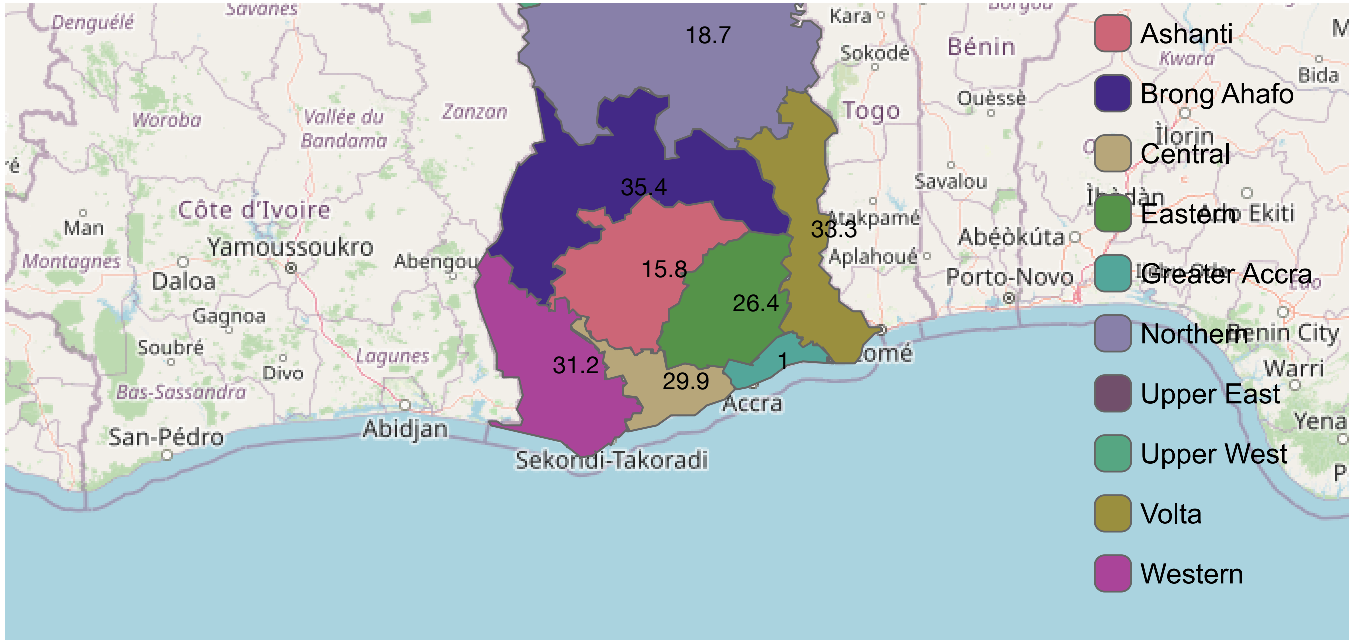
```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev4
```




```
mal_prev4_web <- tmap_leaflet(mal_prev4)
mal_prev4_web
```





Let us trick the system to include the % sign inside the regions

Few manipulations required to do this as below:

```
mal_dat$REGNAME # to see the position of region names
```

```
##  [1] "Upper East\r\n" "Western"        "Upper West"     "Northern"
##  [5] "Brong Ahafo"    "Volta"          "Central"        "Greater Accra"
##  [9] "Ashanti"        "Eastern"
```

```
table(mal_dat$REGNAME,mal_dat$MLPMALCRDT) # Or check from map above
```

```
##
##                  1 15.8 18.7 22.6 26.4 29.9 30.6 31.2 33.3 35.4
##   Ashanti        0    1    0    0    0    0    0    0    0    0
##   Brong Ahafo    0    0    0    0    0    0    0    0    0    1
##   Central        0    0    0    0    0    1    0    0    0    0
##   Eastern        0    0    0    0    1    0    0    0    0    0
##   Greater Accra  1    0    0    0    0    0    0    0    0    0
##   Northern       0    0    1    0    0    0    0    0    0    0
##   Upper East\r\n 0    0    0    0    0    0    1    0    0    0
##   Upper West     0    0    0    1    0    0    0    0    0    0
##   Volta          0    0    0    0    0    0    0    0    1    0
##   Western        0    0    0    0    0    0    0    1    0    0
```

```
mal_dat$MLPMALCRDT_p<-c("30.6%","31.2%","22.6%","18.7%","35.4%","33.3%",
                             "29.9%","1.0%","15.8%","26.4%")

summary(mal_dat)
```

```
##      ISO                FIPS               DHSCC              SVYTYPE
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     SVYYEAR      CNTRYNAMEE         CNTRYNAMEF         CNTRYNAMES
##  Min.   :2019   Length:10          Length:10          Length:10
##  1st Qu.:2019   Class :character   Class :character   Class :character
##  Median :2019   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2019
##  3rd Qu.:2019
##  Max.   :2019
##    DHSREGEN           DHSREGFR           DHSREGSP              SVYID
##  Length:10          Length:10          Length:10          Min.   :557
##  Class :character   Class :character   Class :character   1st Qu.:557
##  Mode  :character   Mode  :character   Mode  :character   Median :557
##                                                           Mean   :557
##                                                           3rd Qu.:557
##                                                           Max.   :557
##     REG_ID            Svy_Map           MULTLEVEL            LEVELRNK
##  Length:10          Length:10          Length:10          Min.   :1
##  Class :character   Class :character   Class :character   1st Qu.:1
##  Mode  :character   Mode  :character   Mode  :character   Median :1
##                                                           Mean   :1
##                                                           3rd Qu.:1
##                                                           Max.   :1
##     REGVAR             REGCODE         REGNAME            OTHREGVAR
##  Length:10          Min.   : 1.00   Length:10          Length:10
##  Class :character   1st Qu.: 3.25   Class :character   Class :character
##  Mode  :character   Median : 5.50   Mode  :character   Mode  :character
##                     Mean   : 5.50
##                     3rd Qu.: 7.75
##                     Max.   :10.00
##     OTHREGCO      OTHREGNA            LEVELCO            LEVELNA
##  Min.   :9999   Length:10          Length:10          Length:10
##  1st Qu.:9999   Class :character   Class :character   Class :character
```

```
##  Median :9999   Mode  :character   Mode  :character   Mode  :character
##  Mean   :9999
##  3rd Qu.:9999
##  Max.   :9999
##   REPALLIND           REGNOTES           SVYNOTES           MLCMLTCANM
##  Length:10          Length:10          Length:10          Min.   : 92.10
##  Class :character   Class :character   Class :character   1st Qu.: 95.72
##  Mode  :character   Mode  :character   Mode  :character   Median : 98.30
##                                                           Mean   : 97.33
##                                                           3rd Qu.: 99.70
##                                                           Max.   :100.00
##    MLCMLTCRDT       MLCMLTCMSY       MLCMLTCNUM      MLCMLTCUNW
##  Min.   : 91.60   Min.   : 92.10   Min.   :203.0   Min.   :9999
##  1st Qu.: 95.12   1st Qu.: 95.47   1st Qu.:256.0   1st Qu.:9999
##  Median : 98.30   Median : 98.15   Median :274.5   Median :9999
##  Mean   : 97.08   Mean   : 97.21   Mean   :292.0   Mean   :9999
##  3rd Qu.: 99.00   3rd Qu.: 99.58   3rd Qu.:300.0   3rd Qu.:9999
##  Max.   :100.00   Max.   :100.00   Max.   :518.0   Max.   :9999
##    MLPMALCRDT      MLPMALCRDE      MLPMALCRDR      MLPMALCRDL
##  Min.   : 1.00   Min.   :0.800   Min.   :0.200   Min.   : 0.00
##  1st Qu.:19.68   1st Qu.:4.400   1st Qu.:0.200   1st Qu.:12.12
##  Median :28.15   Median :4.750   Median :0.200   Median :15.75
##  Mean   :24.49   Mean   :4.800   Mean   :0.280   Mean   :14.94
##  3rd Qu.:31.05   3rd Qu.:5.175   3rd Qu.:0.275   3rd Qu.:20.75
##  Max.   :35.40   Max.   :9.000   Max.   :0.800   Max.   :23.70
##    MLPMALCRDU      MLPMALCNMR      MLPMALCUNR      MLPMALCUER
##  Min.   : 2.60   Min.   : 83.0   Min.   :186.0   Min.   :203.0
##  1st Qu.:27.25   1st Qu.:197.2   1st Qu.:250.2   1st Qu.:256.0
##  Median :37.95   Median :275.0   Median :270.5   Median :274.5
##  Mean   :34.11   Mean   :261.1   Mean   :284.3   Mean   :292.0
##  3rd Qu.:41.27   3rd Qu.:303.2   3rd Qu.:294.2   3rd Qu.:300.0
##  Max.   :51.20   Max.   :421.0   Max.   :509.0   Max.   :518.0
##    MLPMALCMSY      MLPMALCMSE      MLPMALCMSR     MLPMALCMSL      MLPMALCMSU
##  Min.   : 2.40   Min.   :1.300   Min.   :0.20   Min.   : 0.00   Min.   : 5.00
##  1st Qu.:10.43   1st Qu.:2.750   1st Qu.:0.20   1st Qu.: 4.30   1st Qu.:16.05
##  Median :12.65   Median :3.400   Median :0.25   Median : 5.45   Median :21.10
##  Mean   :14.07   Mean   :3.260   Mean   :0.29   Mean   : 7.58   Mean   :20.62
##  3rd Qu.:17.52   3rd Qu.:4.075   3rd Qu.:0.30   3rd Qu.:10.78   3rd Qu.:24.32
```

```
##  Max.   :27.00   Max.   :4.300   Max.   :0.60   Max.   :18.40   Max.   :35.60
##    MLPMALCNMM      MLPMALCUNM      MLPMALCUEM             geometry
##  Min.   : 83.0   Min.   :187.0   Min.   :203.0   MULTIPOLYGON :10
##  1st Qu.:197.0   1st Qu.:249.2   1st Qu.:256.0   epsg:4326    : 0
##  Median :276.5   Median :271.5   Median :274.5   +proj=long...: 0
##  Mean   :261.8   Mean   :284.6   Mean   :292.0
##  3rd Qu.:305.5   3rd Qu.:294.2   3rd Qu.:300.0
##  Max.   :421.0   Max.   :509.0   Max.   :518.0
##  MLPMALCRDT_p
##  Length:10
##  Class :character
##  Mode  :character
##
##
##
```

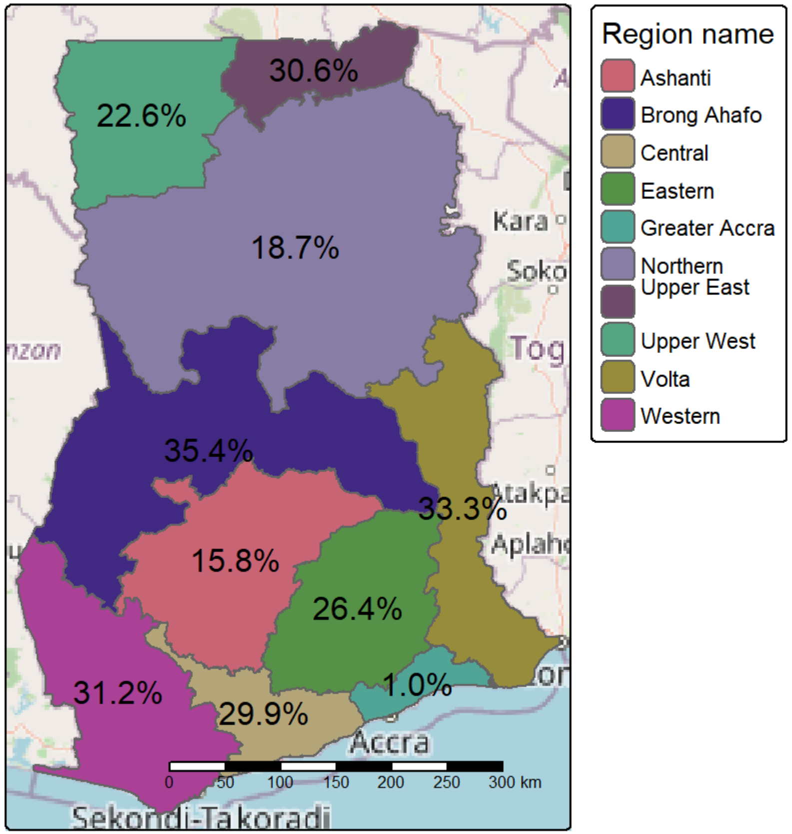
```
# Same but insert the percentage sign inside the respective regions

mal_prev5<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("REGNAME",title="Region name",
              style = "fixed",textNA="",
              #breaks = c(92.1,94.0,97.0,100.0),
              legend.hist = F) +
              tm_text("MLPMALCRDT_p",size=1)+
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

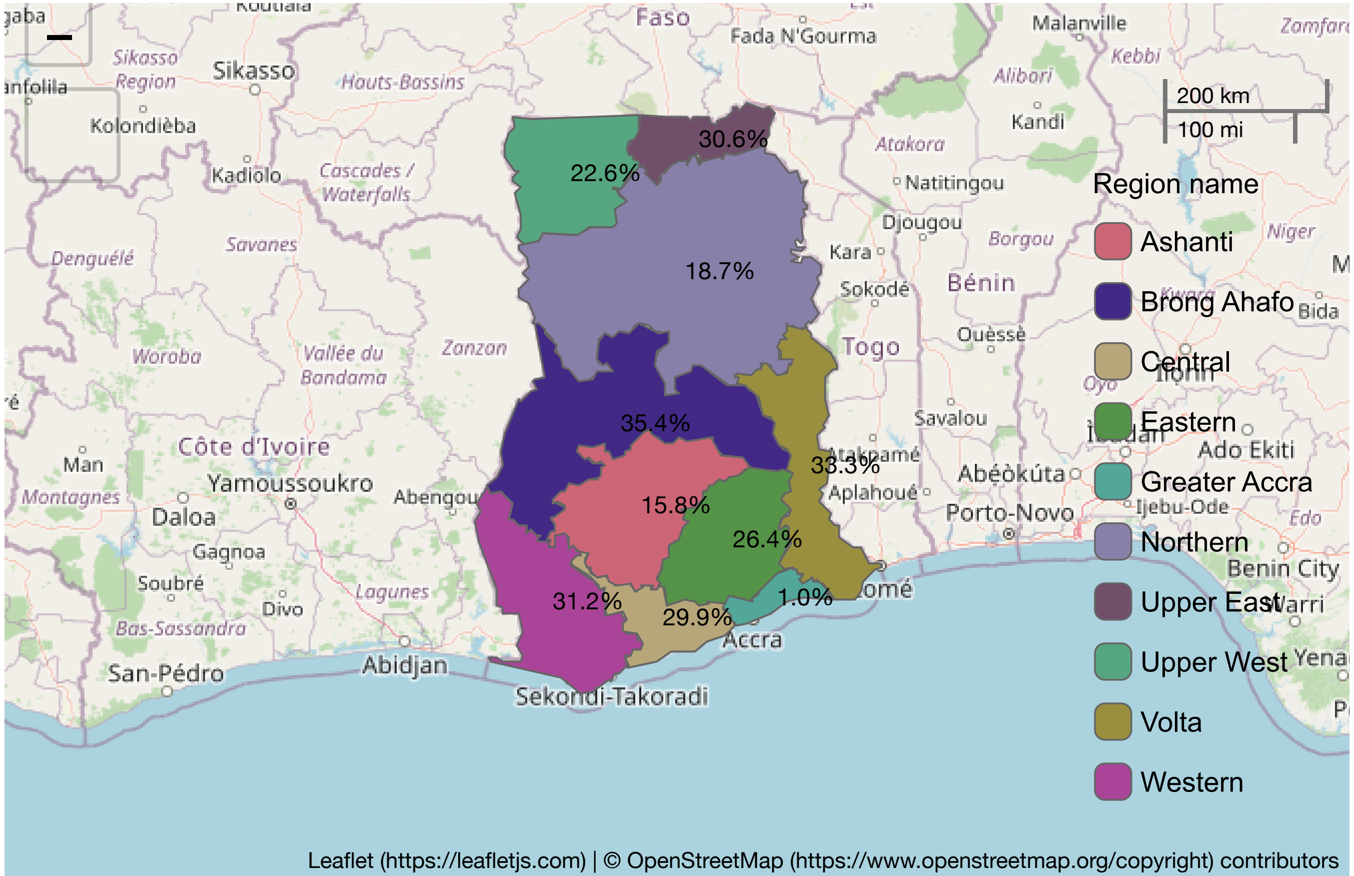
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples, specify a
'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev5
```




```
mal_prev5_web <- tmap_leaflet(mal_prev5)
mal_prev5_web
```



Same as above, easier data manipulation less human error, and saves time when you have several data points but technically heavy!

```
mal_dat<-mal_dat%>%mutate(MLPMALCRDT_per=paste0(round(MLPMALCRDT,1),"%"))
summary(mal_dat)
```

```
##      ISO                FIPS               DHSCC              SVYTYPE
##  Length:10          Length:10          Length:10          Length:10
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     SVYYEAR      CNTRYNAMEE         CNTRYNAMEF         CNTRYNAMES
##  Min.   :2019   Length:10          Length:10          Length:10
##  1st Qu.:2019   Class :character   Class :character   Class :character
##  Median :2019   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2019
##  3rd Qu.:2019
##  Max.   :2019
##    DHSREGEN           DHSREGFR           DHSREGSP             SVYID
##  Length:10          Length:10          Length:10          Min.   :557
##  Class :character   Class :character   Class :character   1st Qu.:557
##  Mode  :character   Mode  :character   Mode  :character   Median :557
##                                                           Mean   :557
##                                                           3rd Qu.:557
##                                                           Max.   :557
##     REG_ID            Svy_Map           MULTLEVEL            LEVELRNK
##  Length:10          Length:10          Length:10          Min.   :1
##  Class :character   Class :character   Class :character   1st Qu.:1
##  Mode  :character   Mode  :character   Mode  :character   Median :1
##                                                           Mean   :1
##                                                           3rd Qu.:1
##                                                           Max.   :1
##     REGVAR             REGCODE        REGNAME            OTHREGVAR
##  Length:10          Min.   : 1.00   Length:10          Length:10
##  Class :character   1st Qu.: 3.25   Class :character   Class :character
##  Mode  :character   Median : 5.50   Mode  :character   Mode  :character
##                     Mean   : 5.50
##                     3rd Qu.: 7.75
##                     Max.   :10.00
##     OTHREGCO      OTHREGNA            LEVELCO            LEVELNA
##  Min.   :9999   Length:10          Length:10          Length:10
##  1st Qu.:9999   Class :character   Class :character   Class :character
```

```
##  Median :9999   Mode  :character   Mode  :character   Mode  :character
##  Mean   :9999
##  3rd Qu.:9999
##  Max.   :9999
##   REPALLIND           REGNOTES           SVYNOTES           MLCMLTCANM
##  Length:10          Length:10          Length:10          Min.   : 92.10
##  Class :character   Class :character   Class :character   1st Qu.: 95.72
##  Mode  :character   Mode  :character   Mode  :character   Median : 98.30
##                                                           Mean   : 97.33
##                                                           3rd Qu.: 99.70
##                                                           Max.   :100.00
##    MLCMLTCRDT       MLCMLTCMSY       MLCMLTCNUM      MLCMLTCUNW
##  Min.   : 91.60   Min.   : 92.10   Min.   :203.0   Min.   :9999
##  1st Qu.: 95.12   1st Qu.: 95.47   1st Qu.:256.0   1st Qu.:9999
##  Median : 98.30   Median : 98.15   Median :274.5   Median :9999
##  Mean   : 97.08   Mean   : 97.21   Mean   :292.0   Mean   :9999
##  3rd Qu.: 99.00   3rd Qu.: 99.58   3rd Qu.:300.0   3rd Qu.:9999
##  Max.   :100.00   Max.   :100.00   Max.   :518.0   Max.   :9999
##    MLPMALCRDT      MLPMALCRDE      MLPMALCRDR      MLPMALCRDL
##  Min.   : 1.00   Min.   :0.800   Min.   :0.200   Min.   : 0.00
##  1st Qu.:19.68   1st Qu.:4.400   1st Qu.:0.200   1st Qu.:12.12
##  Median :28.15   Median :4.750   Median :0.200   Median :15.75
##  Mean   :24.49   Mean   :4.800   Mean   :0.280   Mean   :14.94
##  3rd Qu.:31.05   3rd Qu.:5.175   3rd Qu.:0.275   3rd Qu.:20.75
##  Max.   :35.40   Max.   :9.000   Max.   :0.800   Max.   :23.70
##    MLPMALCRDU      MLPMALCNMR      MLPMALCUNR      MLPMALCUER
##  Min.   : 2.60   Min.   : 83.0   Min.   :186.0   Min.   :203.0
##  1st Qu.:27.25   1st Qu.:197.2   1st Qu.:250.2   1st Qu.:256.0
##  Median :37.95   Median :275.0   Median :270.5   Median :274.5
##  Mean   :34.11   Mean   :261.1   Mean   :284.3   Mean   :292.0
##  3rd Qu.:41.27   3rd Qu.:303.2   3rd Qu.:294.2   3rd Qu.:300.0
##  Max.   :51.20   Max.   :421.0   Max.   :509.0   Max.   :518.0
##    MLPMALCMSY      MLPMALCMSE      MLPMALCMSR     MLPMALCMSL      MLPMALCMSU
##  Min.   : 2.40   Min.   :1.300   Min.   :0.20   Min.   : 0.00   Min.   : 5.00
##  1st Qu.:10.43   1st Qu.:2.750   1st Qu.:0.20   1st Qu.: 4.30   1st Qu.:16.05
##  Median :12.65   Median :3.400   Median :0.25   Median : 5.45   Median :21.10
##  Mean   :14.07   Mean   :3.260   Mean   :0.29   Mean   : 7.58   Mean   :20.62
##  3rd Qu.:17.52   3rd Qu.:4.075   3rd Qu.:0.30   3rd Qu.:10.78   3rd Qu.:24.32
```

```
##  Max.   :27.00   Max.   :4.300   Max.   :0.60   Max.   :18.40   Max.   :35.60
##    MLPMALCNMM      MLPMALCUNM      MLPMALCUEM             geometry
##  Min.   : 83.0   Min.   :187.0   Min.   :203.0   MULTIPOLYGON :10
##  1st Qu.:197.0   1st Qu.:249.2   1st Qu.:256.0   epsg:4326    : 0
##  Median :276.5   Median :271.5   Median :274.5   +proj=long...: 0
##  Mean   :261.8   Mean   :284.6   Mean   :292.0
##  3rd Qu.:305.5   3rd Qu.:294.2   3rd Qu.:300.0
##  Max.   :421.0   Max.   :509.0   Max.   :518.0
##  MLPMALCRDT_p       MLPMALCRDT_per
##  Length:10          Length:10
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

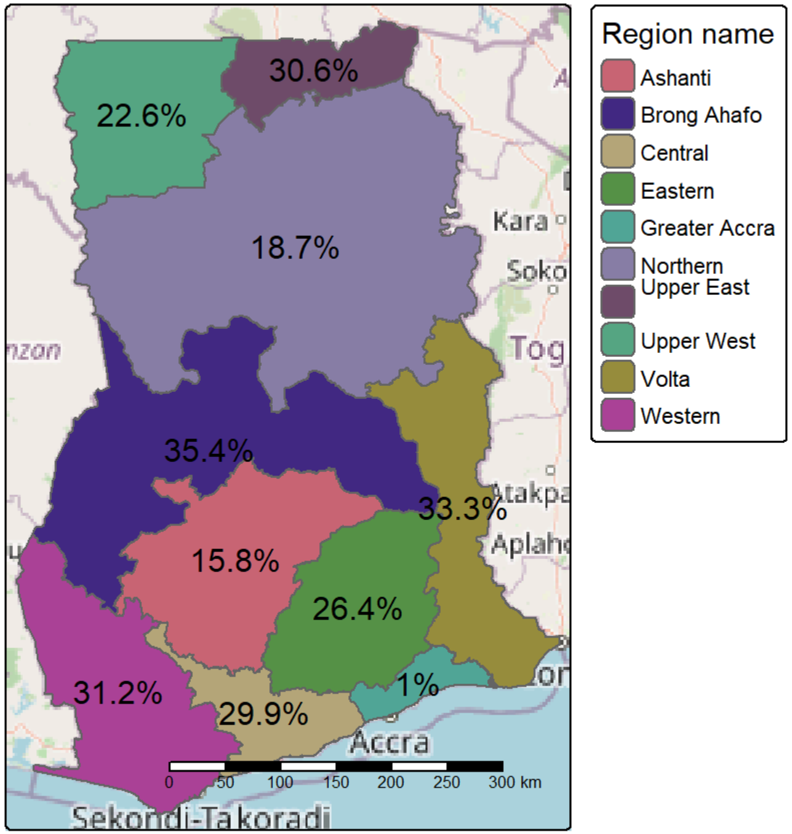
```
mal_prev6<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("REGNAME",title="Region name",
             style = "fixed",textNA="",
             legend.hist = F) +
             tm_text("MLPMALCRDT_per",size=1)+
             tm_layout(legend.outside = TRUE)+
             tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

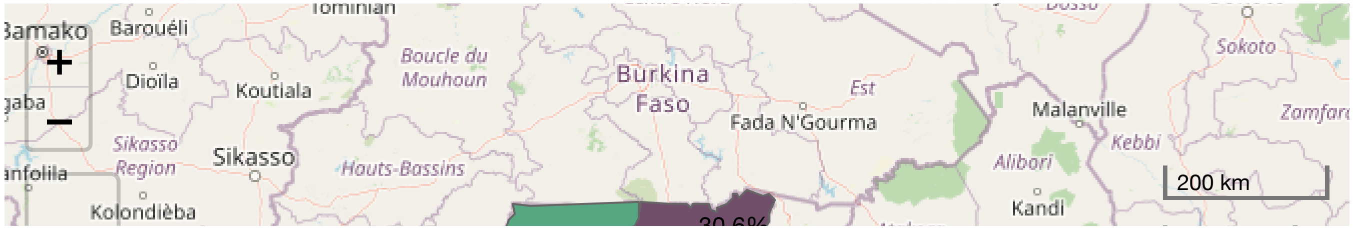
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples, specify a
'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

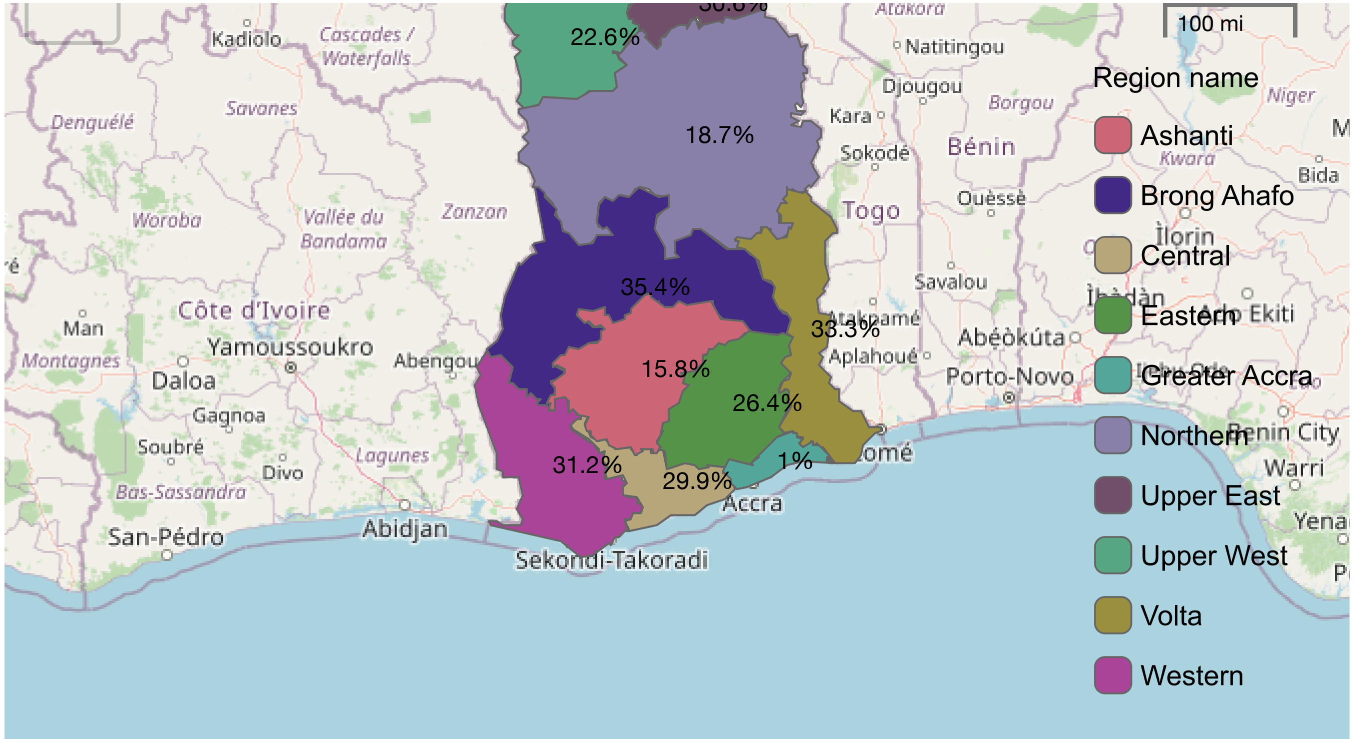
```
## [v3->v4] polygons(): migrate the argument(s) related to the legend of the visual variable 'fill', namely 'titl
e' to 'fill.legend = tm_legend(<HERE>)'
```

```
mal_prev6
```




```
mal_prev6_web <- tmap_leaflet(mal_prev6)
mal_prev6_web
```





Finally, we tricked the system in an attempt to put region names and the prevalence together

x11()# divide the result screen into 2 for the second map to aid

comparison

```
mal_prev7<-tm_basemap(leaflet::providers$OpenStreetMap) +
 tm_shape(mal_dat)+
 tm_polygons("REGNAME",title="Region name",legend.show = FALSE,
             style = "fixed",textNA="",
             legend.hist = F) +
             tm_text("MLPMALCRDT_per",size=1,just="top")+
             tm_shape(mal_dat)+
             tm_text("REGNAME",just="bottom", remove.overlap = F)+
             tm_layout(legend.outside = TRUE,legend.show=F)+
             tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

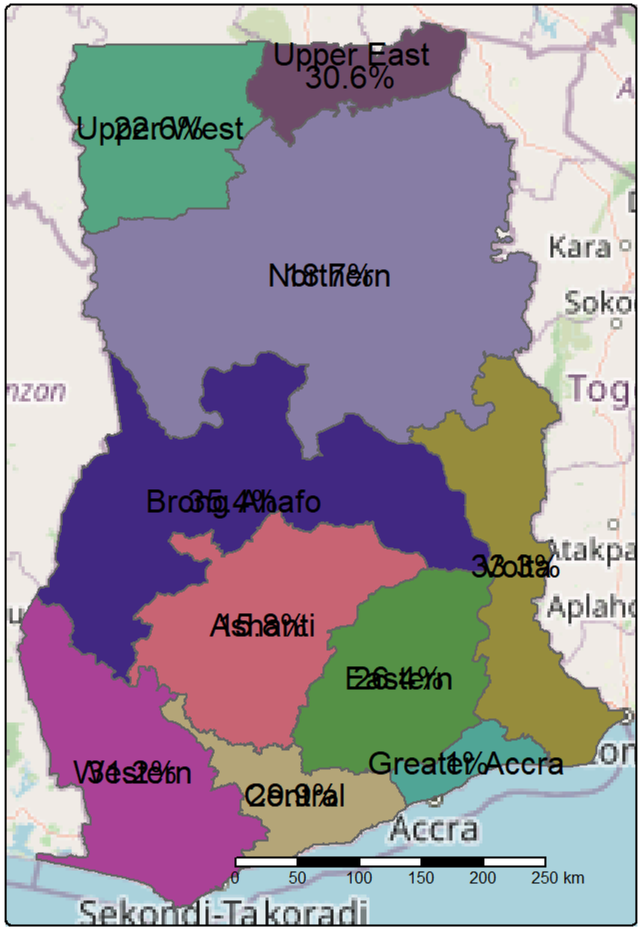
```
## [v3->v4] polygons(): instead of 'style = "fixed"', use 'fill.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'textNA' (rename to 'label.na') to 'tm_scale_intervals(<HERE>)'. For small multiples, specify a
'tm_scale_' for each multiple, and put them in a list: 'fill.scale = list(<scale1>, <scale2>, ...)'
```

```
## [v3->v4] polygons(): use 'fill.legend = tm_legend_hide()' instead of 'legend.show = FALSE
```

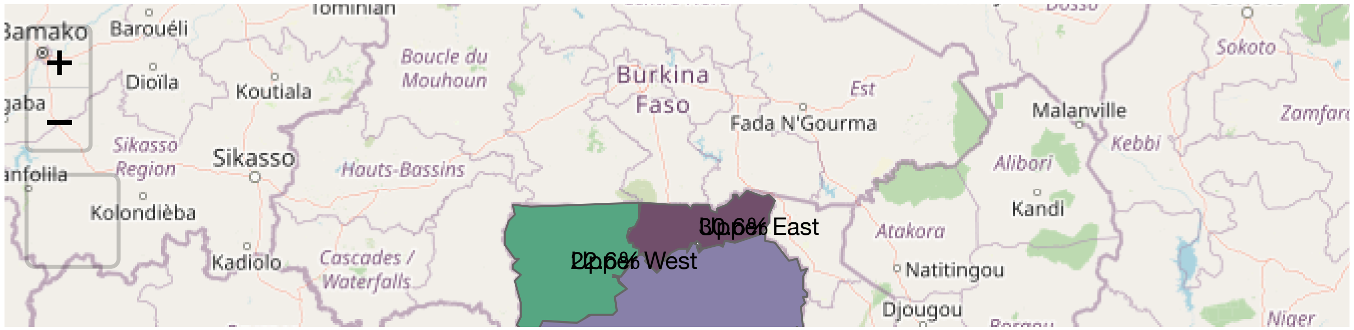
```
## [v3->v4] tm_text(): migrate the layer options 'just' to 'options = opt_tm_text(<HERE>)'
```

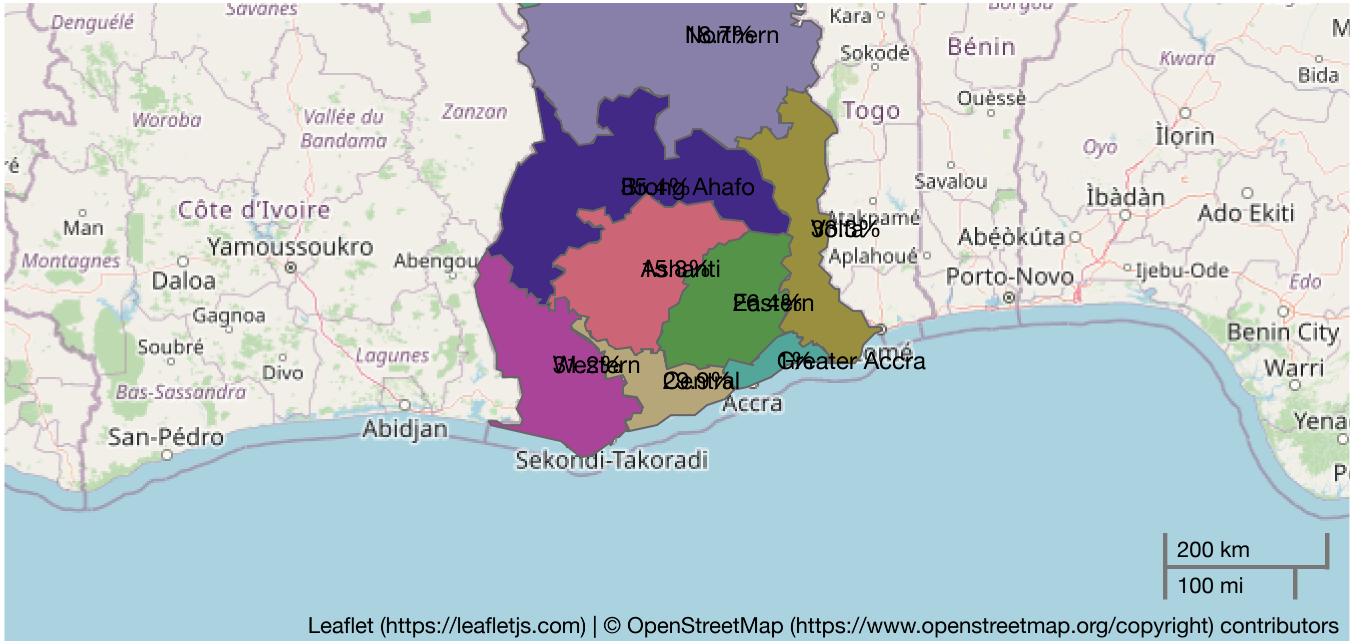
```
## [v3->v4] tm_text(): migrate the layer options 'just', 'remove.overlap' to 'options = opt_tm_text(<HERE>)'
```

```
mal_prev7
```




```
mal_prev7_web <- tmap_leaflet(mal_prev7)
mal_prev7_web
```





End of mapping lattice data based on SpatialPolygonsDataFrame object

Mapping with lattice data based on raster objects

Mapping percentage of children who tested positive for malaria

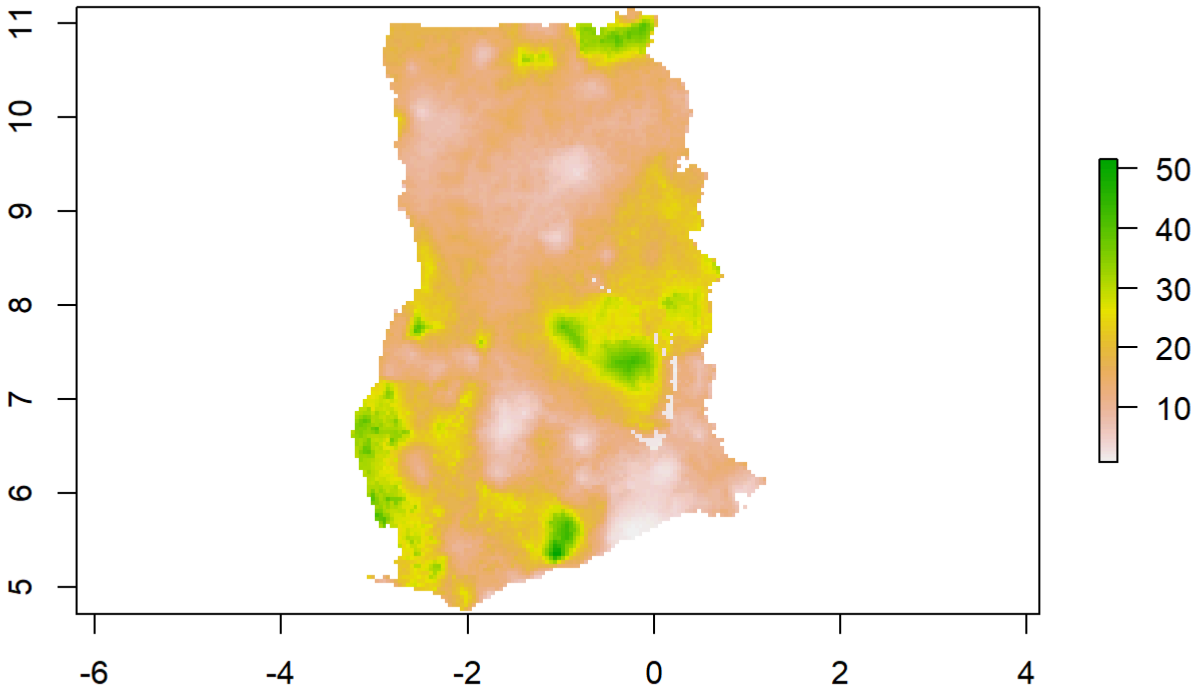
```
str_name<-'malaria_Ghana_5.tif' # First capture the .tiff file (i.e., the raster)

malaria=raster(str_name) # worked
class(malaria)
```

```
## [1] "RasterLayer"
## attr(,"package")
## [1] "raster"
```

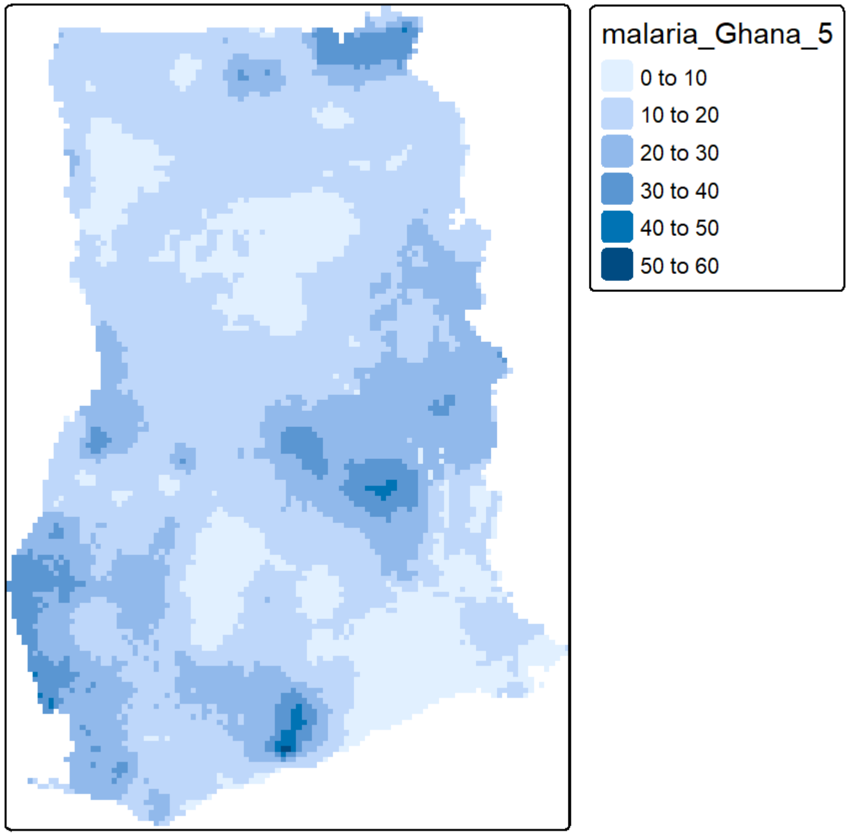
```
summary(malaria)
```

```
##          malaria_Ghana_5
## Min.           0.7434092
## 1st Qu.       11.1576664
## Median        15.0899386
## 3rd Qu.       20.8507910
## Max.          51.6266060
## NA's        5323.0000000
```

```
## default plot for raster file
plot(malaria) # modelled under-five malaria prevalence (aged 6-59)
```



```
## Using tmap package

#malaria<-st_as_sf(malaria)

tm_shape(malaria)+
tm_raster(col="malaria_Ghana_5")
```

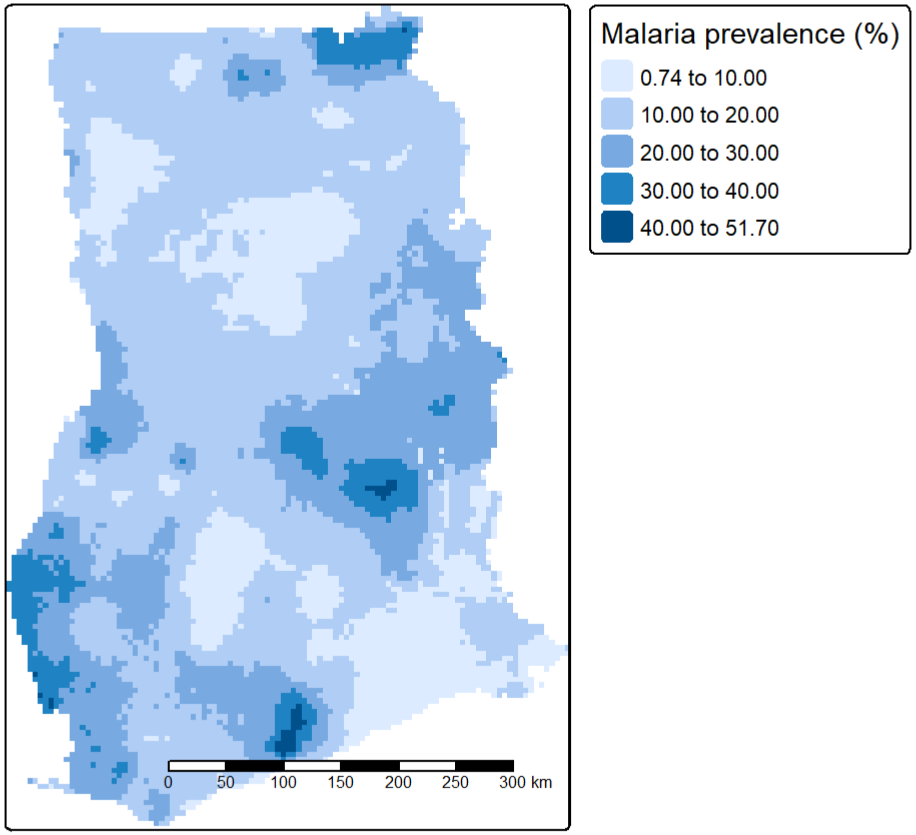


```
# Create a beautiful map for Ghana using tmap package for raster file

tm_shape(malaria)+
tm_raster(col="malaria_Ghana_5",title="Malaria prevalence (%)",
              style = "fixed",
              breaks = c(0.74,10,20,30,40,51.7),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

```
## [v3->v4] tm_raster(): instead of 'style = "fixed"', use 'col.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks' to 'tm_scale_intervals(<HERE>)'
```

```
## [v3->v4] tm_raster(): migrate the argument(s) related to the legend of the visual variable 'col', namely 'titl
e' to 'col.legend = tm_legend(<HERE>)'
```


Malaria prevalence (%)
0.74 to 10.00
10.00 to 20.00
20.00 to 30.00
30.00 to 40.00
40.00 to 51.70
0
50
100
150
200
250
300 km

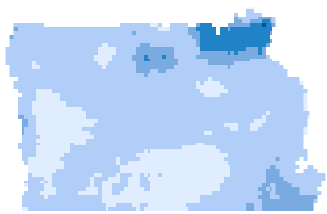
```
## Same but break the legend title and produce interactive web-based maps



## Same but assign object name

mal<-tm_shape(malaria)+
tm_raster(col="malaria_Ghana_5",title="Malaria \n prevalence (%)",
              style = "fixed",
              breaks = c(0.74,10,20,30,40,51.7),
              legend.hist = F) +
              tm_layout(legend.outside = TRUE)+
              tm_scalebar(position=c("right", "bottom"))
```

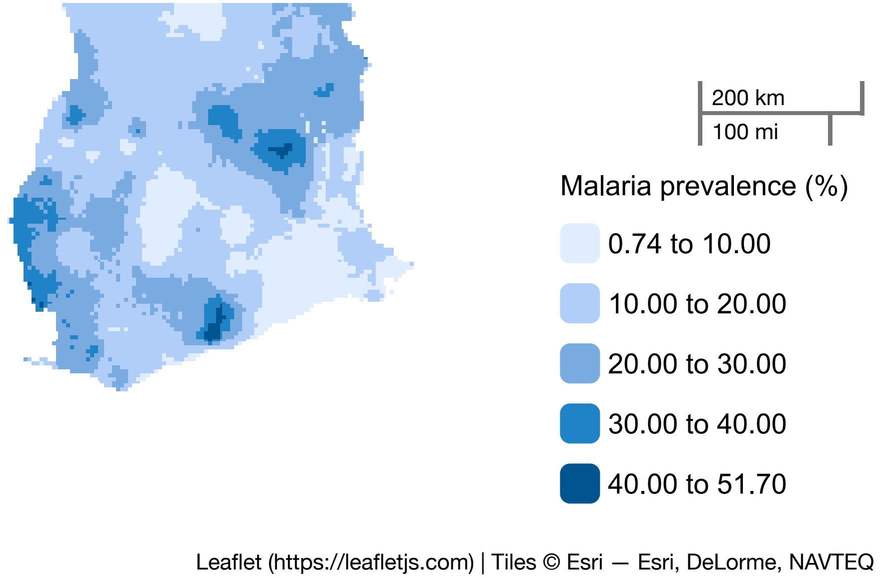
```
## -- tmap v3 code detected --
```

```
## [v3->v4] tm_raster(): instead of 'style = "fixed"', use 'col.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks' to 'tm_scale_intervals(<HERE>)'
```

```
## [v3->v4] tm_raster(): migrate the argument(s) related to the legend of the visual variable 'col', namely 'titl
e' to 'col.legend = tm_legend(<HERE>)'
```

```
mal_web <- tmap_leaflet(mal)
mal_web
```





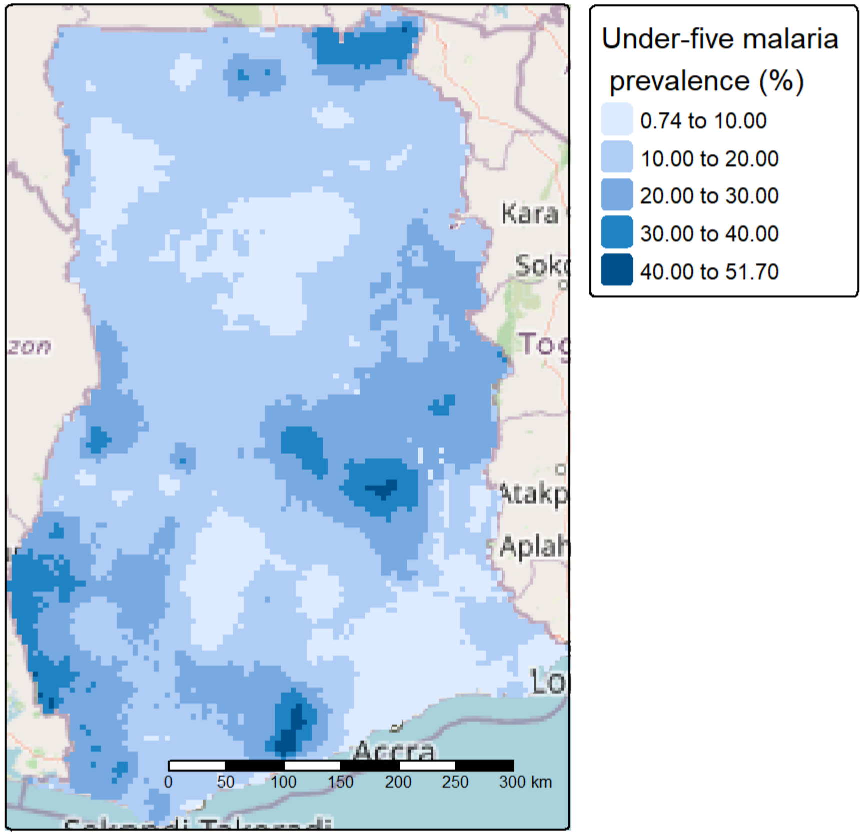

```
## Same but change tiles to OpenStreetMap

mal2<-tm_basemap(leaflet::providers$OpenStreetMap) +
      tm_shape(malaria)+
      tm_raster(col="malaria_Ghana_5",title="Under-five malaria \n prevalence (%)",
      style = "fixed",
      breaks = c(0.74,10,20,30,40,51.7),
      legend.hist = F) +
      tm_layout(legend.outside = TRUE)+
      tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

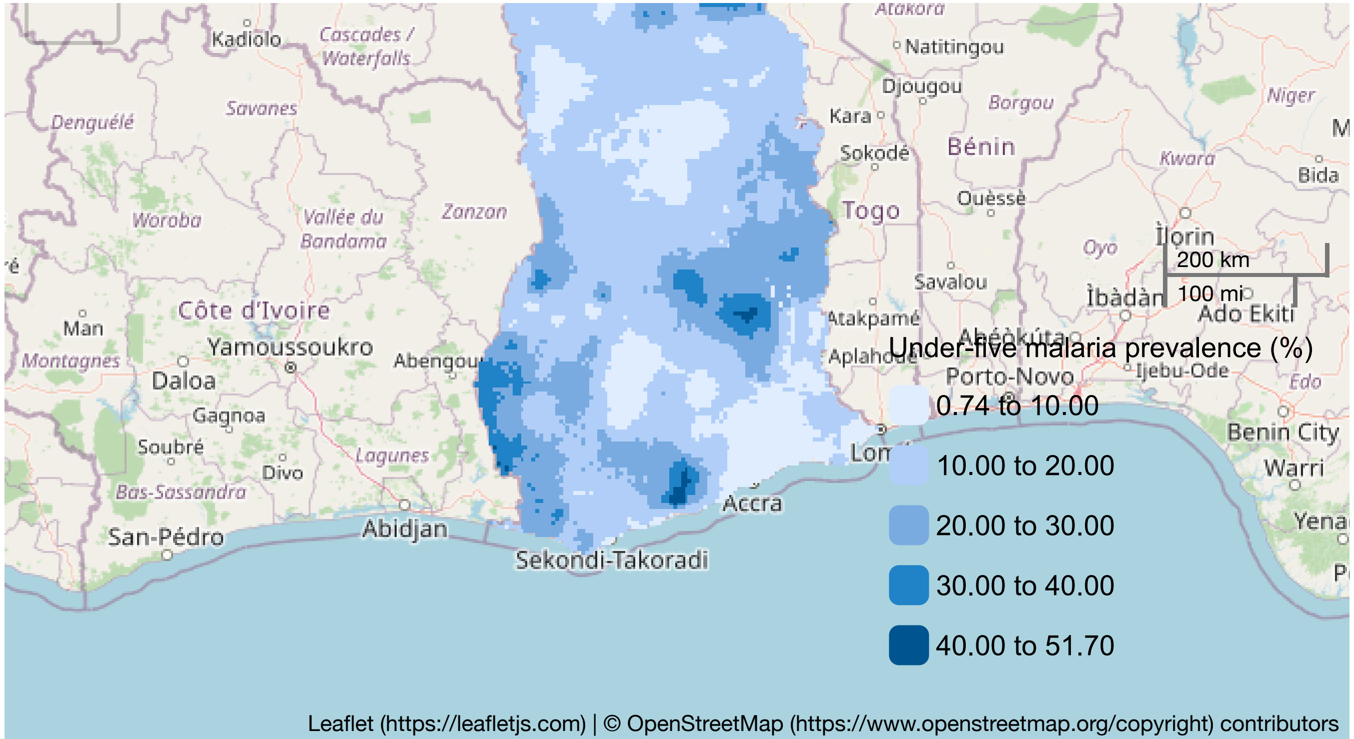
```
## [v3->v4] tm_raster(): instead of 'style = "fixed"', use 'col.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks' to 'tm_scale_intervals(<HERE>)'
```

```
## [v3->v4] tm_raster(): migrate the argument(s) related to the legend of the visual variable 'col', namely 'titl
e' to 'col.legend = tm_legend(<HERE>)'
```

```
mal2
```




```
mal2_web <- tmap_leaflet(mal2)
mal2_web
```



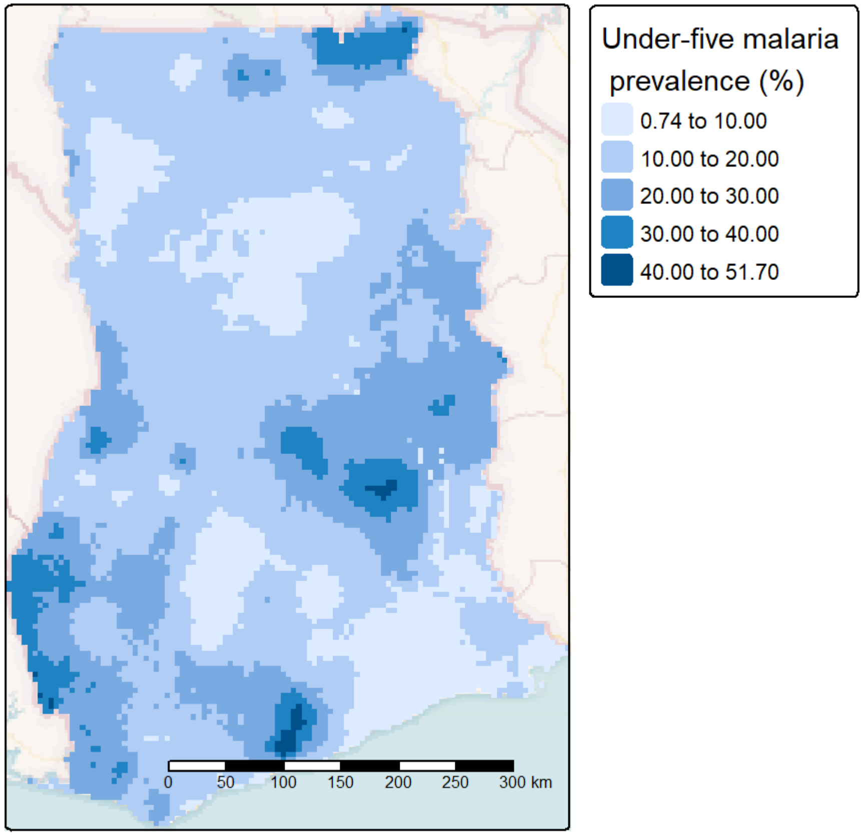

```
## Same but no basemap tile label

mal3<-tm_basemap(leaflet::providers$CartoDB.VoyagerNoLabels) +
      tm_shape(malaria)+
      tm_raster(col="malaria_Ghana_5",title="Under-five malaria \n prevalence (%)",
      style = "fixed",
      breaks = c(0.74,10,20,30,40,51.7),
      legend.hist = F) +
      tm_layout(legend.outside = TRUE)+
      tm_scalebar(position=c("right", "bottom"))
```

```
## -- tmap v3 code detected --
```

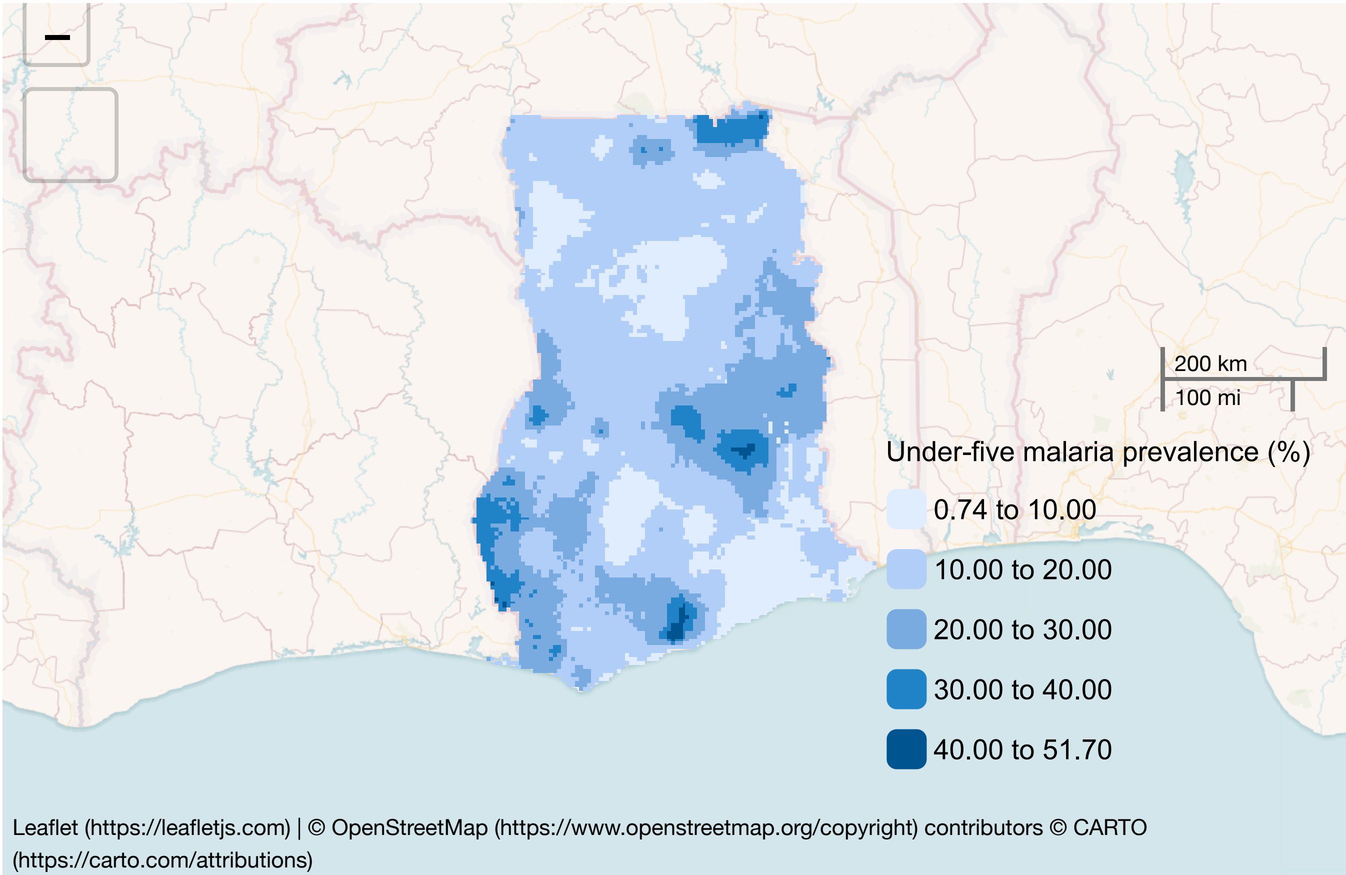
```
## [v3->v4] tm_raster(): instead of 'style = "fixed"', use 'col.scale = tm_scale_intervals()' and migrate the arg
ument(s) 'style', 'breaks' to 'tm_scale_intervals(<HERE>)'
```

```
## [v3->v4] tm_raster(): migrate the argument(s) related to the legend of the visual variable 'col', namely 'titl
e' to 'col.legend = tm_legend(<HERE>)'
```

```
mal3
```




```
mal3_web <- tmap_leaflet(mal3)
mal3_web
```



Geospatial mapping of malaria in R (Geostatistical data)

Geospatial mapping of malaria in R (Geostatistical data)

Reading CSV file into R

This data from from the 2019 Ghana Malaria Indicator Survey (MIS) at cluster

locations

```
mal_d<-read.csv("mal_d.csv")
names(mal_d)
```

```
## [1] "DHSCLUST" "mal_prev" "lon"      "lat"
```

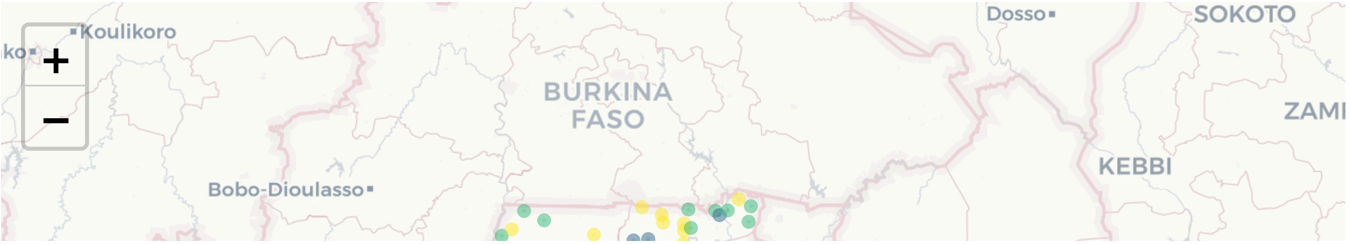
```
summary(mal_d)
```

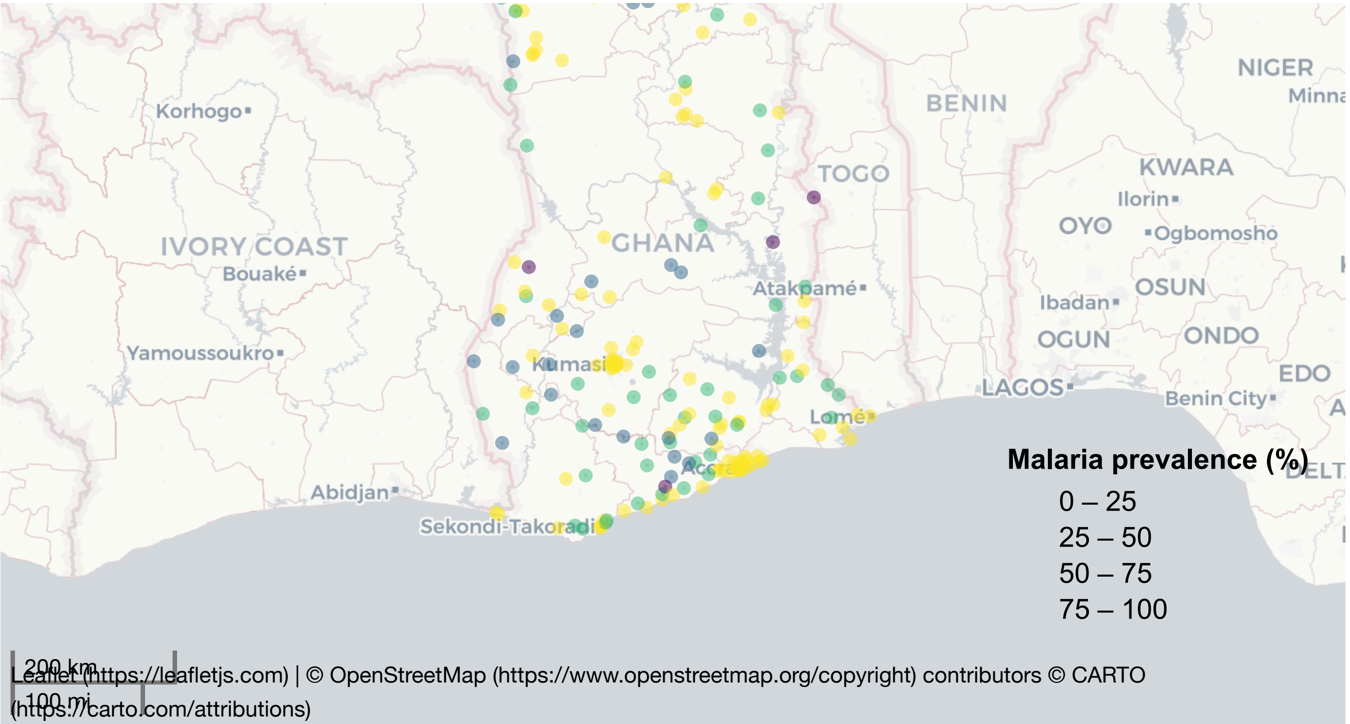
```
##     DHSCLUST         mal_prev           lon                lat
##  Min.   :  1.00   Min.   :0.0000   Min.   :-3.1526   Min.   : 4.894
##  1st Qu.: 50.75   1st Qu.:0.0000   1st Qu.:-1.7744   1st Qu.: 5.649
##  Median :101.50   Median :0.1791   Median :-0.9759   Median : 6.587
##  Mean   :101.04   Mean   :0.2223   Mean   :-1.0792   Mean   : 7.192
##  3rd Qu.:151.25   3rd Qu.:0.3750   3rd Qu.:-0.2612   3rd Qu.: 8.502
##  Max.   :200.00   Max.   :0.9000   Max.   : 1.1973   Max.   :11.066
```

```
## Convert the proportion to percentage
mal_d$mal_prev_per<-mal_d$mal_prev*100
summary(mal_d)
```

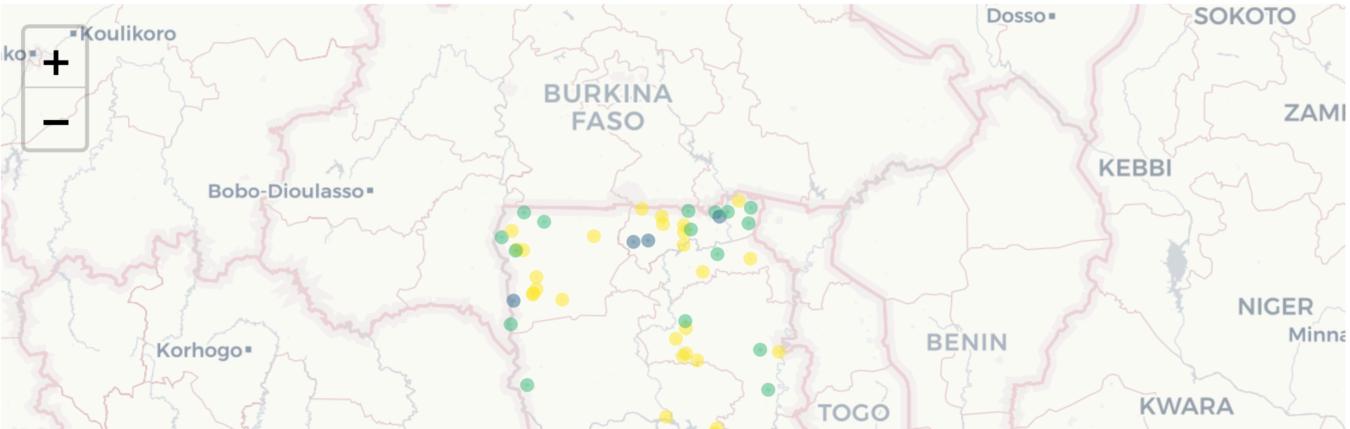
```
##     DHSCLUST         mal_prev           lon                lat        
##  Min.   :  1.00   Min.   :0.0000   Min.   :-3.1526   Min.   : 4.894  
##  1st Qu.: 50.75   1st Qu.:0.0000   1st Qu.:-1.7744   1st Qu.: 5.649  
##  Median :101.50   Median :0.1791   Median :-0.9759   Median : 6.587  
##  Mean   :101.04   Mean   :0.2223   Mean   :-1.0792   Mean   : 7.192  
##  3rd Qu.:151.25   3rd Qu.:0.3750   3rd Qu.:-0.2612   3rd Qu.: 8.502  
##  Max.   :200.00   Max.   :0.9000   Max.   : 1.1973   Max.   :11.066  
##   mal_prev_per  
##  Min.   : 0.00  
##  1st Qu.: 0.00  
##  Median :17.91  
##  Mean   :22.23  
##  3rd Qu.:37.50  
##  Max.   :90.00
```

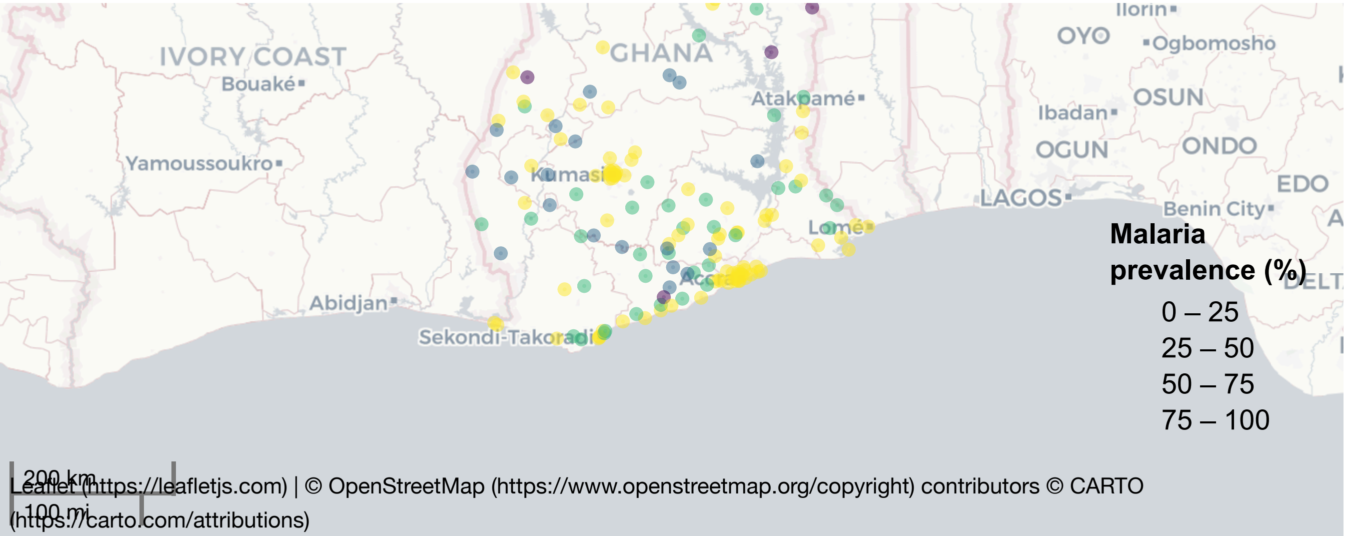
```
##  Mapping Using leaflet package (Our focus)

library(leaflet)
pal <- colorBin("viridis", bins = c(0, 25, 50, 75, 100),reverse=T)# note reverse

## pal <- colorBin("plasma", bins = c(0, 25, 50, 75, 100),reverse=T)
## pal <- colorBin("inferno", bins = c(0, 25, 50, 75, 100),reverse=T)



## Mapping R file using leaflet package
leaflet(mal_d) %>%  addProviderTiles(providers$CartoDB.Positron) %>%
  addCircles(lng = ~lon, lat = ~lat, color = ~pal(mal_d$mal_prev_per)) %>%
   addLegend("bottomright", pal = pal, values = ~mal_prev_per,
  title = "Malaria prevalence (%)") %>%
  addScaleBar(position = c("bottomleft"))
```






```
leaflet(mal_d) %>%  addProviderTiles(providers$CartoDB.Positron) %>%
  addCircles(lng = ~lon, lat = ~lat, color = ~pal(mal_d$mal_prev_per)) %>%
   addLegend("bottomright", pal = pal, values = ~mal_prev_per,
  title = "Malaria <br> prevalence (%)") %>%
  addScaleBar(position = c("bottomleft"))
```

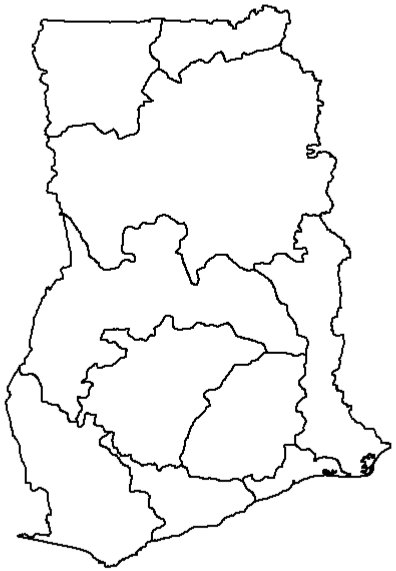





```
#  Final

## Sunsetting polygons for targetted analysis

load("reg_data.RData")
names(reg_data)
```

```
##  [1] "NAME_1"    "ID_0"      "COUNTRY"   "ID_1.x"    "VARNAME_1" "NL_NAME_1"
##  [7] "TYPE_1"    "ENGTYPE_1" "CC_1"      "HASC_1"    "ISO_1"     "ID_1.y"
## [13] "malaria_p"
```

```
#reg_data@data
plot(reg_data)
```



```
class(reg_data)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

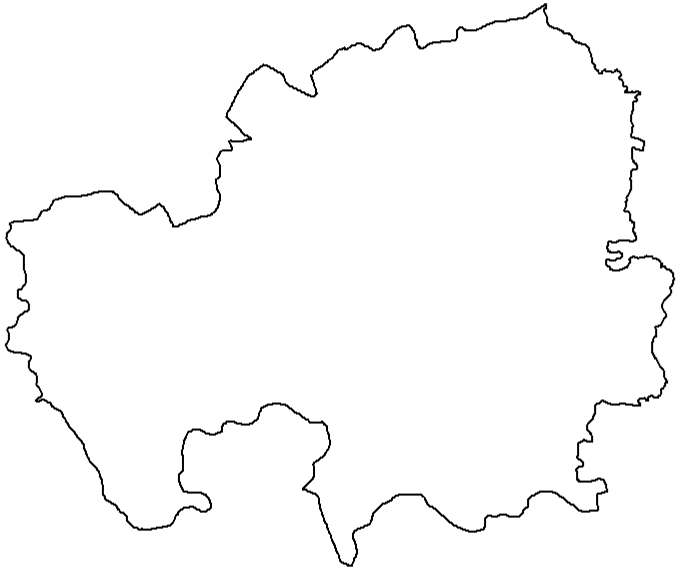
```
# Let us extract only the Northern region polygon
#reg_data@data$NAME_1

Northern <- subset(reg_data, NAME_1 == "Northern")
class(Northern)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
summary(Northern)
```

```
## Object of class SpatialPolygonsDataFrame
## Coordinates:
##         min       max
## x -2.779665  0.563871
## y  7.965667 10.742073
## Is projected: FALSE
## proj4string : [+proj=longlat +datum=WGS84 +no_defs]
## Data attributes:
##     NAME_1              ID_0              COUNTRY             ID_1.x         
##  Length:1           Length:1           Length:1           Length:1          
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                            
##                                                                            
##                                                                            
##   VARNAME_1          NL_NAME_1            TYPE_1           ENGTYPE_1        
##  Length:1           Length:1           Length:1           Length:1          
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                            
##                                                                            
##                                                                            
##      CC_1              HASC_1             ISO_1              ID_1.y         
##  Length:1           Length:1           Length:1           Length:1          
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                            
##                                                                            
##                                                                            
##    malaria_p   
##  Min.   :18.7  
##  1st Qu.:18.7  
##  Median :18.7  
##  Mean   :18.7  
##  3rd Qu.:18.7  
##  Max.   :18.7
```

```
plot(Northern) # Plotting the Northern region polygon
```



```
dist<-readOGR("gadm40_GHA_shp","gadm40_GHA_2")
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

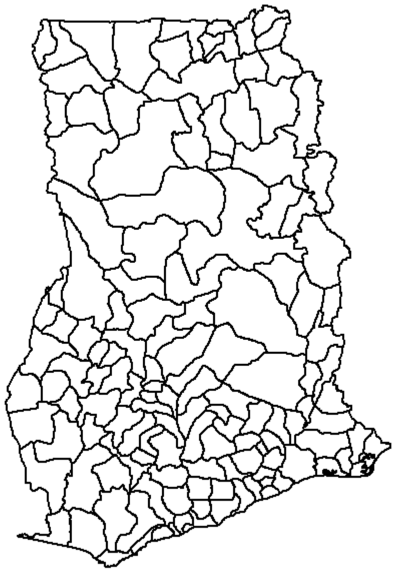
```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\jmkaheto\Documents\Dropbox\Consult\Dr.Jaline Gerardin Northwestern Univ USA\AMMNet_Workshop_
Ghana_Aheto\gadm40_GHA_shp", layer: "gadm40_GHA_2"
## with 137 features
## It has 12 fields
```

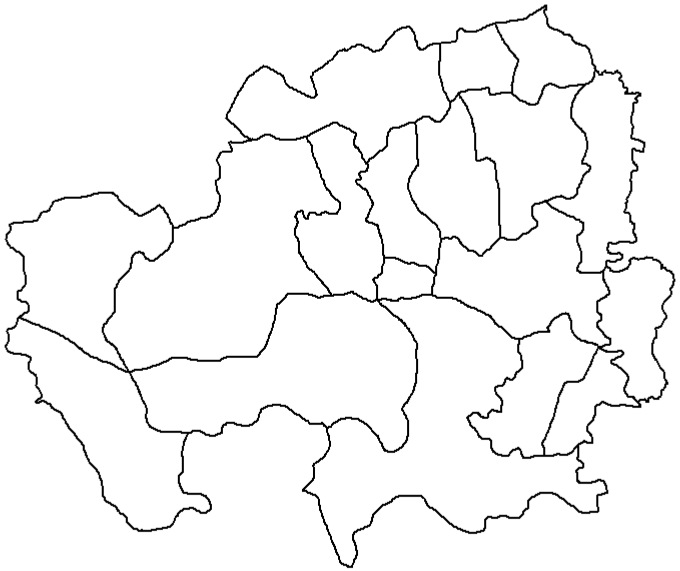
```
plot(dist)
```



```
names(dist)
```

```
##  [1] "ID_0"      "COUNTRY"   "NAME_1"    "NL_NAME_1" "ID_2"      "NAME_2"
##  [7] "VARNAME_2" "NL_NAME_2" "TYPE_2"    "ENGTYPE_2" "CC_2"      "HASC_2"
```

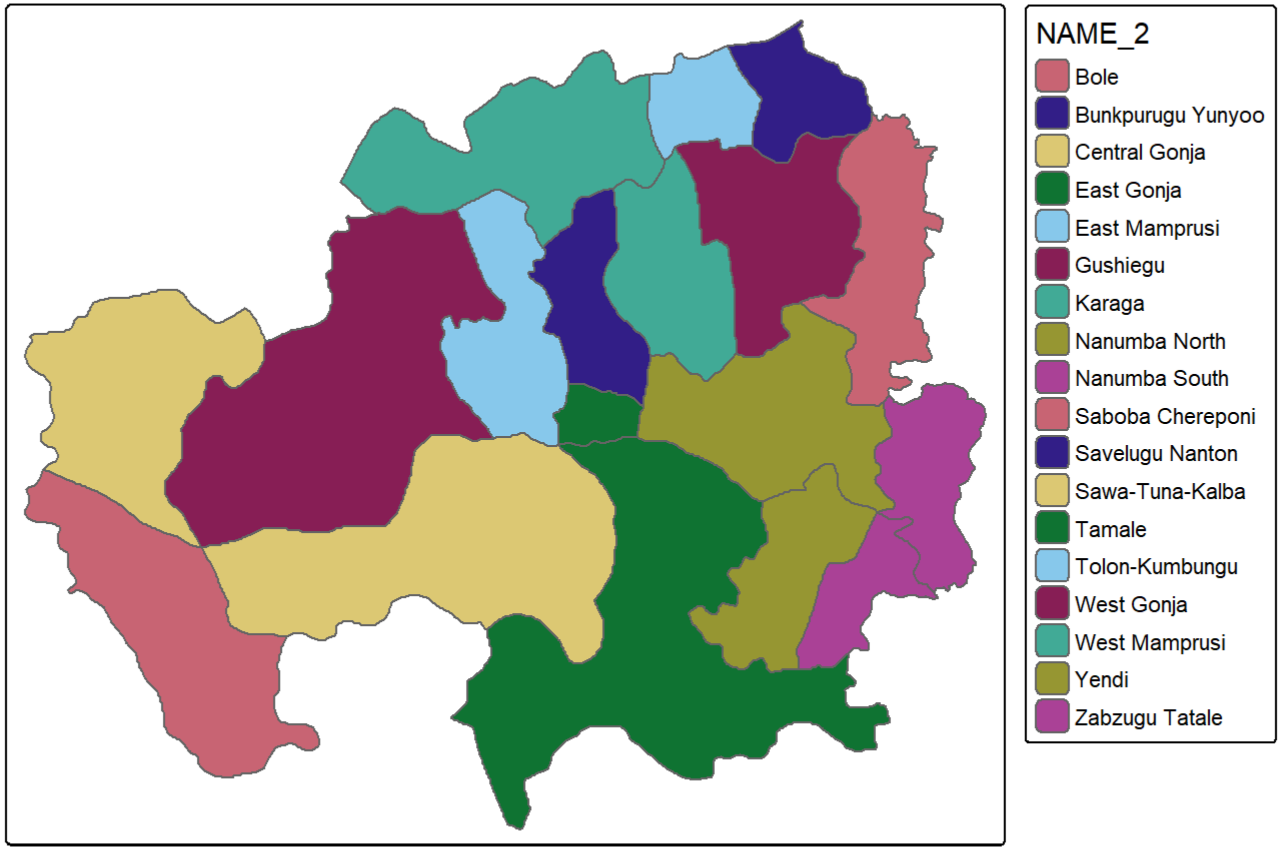
```
Northern2 <- subset(dist, NAME_1 == "Northern")
plot(Northern2)
```



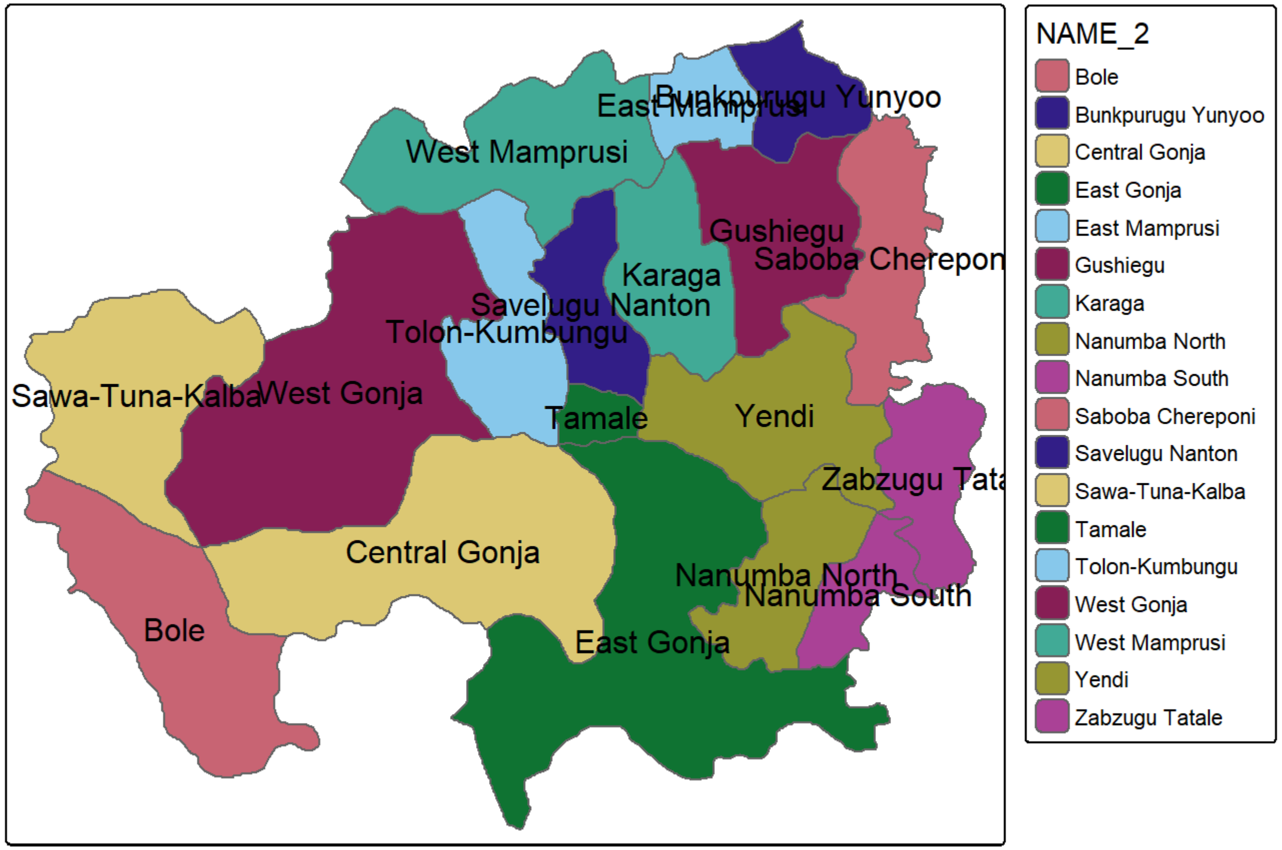
```
names(Northern2)
```

```
##  [1] "ID_0"      "COUNTRY"   "NAME_1"    "NL_NAME_1" "ID_2"      "NAME_2"
##  [7] "VARNAME_2" "NL_NAME_2" "TYPE_2"    "ENGTYPE_2" "CC_2"      "HASC_2"
```

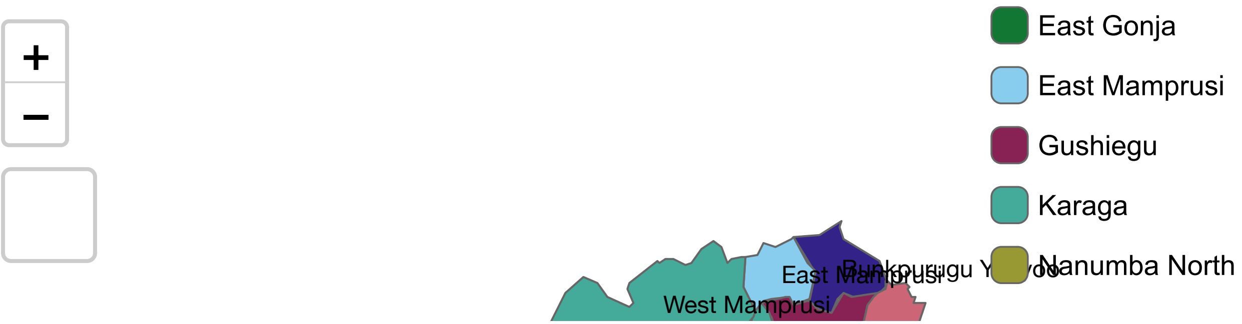
```
Northern2<-st_as_sf(Northern2)

tm_shape(Northern2)+
tm_polygons("NAME_2")
```

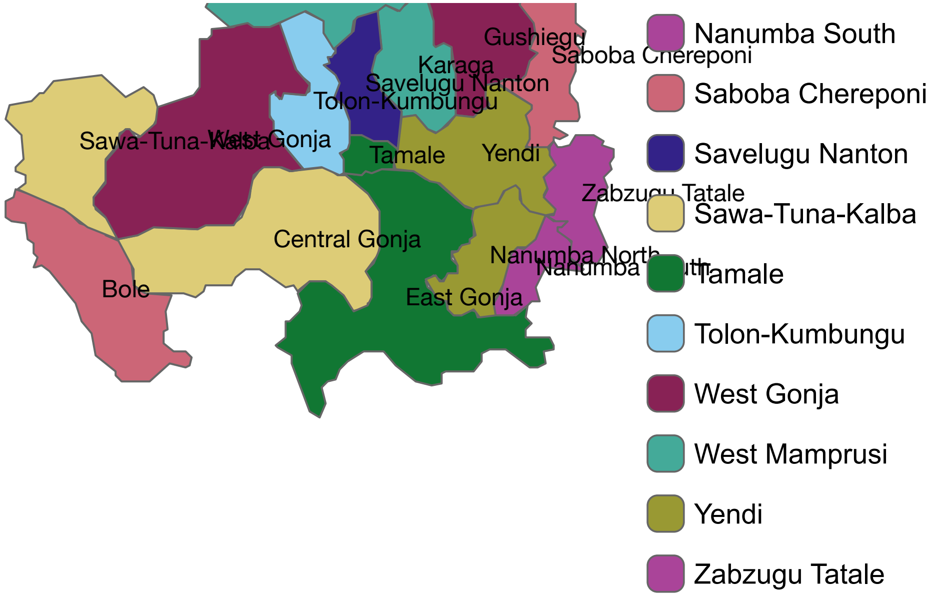



```
wmap<-tm_shape(Northern2) +
    tm_polygons("NAME_2")+
    tm_text("NAME_2")

wmap
```




```
map_web <- tmap_leaflet(wmap)
map_web
```



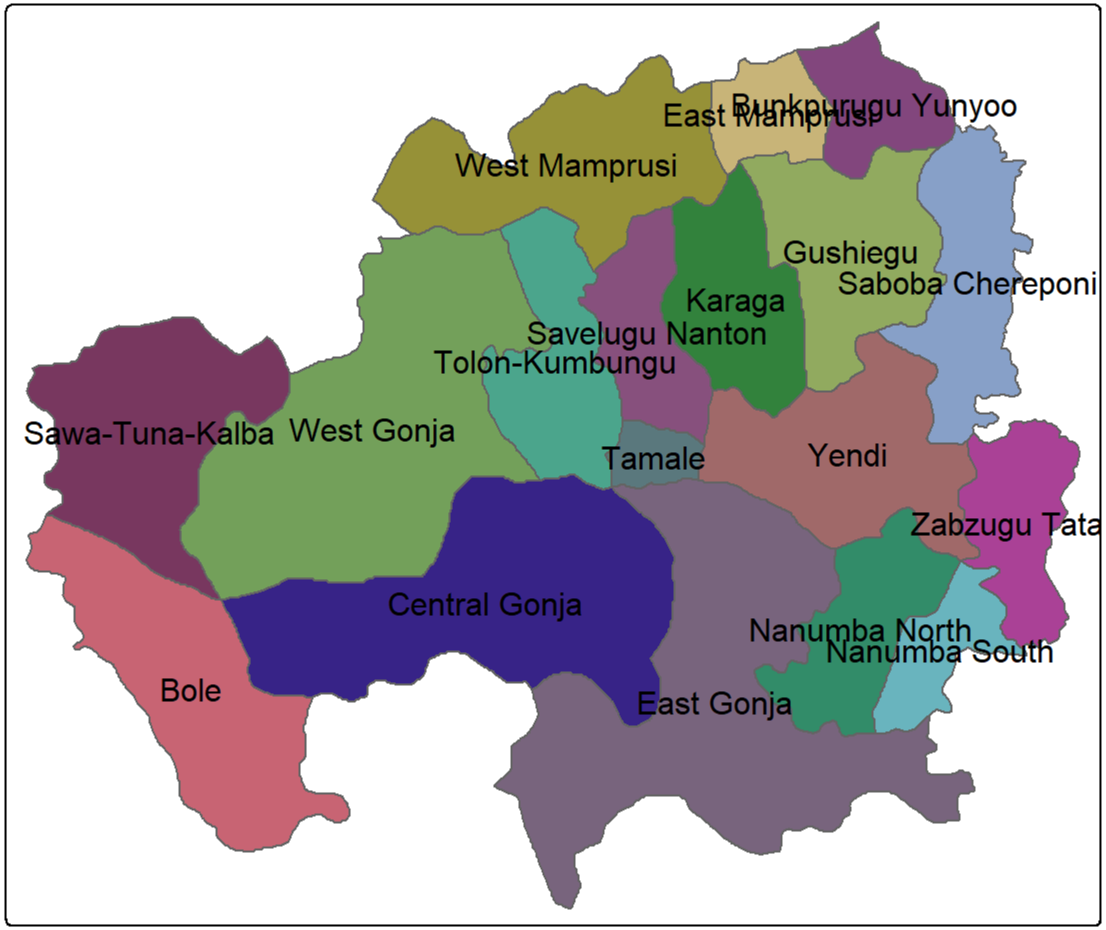



```
# Same as above BUT no legend!
wmap2<-tm_shape(Northern2) +
    tm_polygons("NAME_2",legend.show = FALSE)+
    tm_text("NAME_2")
```

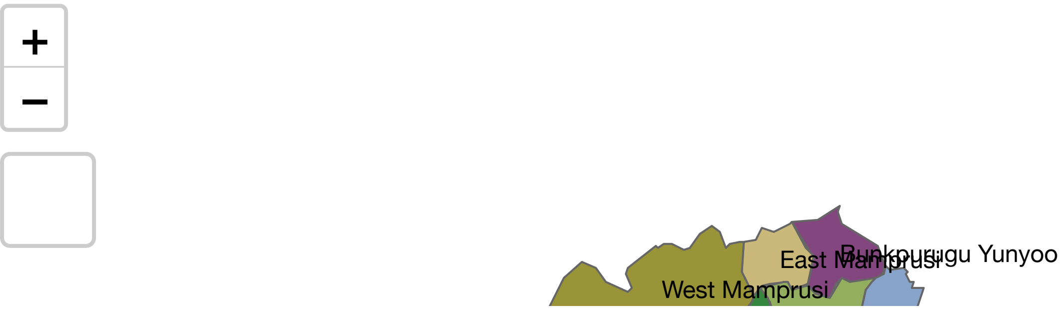
```
## -- tmap v3 code detected --
```

```
## [v3->v4] polygons(): use 'fill.legend = tm_legend_hide()' instead of 'legend.show = FALSE
```

```
wmap2
```




```
map_web2 <- tmap_leaflet(wmap2)
map_web2
```

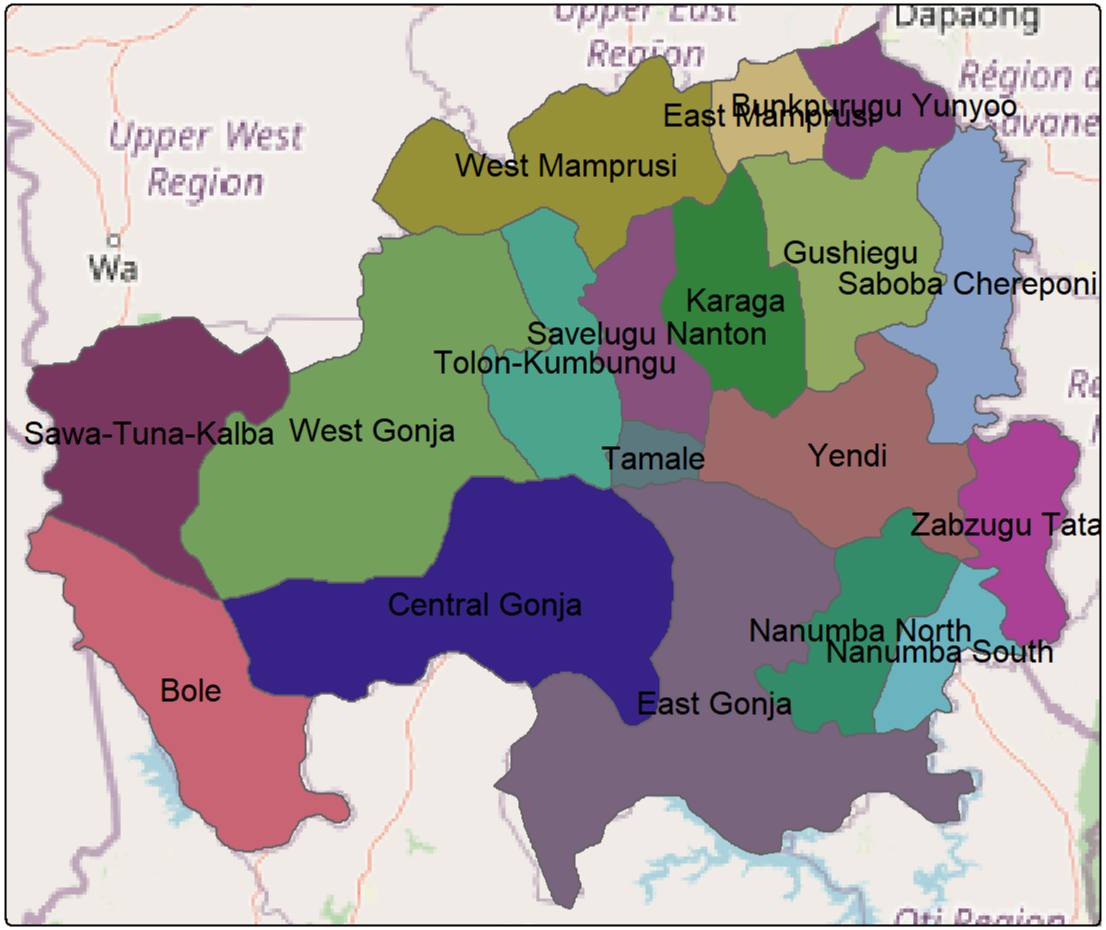


Leaflet (https://leafletjs.com) | Tiles © Esri — Esri, DeLorme, NAVTEQ

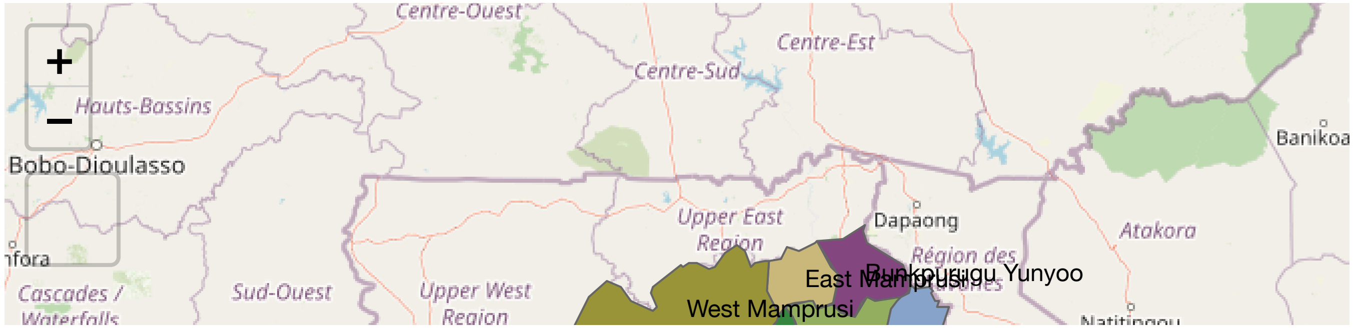
```
# Same as above BUT change the basemap tile to Open Street Map!
wmap3<-tm_basemap(leaflet::providers$OpenStreetMap) +
    tm_shape(Northern2) +
    tm_polygons("NAME_2",legend.show = FALSE,title = "Region name")+
    tm_text("NAME_2")
```

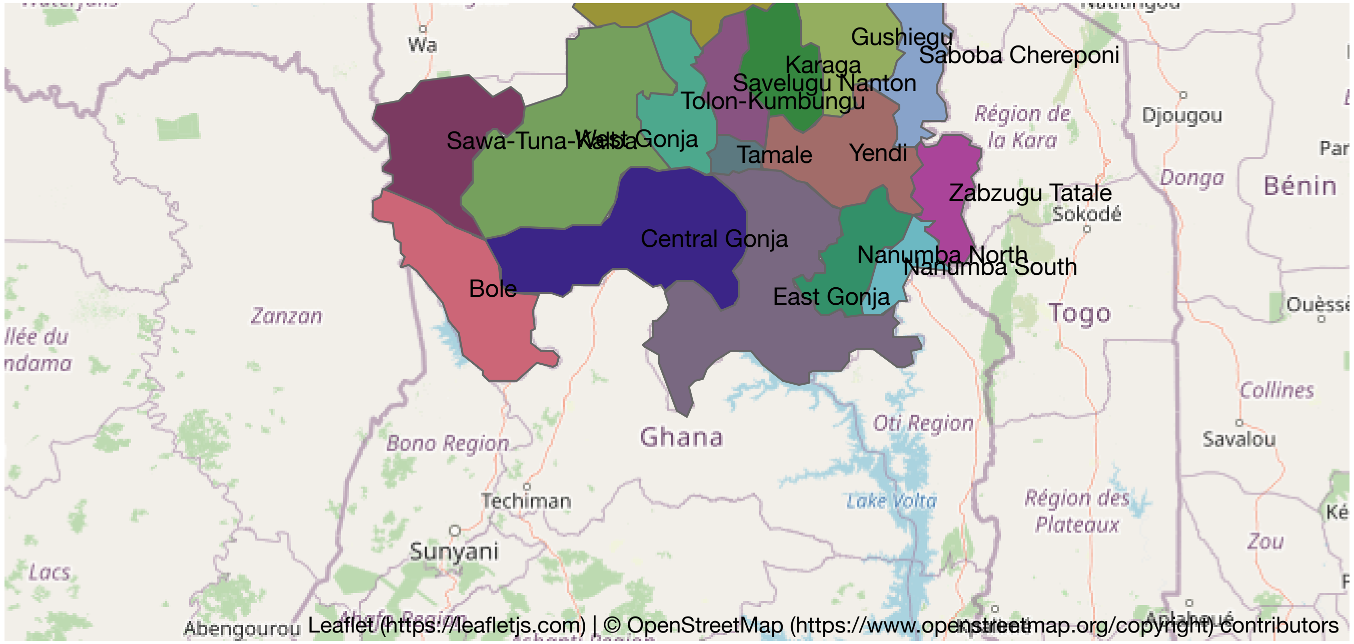
```
## -- tmap v3 code detected --
## [v3->v4] polygons(): use 'fill.legend = tm_legend_hide()' instead of 'legend.show = FALSE
```

```
wmap3
```




```
map_web3 <- tmap_leaflet(wmap3)
map_web3
```





Mini project work: Map the malaria prevalence (%) according to Microscopy test across the 10 regions of Ghana using the 2019 Ghana Malaria Indicator Survey report

Steps:

1. Download the regional shapefile for Ghana (for the 10 regions)
2. Obtain the malaria prevalence (%) data for each region
3. Add the data obtained in step 2 above to the shapefile obtained in step 1.
4. Check the data in step 3 for validation and accuracy
5. Produce a regional map for the final data in step 4.
6. Perform the same mapping in step 5 but insert the actual observed figures with the percentage sign inside the map.
7. In 6 lines of A4 paper, interpret and discuss the map produced for targeted policy decision making.

END OF GEOSPATIAL MAPPING

Geostatistical modelling and mapping

Load packages

```
library(rgdal)
library(sp)
library(INLA)
```

```
## Loading required package: Matrix
```

```
## This is INLA_24.04.25-1 built 2024-04-25 17:05:50 UTC.
##  - See www.r-inla.org/contact-us for how to get help.
##  - List available models/likelihoods/etc with inla.list.models()
##  - Use inla.doc(<NAME>) to access documentation
```

```
library(leaflet)
library(leafem)



# Geostatistical modelling and web-based mapping of malaria risk in Ghana:

load("mal.imp.RData")
summary(mal.imp)
```

```
##      hhid               hv001          hv002            hv005        
##  Length:2867        Min.   :  1.0   Min.   :  0.00   Min.   : 149717  
##  Class :character   1st Qu.: 64.0   1st Qu.: 38.00   1st Qu.: 536156  
##  Mode  :character   Median :127.0   Median : 77.00   Median : 790561  
##                     Mean   :114.7   Mean   : 93.95   Mean   : 918146  
##                     3rd Qu.:167.0   3rd Qu.:133.00   3rd Qu.:1168775  
##                     Max.   :200.0   Max.   :440.00   Max.   :5399987  
##      hv024          hv025          hv206            hv208       
##  Min.   : 1.00   Min.   :1.00   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.: 3.00   1st Qu.:1.00   1st Qu.:0.0000   1st Qu.:0.0000  
##  Median : 6.00   Median :2.00   Median :1.0000   Median :1.0000  
##  Mean   : 5.88   Mean   :1.63   Mean   :0.7213   Mean   :0.5609  
##  3rd Qu.: 8.00   3rd Qu.:2.00   3rd Qu.:1.0000   3rd Qu.:1.0000  
##  Max.   :10.00   Max.   :2.00   Max.   :1.0000   Max.   :1.0000  
##      hv219          hv227           hv270           hv104      
##  Min.   :1.000   Min.   :0.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000  
##  Median :1.000   Median :1.000   Median :2.000   Median :1.000  
##  Mean   :1.285   Mean   :0.864   Mean   :2.477   Mean   :1.489  
##  3rd Qu.:2.000   3rd Qu.:1.000   3rd Qu.:4.000   3rd Qu.:2.000  
##  Max.   :2.000   Max.   :1.000   Max.   :5.000   Max.   :2.000  
##       hc57            pos             hv228n          hv253n      
##  Min.   :1.000   Min.   :0.0000   Min.   :0.000   Min.   :0.0000  
##  1st Qu.:2.000   1st Qu.:0.0000   1st Qu.:1.000   1st Qu.:0.0000  
##  Median :3.000   Median :0.0000   Median :1.000   Median :0.0000  
##  Mean   :3.089   Mean   :0.2504   Mean   :1.164   Mean   :0.1758  
##  3rd Qu.:4.000   3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:0.0000  
##  Max.   :4.000   Max.   :1.0000   Max.   :3.000   Max.   :1.0000  
##      hml2n            hv014n           hml10n         child_age     
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
##  Median :1.0000   Median :1.0000   Median :1.0000   Median :1.0000  
##  Mean   :0.7021   Mean   :0.6697   Mean   :0.5773   Mean   :0.8832  
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000  
##  Max.   :2.0000   Max.   :2.0000   Max.   :1.0000   Max.   :2.0000  
##      hv009n      
##  Min.   :0.0000  
##  1st Qu.:0.0000  
```

```
##  Median :1.0000
##  Mean   :0.6822
##  3rd Qu.:1.0000
##  Max.   :2.0000
```

```
ftable(mal.imp$pos)
```

```
##     0    1
##
##  2149  718
```

```
names(mal.imp)
```

```
##  [1] "hhid"      "hv001"     "hv002"     "hv005"     "hv024"     "hv025"
##  [7] "hv206"     "hv208"     "hv219"     "hv227"     "hv270"     "hv104"
## [13] "hc57"      "pos"       "hv228n"    "hv253n"    "hml2n"     "hv014n"
## [19] "hml10n"    "child_age" "hv009n"
```

```
head(mal.imp)
```

```
save(mal.imp,file="mal.imp.RData")


# Read in shapefile

gh<-readOGR("GHGE81FL","GHGE81FL")
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## Warning: OGR support is provided by the sf and terra packages among others
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\jmkaheto\Documents\Dropbox\Consult\Dr.Jaline Gerardin Northwestern Univ USA\AMMNet_Workshop_
Ghana_Aheto\GHGE81FL", layer: "GHGE81FL"
## with 200 features
## It has 20 fields
```

```
plot(gh,axes=T)
summary(gh)
```

```
## Object of class SpatialPointsDataFrame
## Coordinates:
##                     min       max
## coords.x1 -3.152551e+00  1.197287
## coords.x2  5.684342e-14 11.065508
## coords.x3  0.000000e+00  0.000000
## Is projected: FALSE
## proj4string : [+proj=longlat +datum=WGS84 +no_defs]
## Number of points: 200
## Data attributes:
##     DHSID              DHSCC              DHSYEAR       DHSCLUST
##  Length:200         Length:200         Min.   :2019   Min.   :  1.00
##  Class :character   Class :character   1st Qu.:2019   1st Qu.: 50.75
##  Mode  :character   Mode  :character   Median :2019   Median :100.50
##                                        Mean   :2019   Mean   :100.50
##                                        3rd Qu.:2019   3rd Qu.:150.25
##                                        Max.   :2019   Max.   :200.00
##     CCFIPS            ADM1FIPS          ADM1FIPSNA         ADM1SALBNA
##  Length:200         Length:200         Length:200         Length:200
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   ADM1SALBCO           ADM1DHS         ADM1NAME           DHSREGCO
##  Length:200         Min.   : 1.000   Length:200         Min.   : 1.000
##  Class :character   1st Qu.: 3.000   Class :character   1st Qu.: 3.000
##  Mode  :character   Median : 5.000   Mode  :character   Median : 5.000
##                     Mean   : 5.235                      Mean   : 5.235
##                     3rd Qu.: 7.000                      3rd Qu.: 7.000
##                     Max.   :10.000                      Max.   :10.000
##    DHSREGNA            SOURCE           URBAN_RURA           LATNUM
##  Length:200         Length:200         Length:200         Min.   : 0.000
##  Class :character   Class :character   Class :character   1st Qu.: 5.618
##  Mode  :character   Mode  :character   Mode  :character   Median : 6.473
##                                                           Mean   : 6.905
##                                                           3rd Qu.: 8.110
##                                                           Max.   :11.066
```

```
##     LONGNUM           ALT_GPS         ALT_DEM          DATUM
##  Min.   :-3.1526   Min.   :-40.5   Min.   :   3.0   Length:200
##  1st Qu.:-1.7543   1st Qu.:108.0   1st Qu.:  58.5   Class :character
##  Median :-0.8832   Median :143.5   Median : 163.0   Mode  :character
##  Mean   :-1.0360   Mean   :161.8   Mean   : 208.1
##  3rd Qu.:-0.2296   3rd Qu.:228.8   3rd Qu.: 244.0
##  Max.   : 1.1973   Max.   :559.8   Max.   :9999.0
```

```
save(gh,file="gh.RData")
load("gh.RData")
dim(gh)
```

```
## [1] 200  20
```

```
summary(gh)
```

```
## Object of class SpatialPointsDataFrame
## Coordinates:
##                      min       max
## coords.x1 -3.152551e+00  1.197287
## coords.x2  5.684342e-14 11.065508
## coords.x3  0.000000e+00  0.000000
## Is projected: FALSE
## proj4string : [+proj=longlat +datum=WGS84 +no_defs]
## Number of points: 200
## Data attributes:
##     DHSID              DHSCC              DHSYEAR       DHSCLUST
##  Length:200         Length:200         Min.   :2019   Min.   :  1.00
##  Class :character   Class :character   1st Qu.:2019   1st Qu.: 50.75
##  Mode  :character   Mode  :character   Median :2019   Median :100.50
##                                        Mean   :2019   Mean   :100.50
##                                        3rd Qu.:2019   3rd Qu.:150.25
##                                        Max.   :2019   Max.   :200.00
##     CCFIPS            ADM1FIPS          ADM1FIPSNA         ADM1SALBNA
##  Length:200         Length:200         Length:200         Length:200
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   ADM1SALBCO           ADM1DHS         ADM1NAME           DHSREGCO
##  Length:200         Min.   : 1.000   Length:200         Min.   : 1.000
##  Class :character   1st Qu.: 3.000   Class :character   1st Qu.: 3.000
##  Mode  :character   Median : 5.000   Mode  :character   Median : 5.000
##                     Mean   : 5.235                      Mean   : 5.235
##                     3rd Qu.: 7.000                      3rd Qu.: 7.000
##                     Max.   :10.000                      Max.   :10.000
##    DHSREGNA            SOURCE           URBAN_RURA            LATNUM
##  Length:200         Length:200         Length:200         Min.   : 0.000
##  Class :character   Class :character   Class :character   1st Qu.: 5.618
##  Mode  :character   Mode  :character   Mode  :character   Median : 6.473
##                                                           Mean   : 6.905
##                                                           3rd Qu.: 8.110
##                                                           Max.   :11.066
```

```
##     LONGNUM            ALT_GPS         ALT_DEM          DATUM
##  Min.   :-3.1526   Min.   :-40.5   Min.   :   3.0   Length:200
##  1st Qu.:-1.7543   1st Qu.:108.0   1st Qu.:  58.5   Class :character
##  Median :-0.8832   Median :143.5   Median : 163.0   Mode  :character
##  Mean   :-1.0360   Mean   :161.8   Mean   : 208.1
##  3rd Qu.:-0.2296   3rd Qu.:228.8   3rd Qu.: 244.0
##  Max.   : 1.1973   Max.   :559.8   Max.   :9999.0
```

```
# Read shapefiles

cov0<-read.csv("GHGC82FL/GHGC82FL.csv")
names(cov0)
```

```
##  [1] "DHSID"                           "GPS_Dataset"
##  [3] "DHSCC"                           "DHSYEAR"
##  [5] "DHSCLUST"                        "SurveyID"
##  [7] "All_Population_Count_2005"       "All_Population_Count_2010"
##  [9] "All_Population_Count_2015"       "Annual_Precipitation_2000"
## [11] "Annual_Precipitation_2005"       "Annual_Precipitation_2010"
## [13] "Annual_Precipitation_2015"       "Aridity_2000"
## [15] "Aridity_2005"                    "Aridity_2010"
## [17] "Aridity_2015"                    "BUILT_Population_1990"
## [19] "BUILT_Population_2000"           "BUILT_Population_2014"
## [21] "Day_Land_Surface_Temp_2000"      "Day_Land_Surface_Temp_2005"
## [23] "Day_Land_Surface_Temp_2010"      "Day_Land_Surface_Temp_2015"
## [25] "Diurnal_Temperature_Range_2000" "Diurnal_Temperature_Range_2005"
## [27] "Diurnal_Temperature_Range_2010" "Diurnal_Temperature_Range_2015"
## [29] "Drought_Episodes"                "Enhanced_Vegetation_Index_1985"
## [31] "Enhanced_Vegetation_Index_1990" "Enhanced_Vegetation_Index_1995"
## [33] "Enhanced_Vegetation_Index_2000" "Enhanced_Vegetation_Index_2005"
## [35] "Enhanced_Vegetation_Index_2010" "Enhanced_Vegetation_Index_2015"
## [37] "Frost_Days_2000"                 "Frost_Days_2005"
## [39] "Frost_Days_2010"                 "Frost_Days_2015"
## [41] "Global_Human_Footprint"          "Gross_Cell_Production"
## [43] "Growing_Season_Length"           "Irrigation"
## [45] "ITN_Coverage_2005"               "ITN_Coverage_2010"
## [47] "ITN_Coverage_2015"               "Land_Surface_Temperature_2000"
## [49] "Land_Surface_Temperature_2005"   "Land_Surface_Temperature_2010"
## [51] "Land_Surface_Temperature_2015"   "Livestock_Cattle"
## [53] "Livestock_Chickens"              "Livestock_Ducks"
## [55] "Livestock_Goats"                 "Livestock_Pigs"
## [57] "Livestock_Sheep"                 "Malaria_Incidence_2000"
## [59] "Malaria_Incidence_2005"          "Malaria_Incidence_2010"
## [61] "Malaria_Incidence_2015"          "Malaria_Prevalence_2000"
## [63] "Malaria_Prevalence_2005"         "Malaria_Prevalence_2010"
## [65] "Malaria_Prevalence_2015"         "Maximum_Temperature_2000"
## [67] "Maximum_Temperature_2005"        "Maximum_Temperature_2010"
## [69] "Maximum_Temperature_2015"        "Mean_Temperature_2000"
## [71] "Mean_Temperature_2005"           "Mean_Temperature_2010"
## [73] "Mean_Temperature_2015"           "Minimum_Temperature_2000"
## [75] "Minimum_Temperature_2005"        "Minimum_Temperature_2010"
```

```
##  [77] "Minimum_Temperature_2015"       "Nightlights_Composite"
##  [79] "Night_Land_Surface_Temp_2000"   "Night_Land_Surface_Temp_2005"
##  [81] "Night_Land_Surface_Temp_2010"   "Night_Land_Surface_Temp_2015"
##  [83] "PET_2000"                       "PET_2005"
##  [85] "PET_2010"                       "PET_2015"
##  [87] "Proximity_to_National_Borders"  "Proximity_to_Protected_Areas"
##  [89] "Proximity_to_Water"             "Rainfall_1985"
##  [91] "Rainfall_1990"                  "Rainfall_1995"
##  [93] "Rainfall_2000"                  "Rainfall_2005"
##  [95] "Rainfall_2010"                  "Rainfall_2015"
##  [97] "Slope"                          "SMOD_Population_1990"
##  [99] "SMOD_Population_2000"           "SMOD_Population_2015"
## [101] "Temperature_April"              "Temperature_August"
## [103] "Temperature_December"           "Temperature_February"
## [105] "Temperature_January"            "Temperature_July"
## [107] "Temperature_June"               "Temperature_March"
## [109] "Temperature_May"                "Temperature_November"
## [111] "Temperature_October"            "Temperature_September"
## [113] "Travel_Times_2000"              "Travel_Times_2015"
## [115] "U5_Population_2000"             "U5_Population_2005"
## [117] "U5_Population_2010"             "U5_Population_2015"
## [119] "UN_Population_Count_2000"       "UN_Population_Count_2005"
## [121] "UN_Population_Count_2010"       "UN_Population_Count_2015"
## [123] "UN_Population_Density_2000"     "UN_Population_Density_2005"
## [125] "UN_Population_Density_2010"     "UN_Population_Density_2015"
## [127] "Wet_Days_2000"                  "Wet_Days_2005"
## [129] "Wet_Days_2010"                  "Wet_Days_2015"
```

```
#summary(cov0)
class(cov0)
```

```
## [1] "data.frame"
```

```
save(cov0,file="cov0.RData")
load("cov0.RData")
#names(cov0)

# Potential covariates:
# Aridity_2015, Enhanced_Vegetation_Index_2015, ITN_Coverage_2015,
# Mean_Temperature_2015, Travel_Times_2015, Annual_Precipitation_2015

# Index variables:
# DHSID, DHSCLUST,


 # extract relevant covariates

cov<-cov0[,c("DHSID","DHSCLUST","Aridity_2015","Enhanced_Vegetation_Index_2015",
      "ITN_Coverage_2015","Mean_Temperature_2015","Annual_Precipitation_2015",
      "Travel_Times_2015")]

names(cov)
```

```
## [1] "DHSID"                          "DHSCLUST"
## [3] "Aridity_2015"                   "Enhanced_Vegetation_Index_2015"
## [5] "ITN_Coverage_2015"              "Mean_Temperature_2015"
## [7] "Annual_Precipitation_2015"      "Travel_Times_2015"
```

```
#summary(cov)

save(cov,file="cov.RData")
load("cov.RData")

#summary(cov)



# Assigning NA to values in R
# https://cran.r-project.org/web/packages/naniar/vignettes/replace-with-na.html

# install.packages("naniar")

library(naniar)

cov1<-cov
class(cov1)
```

```
## [1] "data.frame"
```

```
cov1<-cov1 %>%
  replace_with_na(replace = list(Aridity_2015 =-9999,Enhanced_Vegetation_Index_2015=-9999,
                  ITN_Coverage_2015=-9999,Annual_Precipitation_2015=-9999,
                 Mean_Temperature_2015=-9999,Travel_Times_2015=-9999))

summary(cov1)
```

```
##     DHSID              DHSCLUST        Aridity_2015
##  Length:200         Min.   :  1.00   Min.   :15.80
##  Class :character   1st Qu.: 50.75   1st Qu.:20.15
##  Mode  :character   Median :100.50   Median :23.73
##                     Mean   :100.50   Mean   :23.39
##                     3rd Qu.:150.25   3rd Qu.:27.21
##                     Max.   :200.00   Max.   :34.09
##                                      NA's   :26
##  Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015
##  Min.   :   0                   Min.   :0.2235    Min.   :26.64
##  1st Qu.:2589                   1st Qu.:0.4983    1st Qu.:27.23
##  Median :3307                   Median :0.6032    Median :27.62
##  Mean   :3474                   Mean   :0.5701    Mean   :27.84
##  3rd Qu.:4529                   3rd Qu.:0.6707    3rd Qu.:28.32
##  Max.   :5362                   Max.   :0.8873    Max.   :29.25
##  NA's   :13                     NA's   :10        NA's   :26
##  Annual_Precipitation_2015 Travel_Times_2015
##  Min.   : 68.03            Min.   :  0.00
##  1st Qu.: 76.70            1st Qu.:  0.60
##  Median : 82.06            Median : 21.06
##  Mean   : 81.38            Mean   : 40.85
##  3rd Qu.: 84.70            3rd Qu.: 56.80
##  Max.   :109.38            Max.   :263.83
##  NA's   :26                NA's   :8
```

```
dim(cov1)
```

```
## [1] 200   8
```

```
# Or to set -9999 values to NAs for all variables, use
# https://naniar.njtierney.com/articles/replace-with-na.html

cov1<-cov1 %>% replace_with_na_all(condition = ~.x == -9999)
#summary(cov1)


library(mice)
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
tempData <- mice(cov1,m=5,maxit=50,meth='pmm',seed=500)
```

```
##
##  iter imp variable
##   1   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   1   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   1   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   1   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   1   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   2   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   2   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   2   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   2   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   2   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   3   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   3   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   3   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   3   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   3   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   4   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   4   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   4   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
```

```
##   4   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   4   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   5   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   5   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   5   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   5   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   5   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   6   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   6   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   6   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   6   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   6   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   7   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   7   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   7   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   7   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   7   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   8   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   8   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
```

```
##   8   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   8   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   8   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   9   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   9   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   9   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   9   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   9   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Precip
itation_2015  Travel_Times_2015
##   10   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   10   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   10   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   10   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   10   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   11   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   11   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   11   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   11   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   11   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   12   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   12   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   12   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   12   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   12   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   13   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   13   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   13   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   13   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   13   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   14   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   14   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   14   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   14   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   14   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   15   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   15   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   15   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   15   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   15   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   16   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   16   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   16   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   16   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   16   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   17   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   17   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   17   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   17   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   17   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   18   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   18   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   18   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   18   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   18   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   19   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   19   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   19   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   19   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   19   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   20   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   20   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   20   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   20   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   20   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   21   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   21   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   21   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   21   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   21   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   22   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   22   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   22   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   22   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   22   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   23   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   23   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   23   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##    23    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    23    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    24    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    24    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    24    3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    24    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    24    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    25    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    25    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    25    3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    25    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    25    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    26    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    26    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    26    3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    26    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    26    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    27    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    27    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   27   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   27   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   27   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   28   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   28   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   28   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   28   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   28   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   29   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   29   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   29   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   29   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   29   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   30   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   30   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   30   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   30   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   30   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   31   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   31   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   31   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   31   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   31   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   32   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   32   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   32   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   32   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   32   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   33   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   33   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   33   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   33   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   33   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   34   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   34   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   34   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   34   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   34   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   35   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   35   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   35   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   35   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   35   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   36   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   36   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   36   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   36   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   36   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   37   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   37   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   37   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   37   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   37   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   38   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   38   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   38   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   38   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
## 38 5 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 39 1 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 39 2 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 39 3 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 39 4 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 39 5 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 40 1 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 40 2 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 40 3 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 40 4 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 40 5 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 41 1 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 41 2 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 41 3 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 41 4 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 41 5 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 42 1 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 42 2 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
## 42 3 Aridity_2015 Enhanced_Vegetation_Index_2015 ITN_Coverage_2015 Mean_Temperature_2015 Annual_Preci
pitation_2015 Travel_Times_2015
```

```
##    42    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    42    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    43    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    43    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    43    3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    43    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    43    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    44    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    44    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    44    3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    44    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    44    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    45    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    45    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    45    3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    45    4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    45    5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    46    1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##    46    2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   46   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   46   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   46   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   47   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   47   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   47   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   47   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   47   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   48   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   48   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   48   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   48   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   48   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   49   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   49   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   49   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   49   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   49   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   50   1  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
##   50   2  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   50   3  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   50   4  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
##   50   5  Aridity_2015  Enhanced_Vegetation_Index_2015  ITN_Coverage_2015  Mean_Temperature_2015  Annual_Preci
pitation_2015  Travel_Times_2015
```

```
## Warning: Number of logged events: 1
```

```
#summary(tempData)

# Get completed datasets (observed and imputed)
cov2 <- complete(tempData)
#summary(cov2)
dim(cov2)
```

```
## [1] 200   8
```

```
save(cov2,file="cov2.RData")



# merge outcome with covariate data

mal<-mal.imp[,c("hv001","hhid","hv002","pos")]

names(mal)
```

```
## [1] "hv001" "hhid"  "hv002" "pos"
```

```
library(dplyr)

mal_d <- group_by(mal, hv001) %>%
  summarize(total = n(),
            mal_count = sum(pos),
            mal_prev =  mal_count/total
            )

names(mal_d)
```

```
## [1] "hv001"     "total"     "mal_count" "mal_prev"
```

```
dim(mal_d)
```

```
## [1] 200   4
```

```
summary(mal_d)
```

```
##      hv001            total         mal_count        mal_prev
##  Min.   :  1.00   Min.   : 3.00   Min.   : 0.00   Min.   :0.0000
##  1st Qu.: 50.75   1st Qu.: 9.00   1st Qu.: 0.00   1st Qu.:0.0000
##  Median :100.50   Median :13.00   Median : 2.00   Median :0.1791
##  Mean   :100.50   Mean   :14.34   Mean   : 3.59   Mean   :0.2211
##  3rd Qu.:150.25   3rd Qu.:18.00   3rd Qu.: 5.00   3rd Qu.:0.3750
##  Max.   :200.00   Max.   :60.00   Max.   :26.00   Max.   :0.9000
```

```
mal0<-mal_d
names(mal0)
```

```
## [1] "hv001"     "total"     "mal_count" "mal_prev"
```

```
# Renaming clusters in mdata
library(gdata)
```

```
##
## Attaching package: 'gdata'
```

```
## The following objects are masked from 'package:dplyr':
##
##     combine, first, last, starts_with
```

```
## The following objects are masked from 'package:raster':
##
##     resample, trim
```

```
## The following object is masked from 'package:stats':
##
##     nobs
```

```
## The following object is masked from 'package:utils':
##
##     object.size
```

```
## The following object is masked from 'package:base':
##
##     startsWith
```

```
mal0 <- rename.vars(mal0, "hv001", "DHSCLUST")
```

```
##
## Changing in mal0
## From: hv001
## To:   DHSCLUST
```

```
class(mal0)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```
#Combining mdata with shapefile (coordinates)
idx <- match(mal0$DHSCLUST,gh@data$DHSCLUST)
crds <- coordinates(gh)[idx,]
mal0$lon <- crds[,1]
mal0$lat <- crds[,2]
class(mal0)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```
summary(mal0)
```

```
##     DHSCLUST          total         mal_count        mal_prev
##  Min.   :  1.00   Min.   : 3.00   Min.   : 0.00   Min.   :0.0000
##  1st Qu.: 50.75   1st Qu.: 9.00   1st Qu.: 0.00   1st Qu.:0.0000
##  Median :100.50   Median :13.00   Median : 2.00   Median :0.1791
##  Mean   :100.50   Mean   :14.34   Mean   : 3.59   Mean   :0.2211
##  3rd Qu.:150.25   3rd Qu.:18.00   3rd Qu.: 5.00   3rd Qu.:0.3750
##  Max.   :200.00   Max.   :60.00   Max.   :26.00   Max.   :0.9000
##       lon                lat
##  Min.   :-3.1526   Min.   : 0.000
##  1st Qu.:-1.7543   1st Qu.: 5.618
##  Median :-0.8832   Median : 6.473
##  Mean   :-1.0360   Mean   : 6.905
##  3rd Qu.:-0.2296   3rd Qu.: 8.110
##  Max.   : 1.1973   Max.   :11.066
```

```
dim(mal0)
```

```
## [1] 200   6
```

```
#plot(mal0)

mal0<-mal0[!(mal0$lat<4),]# Remove one coordinate in the sea
class(mal0)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

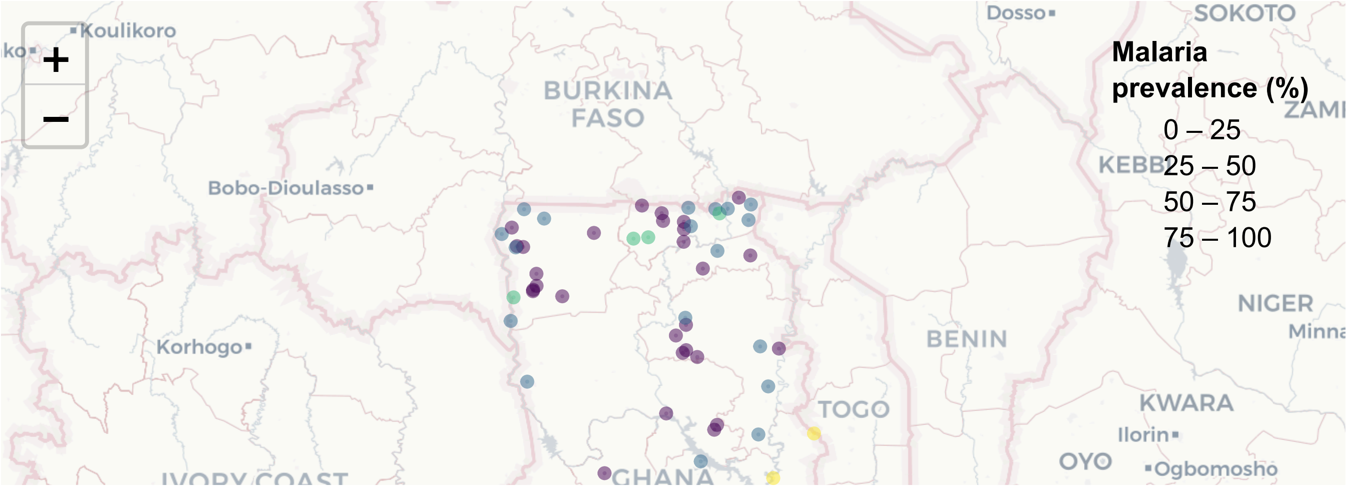
```
dim(mal0)
```

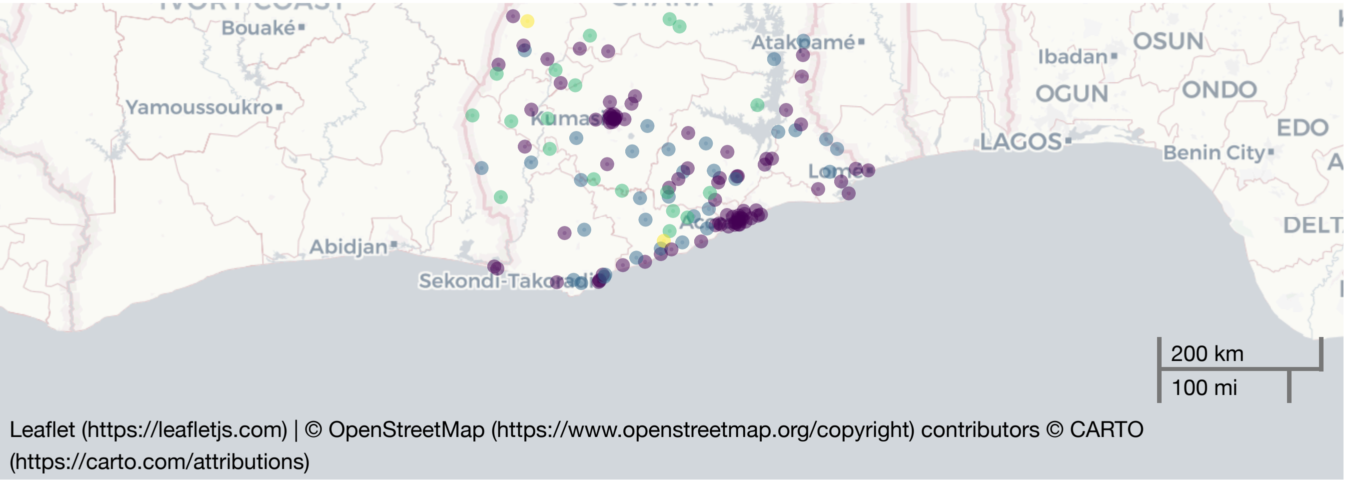
```
## [1] 192   6
```

```
mal0$mal_pc<-(mal0$mal_prev)*100
summary(mal0)
```

```
##     DHSCLUST          total         mal_count         mal_prev     
##  Min.   :  1.00   Min.   : 3.00   Min.   : 0.000   Min.   :0.0000  
##  1st Qu.: 50.75   1st Qu.: 9.00   1st Qu.: 0.000   1st Qu.:0.0000  
##  Median :101.50   Median :13.00   Median : 2.000   Median :0.1791  
##  Mean   :101.04   Mean   :14.22   Mean   : 3.521   Mean   :0.2223  
##  3rd Qu.:151.25   3rd Qu.:18.00   3rd Qu.: 5.000   3rd Qu.:0.3750  
##  Max.   :200.00   Max.   :43.00   Max.   :19.000   Max.   :0.9000  
##       lon               lat             mal_pc     
##  Min.   :-3.1526   Min.   : 4.894   Min.   : 0.00  
##  1st Qu.:-1.7744   1st Qu.: 5.649   1st Qu.: 0.00  
##  Median :-0.9759   Median : 6.587   Median :17.91  
##  Mean   :-1.0792   Mean   : 7.192   Mean   :22.23  
##  3rd Qu.:-0.2612   3rd Qu.: 8.502   3rd Qu.:37.50  
##  Max.   : 1.1973   Max.   :11.066   Max.   :90.00  
```

```
save(mal0,file="mal0.RData")
load("mal0.RData")

library(leaflet)

#pal <- colorBin("viridis", bins = c(0, 0.25, 0.5, 0.75, 1),alpha = F,reverse = F)

#pal <- colorBin("plasma", bins = c(0, 0.25, 0.5, 0.75, 1),alpha = F,reverse = F)

#pal <- colorBin("magma", bins = c(0, 0.25, 0.5, 0.75, 1),alpha = F,reverse = F)


#pal <- colorBin("RGB", bins = c(0, 0.25, 0.5, 0.75, 1),alpha = F,reverse = F)

pal <- colorBin("viridis", bins = c(0, 25, 50, 75, 100),alpha = F,
       reverse = F)

# pal <- colorBin("YlOrRd", domain = 0:100)

# Malaria prevalence
leaflet(mal0) %>%  addProviderTiles(providers$CartoDB.Positron) %>%
  addCircles(lng = ~lon, lat = ~lat, color = ~pal(mal_pc)) %>%
   addLegend("topright", pal = pal, values = ~mal_pc, title = "Malaria<br>prevalence (%)") %>%
  addScaleBar(position = c("bottomright"))
```






```
# Prepare for spatial modelling

mal_d2<-merge(mal_d,cov2,by.x="hv001",by.y="DHSCLUST",all.x=T)
class(mal_d2)
```

```
## [1] "data.frame"
```

```
summary(mal_d2)
```

```
##      hv001            total         mal_count        mal_prev
##  Min.   :  1.00   Min.   : 3.00   Min.   : 0.00   Min.   :0.0000
##  1st Qu.: 50.75   1st Qu.: 9.00   1st Qu.: 0.00   1st Qu.:0.0000
##  Median :100.50   Median :13.00   Median : 2.00   Median :0.1791
##  Mean   :100.50   Mean   :14.34   Mean   : 3.59   Mean   :0.2211
##  3rd Qu.:150.25   3rd Qu.:18.00   3rd Qu.: 5.00   3rd Qu.:0.3750
##  Max.   :200.00   Max.   :60.00   Max.   :26.00   Max.   :0.9000
##     DHSID             Aridity_2015   Enhanced_Vegetation_Index_2015
##  Length:200         Min.   :15.80   Min.   :   0
##  Class :character   1st Qu.:20.15   1st Qu.:2589
##  Mode  :character   Median :24.11   Median :3303
##                     Mean   :23.50   Mean   :3469
##                     3rd Qu.:26.62   3rd Qu.:4524
##                     Max.   :34.09   Max.   :5362
##  ITN_Coverage_2015 Mean_Temperature_2015 Annual_Precipitation_2015
##  Min.   :0.2235    Min.   :26.64         Min.   : 68.03
##  1st Qu.:0.4951    1st Qu.:27.23         1st Qu.: 76.70
##  Median :0.6032    Median :27.62         Median : 82.52
##  Mean   :0.5694    Mean   :27.82         Mean   : 81.38
##  3rd Qu.:0.6669    3rd Qu.:28.27         3rd Qu.: 84.70
##  Max.   :0.8873    Max.   :29.25         Max.   :109.38
##  Travel_Times_2015
##  Min.   :  0.0000
##  1st Qu.:  0.6188
##  Median : 21.6854
##  Mean   : 42.3041
##  3rd Qu.: 60.3445
##  Max.   :263.8283
```

```
dim(mal_d2)
```

```
## [1] 200  11
```

```
save(mal_d2,file="mal_d.RData")
load("mal_d2.RData")

#mal_d3<-mal_d2[which(mal_d2$Aridity_2015> -9999==NA),]
#mal_d3<-mal_d3[which(mal_d3$Enhanced_Vegetation_Index_2015> -9999),]<-NA
#mal_d3<-mal_d3[which(mal_d3$ITN_Coverage_2015> -9999),]
#mal_d3<-mal_d3[which(mal_d3$ITN_Coverage_2015> -9999),]

names(mal_d2)
```

```
##  [1] "hv001"                          "total"
##  [3] "mal_count"                      "mal_prev"
##  [5] "DHSID"                          "Aridity_2015"
##  [7] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
##  [9] "Mean_Temperature_2015"          "Annual_Precipitation_2015"
## [11] "Travel_Times_2015"              "DHSCLUST"
```

```
summary(mal_d2)
```

```
##      hv001            total         mal_count        mal_prev     
##  Min.   :  1.00   Min.   : 3.00   Min.   : 0.00   Min.   :0.0000  
##  1st Qu.: 50.75   1st Qu.: 9.00   1st Qu.: 0.00   1st Qu.:0.0000  
##  Median :100.50   Median :13.00   Median : 2.00   Median :0.1791  
##  Mean   :100.50   Mean   :14.34   Mean   : 3.59   Mean   :0.2211  
##  3rd Qu.:150.25   3rd Qu.:18.00   3rd Qu.: 5.00   3rd Qu.:0.3750  
##  Max.   :200.00   Max.   :60.00   Max.   :26.00   Max.   :0.9000  
##     DHSID            Aridity_2015   Enhanced_Vegetation_Index_2015
##  Length:200         Min.   :15.80   Min.   :   0                  
##  Class :character   1st Qu.:20.15   1st Qu.:2589                  
##  Mode  :character   Median :24.11   Median :3303                  
##                     Mean   :23.50   Mean   :3469                  
##                     3rd Qu.:26.62   3rd Qu.:4524                  
##                     Max.   :34.09   Max.   :5362                  
##  ITN_Coverage_2015 Mean_Temperature_2015 Annual_Precipitation_2015
##  Min.   :0.2235    Min.   :26.64         Min.   : 68.03           
##  1st Qu.:0.4951    1st Qu.:27.23         1st Qu.: 76.70           
##  Median :0.6032    Median :27.62         Median : 82.52           
##  Mean   :0.5694    Mean   :27.82         Mean   : 81.38           
##  3rd Qu.:0.6669    3rd Qu.:28.27         3rd Qu.: 84.70           
##  Max.   :0.8873    Max.   :29.25         Max.   :109.38           
##  Travel_Times_2015     DHSCLUST     
##  Min.   :  0.0000   Min.   :  1.00  
##  1st Qu.:  0.6188   1st Qu.: 50.75  
##  Median : 21.6854   Median :100.50  
##  Mean   : 42.3041   Mean   :100.50  
##  3rd Qu.: 60.3445   3rd Qu.:150.25  
##  Max.   :263.8283   Max.   :200.00  
```

```
dim(mal_d2)
```

```
## [1] 200  12
```

```
mal_d2$DHSCLUST<-mal_d2$hv001
save(mal_d2,file="mal_d2.RData")
load("mal_d2.RData")

summary(mal_d2)
```

```
##      hv001            total         mal_count        mal_prev     
##  Min.   :  1.00   Min.   : 3.00   Min.   : 0.00   Min.   :0.0000  
##  1st Qu.: 50.75   1st Qu.: 9.00   1st Qu.: 0.00   1st Qu.:0.0000  
##  Median :100.50   Median :13.00   Median : 2.00   Median :0.1791  
##  Mean   :100.50   Mean   :14.34   Mean   : 3.59   Mean   :0.2211  
##  3rd Qu.:150.25   3rd Qu.:18.00   3rd Qu.: 5.00   3rd Qu.:0.3750  
##  Max.   :200.00   Max.   :60.00   Max.   :26.00   Max.   :0.9000  
##     DHSID            Aridity_2015   Enhanced_Vegetation_Index_2015
##  Length:200         Min.   :15.80   Min.   :   0                  
##  Class :character   1st Qu.:20.15   1st Qu.:2589                  
##  Mode  :character   Median :24.11   Median :3303                  
##                     Mean   :23.50   Mean   :3469                  
##                     3rd Qu.:26.62   3rd Qu.:4524                  
##                     Max.   :34.09   Max.   :5362                  
##  ITN_Coverage_2015 Mean_Temperature_2015 Annual_Precipitation_2015
##  Min.   :0.2235    Min.   :26.64         Min.   : 68.03           
##  1st Qu.:0.4951    1st Qu.:27.23         1st Qu.: 76.70           
##  Median :0.6032    Median :27.62         Median : 82.52           
##  Mean   :0.5694    Mean   :27.82         Mean   : 81.38           
##  3rd Qu.:0.6669    3rd Qu.:28.27         3rd Qu.: 84.70           
##  Max.   :0.8873    Max.   :29.25         Max.   :109.38           
##  Travel_Times_2015     DHSCLUST     
##  Min.   :  0.0000   Min.   :  1.00  
##  1st Qu.:  0.6188   1st Qu.: 50.75  
##  Median : 21.6854   Median :100.50  
##  Mean   : 42.3041   Mean   :100.50  
##  3rd Qu.: 60.3445   3rd Qu.:150.25  
##  Max.   :263.8283   Max.   :200.00  
```

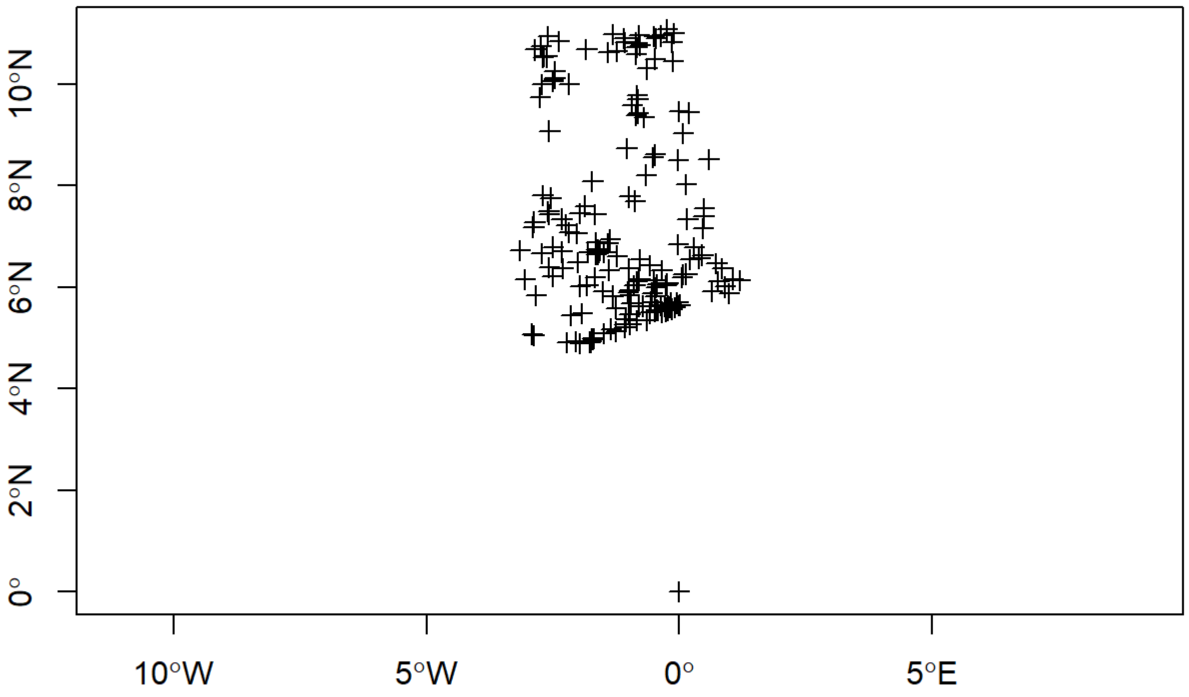
```
names(mal_d2)
```

```
##  [1] "hv001"                           "total"
##  [3] "mal_count"                       "mal_prev"
##  [5] "DHSID"                           "Aridity_2015"
##  [7] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
##  [9] "Mean_Temperature_2015"           "Annual_Precipitation_2015"
## [11] "Travel_Times_2015"               "DHSCLUST"
```

```
dim(mal_d2)
```

```
## [1] 200  12
```

```
# merge shapefile and the combined dataset

library(sp)

mal_sp<-merge(gh,mal_d2,by="DHSCLUST",all.x=T)

plot(mal_sp,axes=T)
```



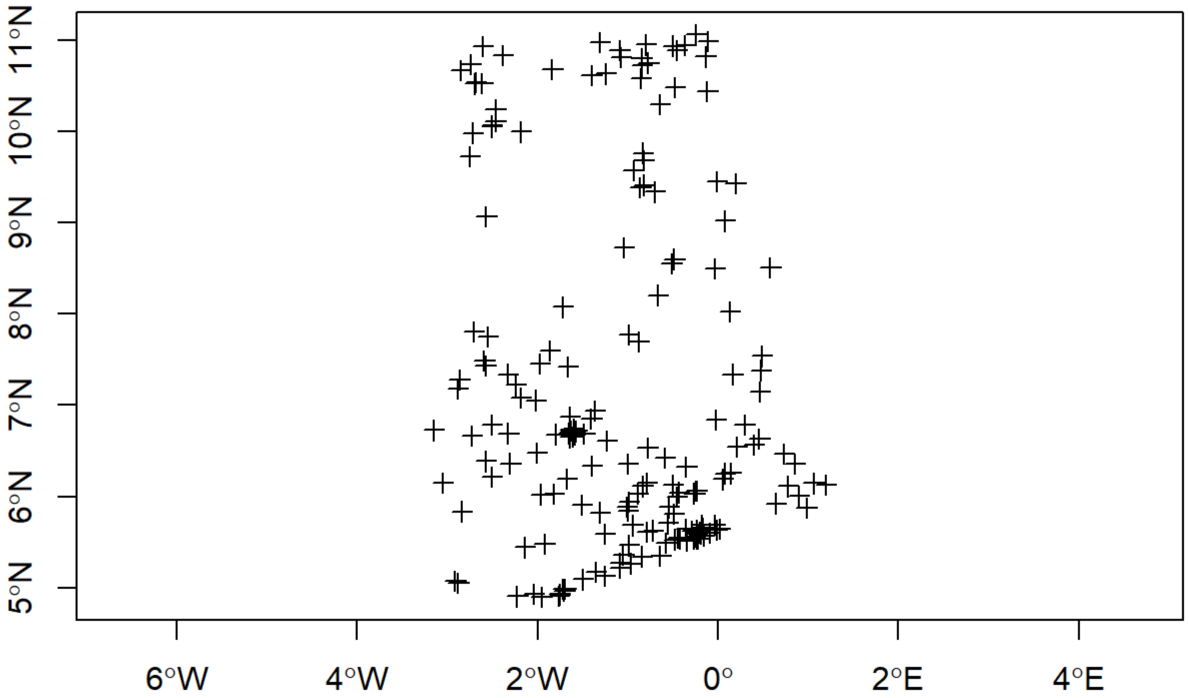

```
#Combining mdata with shapefile (coordinates)
idx <- match(mal_sp$DHSCLUST,gh@data$DHSCLUST)
crds <- coordinates(gh)[idx,]
mal_sp$lon <- crds[,1]
mal_sp$lat <- crds[,2]
class(mal_sp)
```

```
## [1] "SpatialPointsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
#summary(mal_sp)

plot(mal_sp,axes=T)

mal_sp<-mal_sp[!(mal_sp$lat<4),]# Remove one coordinate in the sea
dim(mal_sp)
```

```
## [1] 192  33
```

```
plot(mal_sp,axes=T)
```




```
dim(mal_sp)
```

```
## [1] 192  33
```

```
names(mal_sp)
```

```
##  [1] "DHSCLUST"                         "DHSID.x"
##  [3] "DHSCC"                            "DHSYEAR"
##  [5] "CCFIPS"                           "ADM1FIPS"
##  [7] "ADM1FIPSNA"                       "ADM1SALBNA"
##  [9] "ADM1SALBCO"                       "ADM1DHS"
## [11] "ADM1NAME"                         "DHSREGCO"
## [13] "DHSREGNA"                         "SOURCE"
## [15] "URBAN_RURA"                       "LATNUM"
## [17] "LONGNUM"                          "ALT_GPS"
## [19] "ALT_DEM"                          "DATUM"
## [21] "hv001"                            "total"
## [23] "mal_count"                        "mal_prev"
## [25] "DHSID.y"                          "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"            "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"                "lon"
## [33] "lat"
```

```
save(mal_sp,file="mal_sp.RData")

load("mal_sp.RData")


# Plotting with leaflet

mal_sp2<-as.data.frame(mal_sp)
class(mal_sp2)
```

```
## [1] "data.frame"
```

```
names(mal_sp2)
```

```
##  [1] "DHSCLUST"                         "DHSID.x"
##  [3] "DHSCC"                            "DHSYEAR"
##  [5] "CCFIPS"                           "ADM1FIPS"
##  [7] "ADM1FIPSNA"                       "ADM1SALBNA"
##  [9] "ADM1SALBCO"                       "ADM1DHS"
## [11] "ADM1NAME"                         "DHSREGCO"
## [13] "DHSREGNA"                         "SOURCE"
## [15] "URBAN_RURA"                       "LATNUM"
## [17] "LONGNUM"                          "ALT_GPS"
## [19] "ALT_DEM"                          "DATUM"
## [21] "hv001"                            "total"
## [23] "mal_count"                        "mal_prev"
## [25] "DHSID.y"                          "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"            "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"                "lon"
## [33] "lat"                              "coords.x1"
## [35] "coords.x2"                        "coords.x3"
```

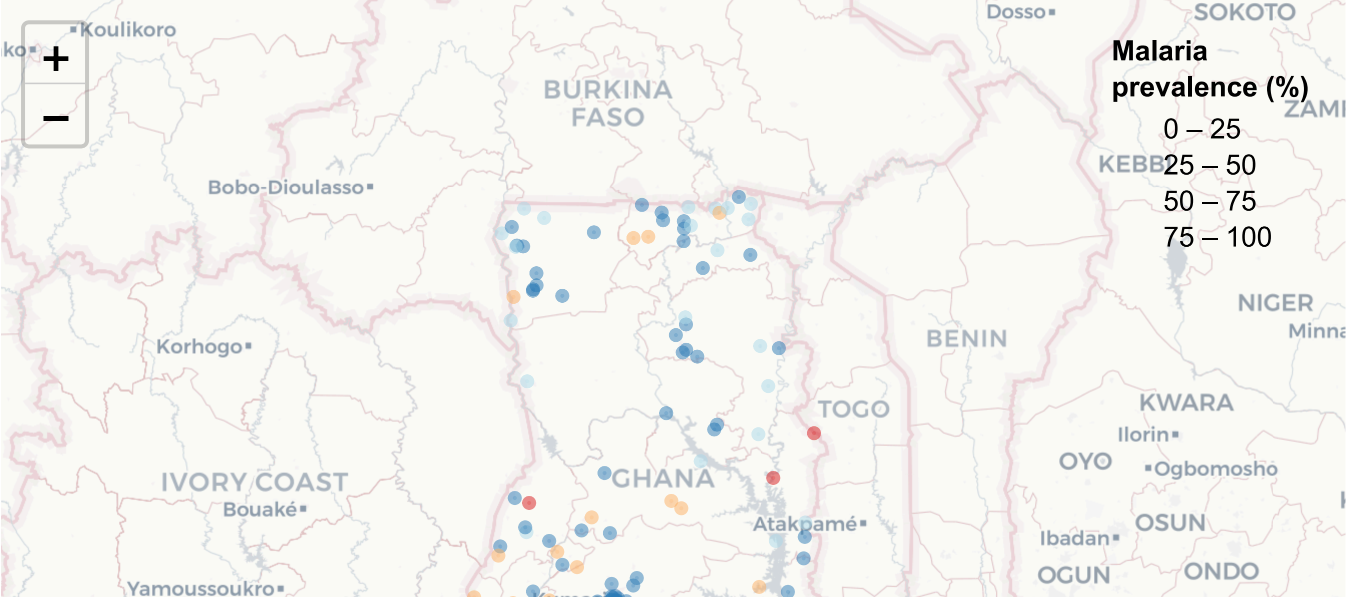
```
save(mal_sp2,file="mal_sp2.RData")

load("mal_sp2.RData")
#summary(mal_sp2)
dim(mal_sp2)
```

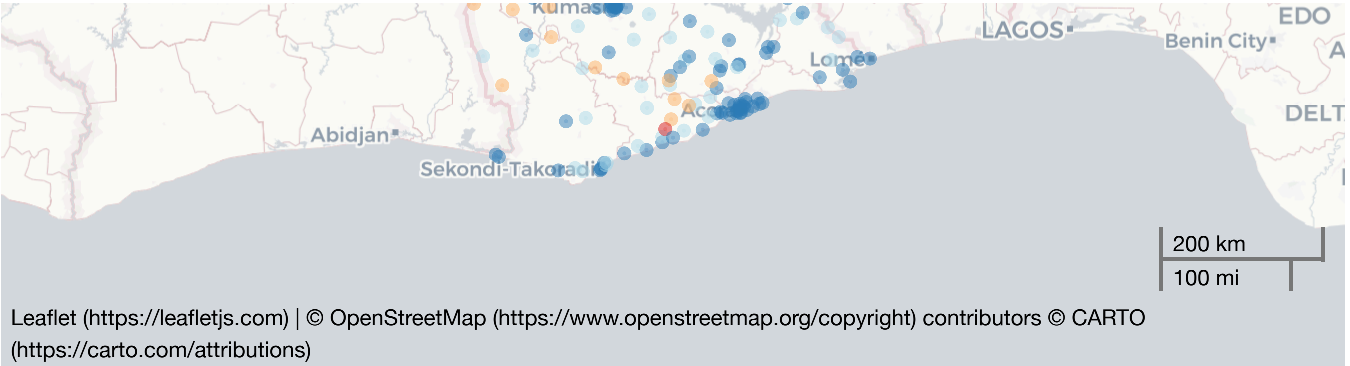
```
## [1] 192  36
```

```
names(mal_sp2)
```

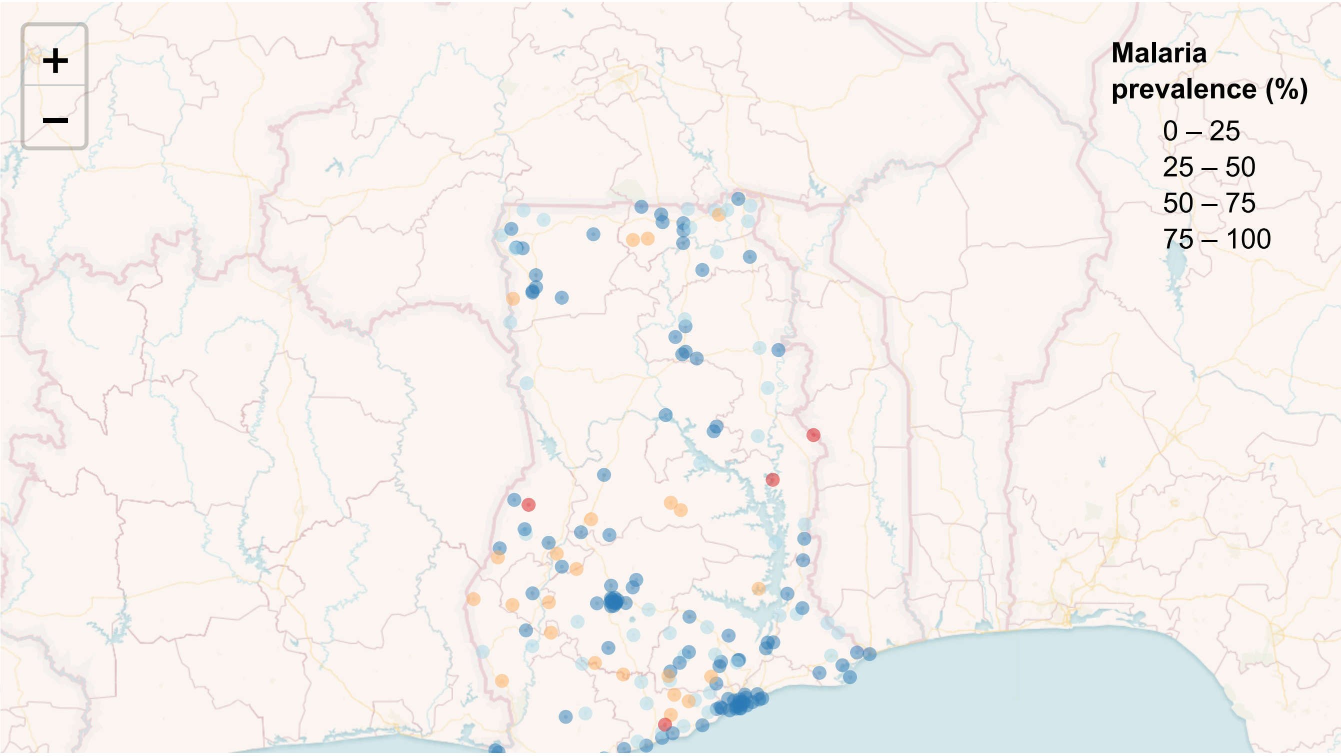
```
##  [1] "DHSCLUST"                        "DHSID.x"
##  [3] "DHSCC"                           "DHSYEAR"
##  [5] "CCFIPS"                          "ADM1FIPS"
##  [7] "ADM1FIPSNA"                      "ADM1SALBNA"
##  [9] "ADM1SALBCO"                      "ADM1DHS"
## [11] "ADM1NAME"                        "DHSREGCO"
## [13] "DHSREGNA"                        "SOURCE"
## [15] "URBAN_RURA"                      "LATNUM"
## [17] "LONGNUM"                         "ALT_GPS"
## [19] "ALT_DEM"                         "DATUM"
## [21] "hv001"                           "total"
## [23] "mal_count"                       "mal_prev"
## [25] "DHSID.y"                         "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"           "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"               "lon"
## [33] "lat"                             "coords.x1"
## [35] "coords.x2"                       "coords.x3"
```

```
mal_sp2$mal_pc<-(mal_sp2$mal_prev)*100

library(leaflet)
pal <- colorBin("viridis", bins = c(0, 25, 50, 75, 100),reverse=F)
pal <- colorBin("RdYlBu", bins = c(0, 25, 50, 75, 100),reverse=T)

#pal <- colorBin("YlOrRd", domain = 0:100)

# pal<-brewer.pal(5,"YlOrRd") # 5 colour bands for map plot

# pal <- colorBin("YlOrRd", bins = c(0, 25, 50, 75, 100))


pal <- colorBin("RdYlBu", bins = c(0, 25, 50, 75, 100),reverse=T)


# Malaria prevalence
leaflet(mal_sp2) %>%  addProviderTiles(providers$CartoDB.Positron) %>%
  addCircles(lng = ~lon, lat = ~lat, color = ~pal(mal_pc)) %>%
   addLegend("topright", pal = pal, values = ~mal_pc, title = "Malaria<br>prevalence (%)") %>%
  addScaleBar(position = c("bottomright"))
```

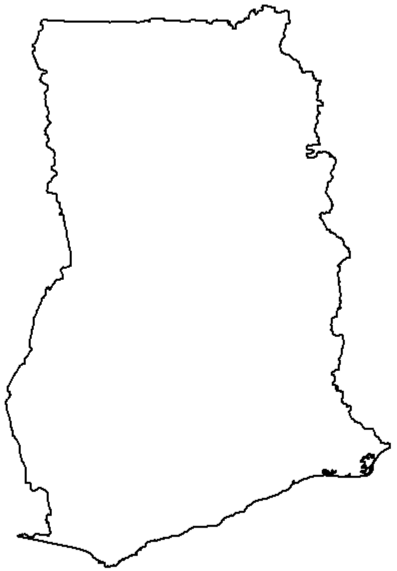




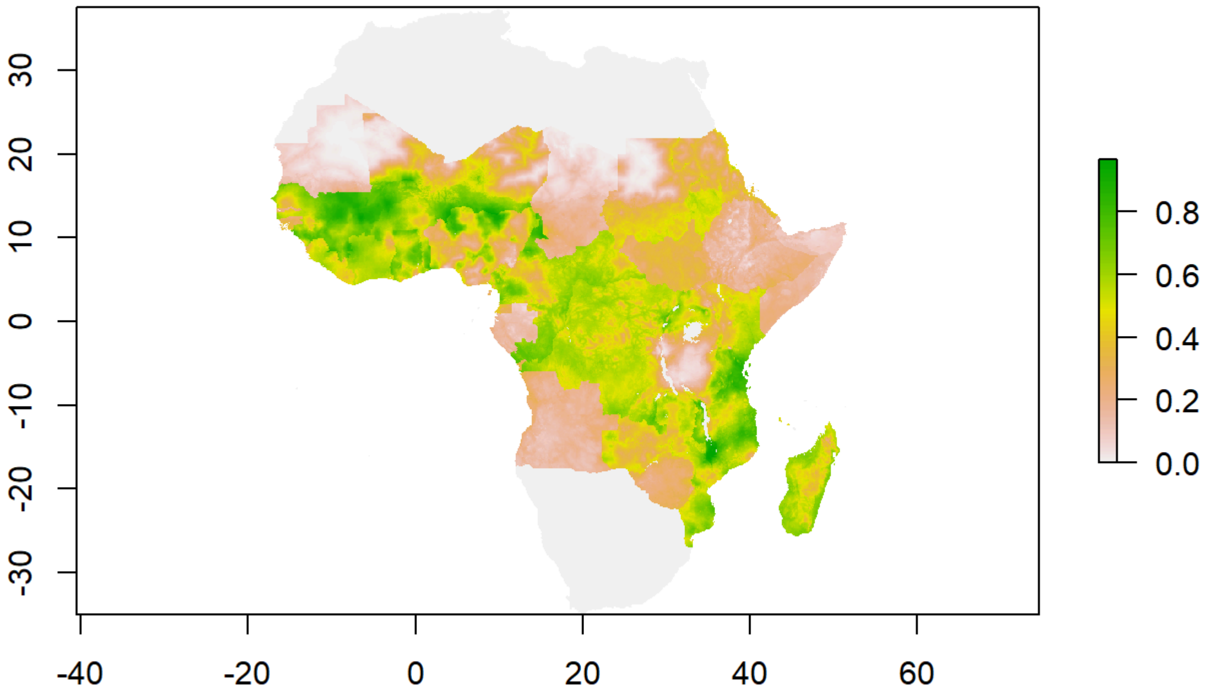

```
# Same but for No Labels (clear backgrounds) using 'CartoDB.VoyagerNoLabels'

leaflet(mal_sp2) %>%  addProviderTiles(providers$CartoDB.VoyagerNoLabels) %>%
  addCircles(lng = ~lon, lat = ~lat, color = ~pal(mal_pc)) %>%
   addLegend("topright", pal = pal, values = ~mal_pc, title = "Malaria<br>prevalence (%)") %>%
  addScaleBar(position = c("bottomright"))
```



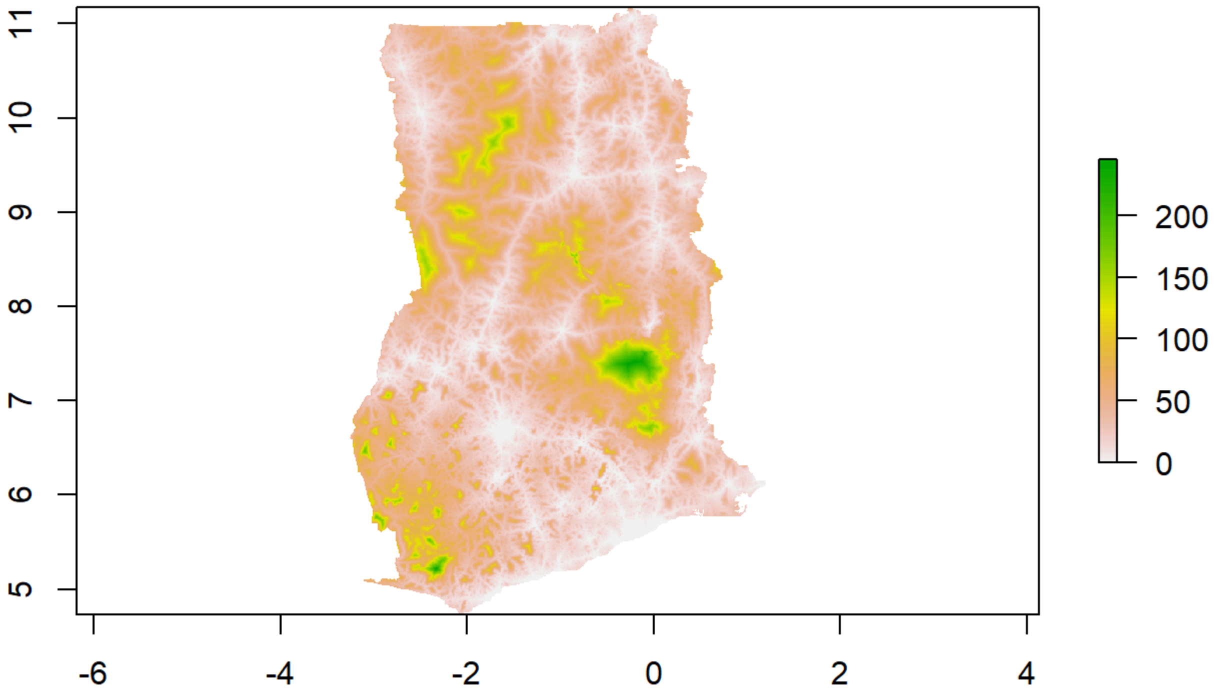

```
#####################################################################################

# Modelling

# Prepare covariates for prediction

library(raster)

#install.packages("rgdal")
library(rgdal)

# Load Ghana polygon

load("gh.border.RData")
plot(gh.border)
```



```
# ITN data from the website below
# https://malariaatlas.org/research-project/metrics-of-insecticide-treated-nets-distribution/

str_name1<-'ITN_2019_use_mean.tif'
ITN_Coverage_2015=raster(str_name1)# import raster file for ITN
plot(ITN_Coverage_2015)
```



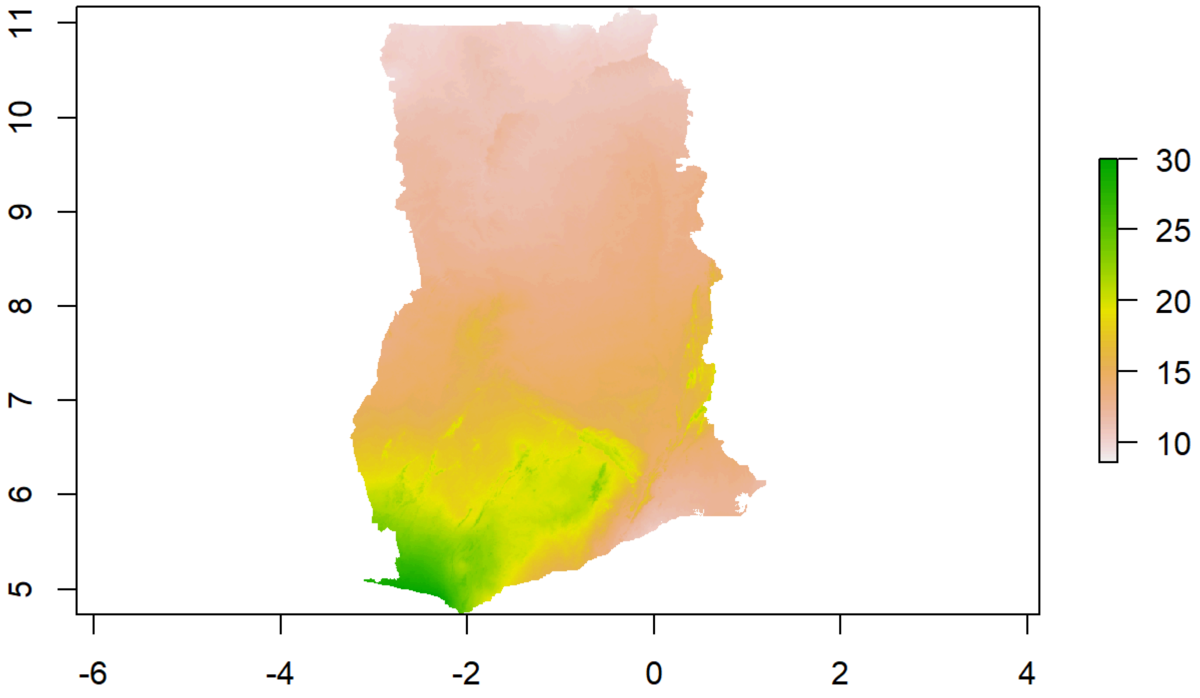

```
save(ITN_Coverage_2015,file="ITN_Coverage_2015.RData")

# https://forobs.jrc.ec.europa.eu/products/gam/download.php

str_name2<-'GHA_traveltime_1k.tif'
Travel_Times_2015=raster(str_name2)*0.4# import raster file for travel time to city (>=50,000)
plot(Travel_Times_2015)
```




```
summary(Travel_Times_2015)
```

```
##         GHA_traveltime_1k
## Min.                  0.0
## 1st Qu.              19.6
## Median               36.8
## 3rd Qu.              60.0
## Max.                245.6
## NA's             132006.0
```

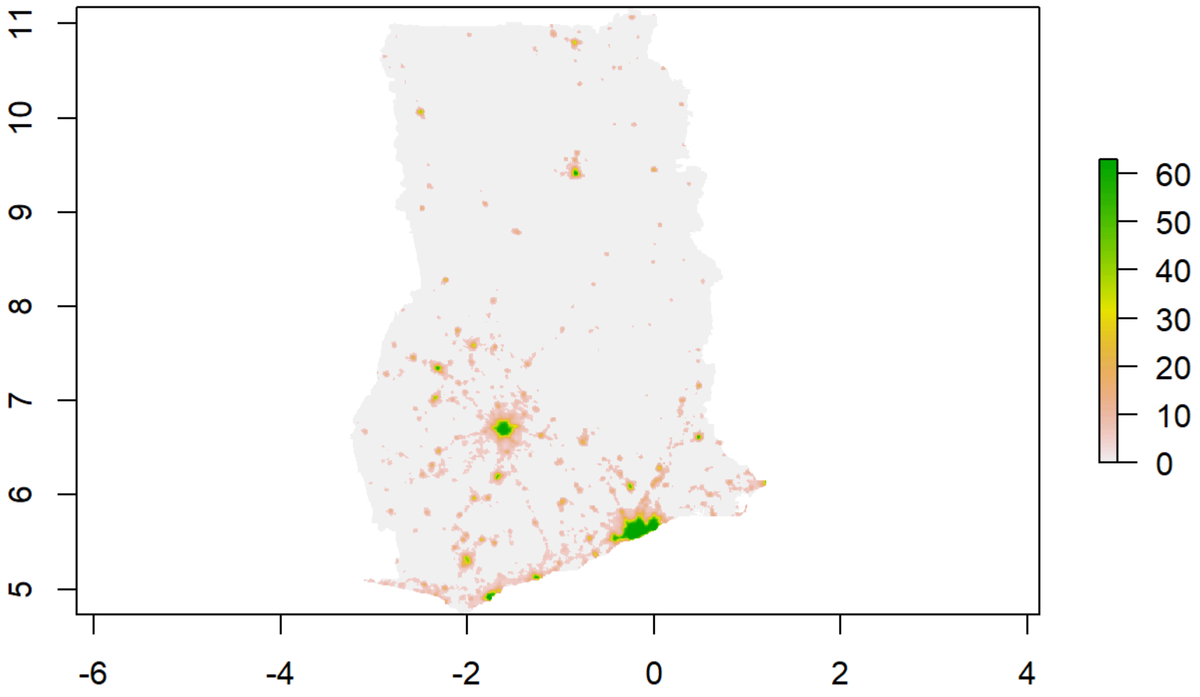
```
save(Travel_Times_2015,file="Travel_Times_2015.RData")

str_name3<-'GHA_CGIAR_AI_1k_final.tif'
Aridity_2015=raster(str_name3)*20# import raster file for Aridity
plot(Aridity_2015)
```




```
summary(Aridity_2015)
```

```
##         GHA_CGIAR_AI_1k_final
## Min.                     8.552
## 1st Qu.                 11.858
## Median                  13.586
## 3rd Qu.                 16.304
## Max.                    30.014
## NA's                131999.000
```

```
save(Aridity_2015,file="Aridity_2015.RData")

str_name4<-'GHA_VIIRS_NL_1k.tif'
Enhanced_Vegetation_Index_2015=raster(str_name4)# import raster file for travel time to city (>=50,000)
plot(Enhanced_Vegetation_Index_2015)
```

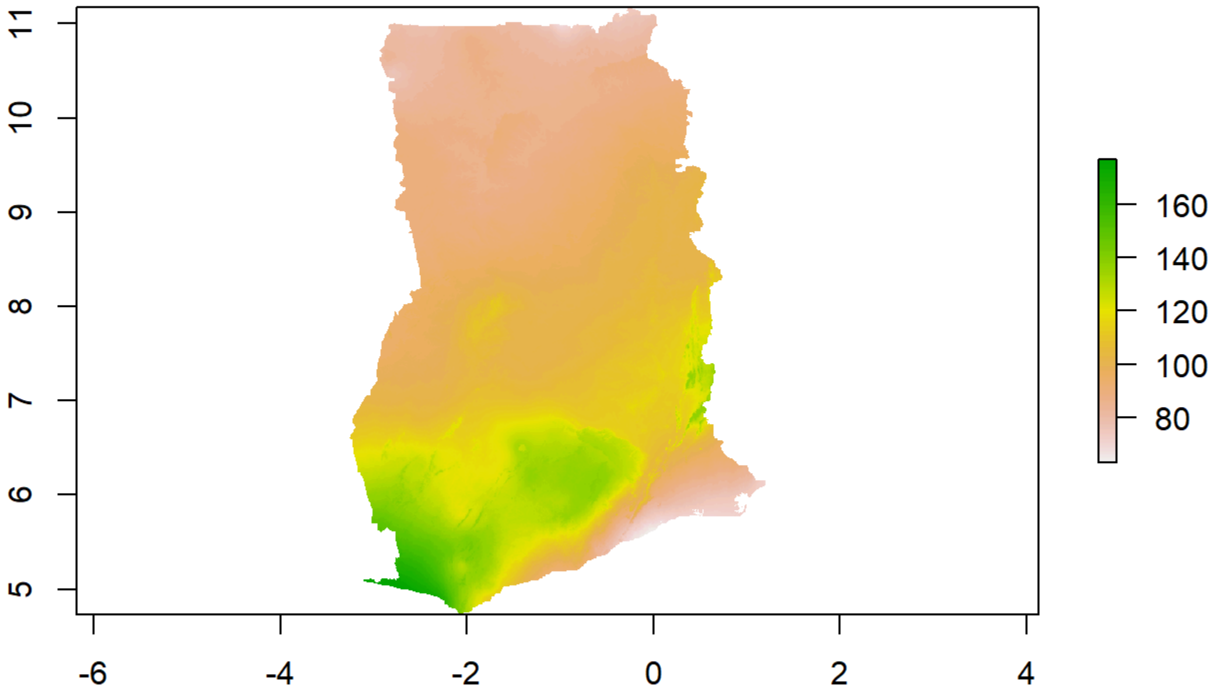



```
summary(Enhanced_Vegetation_Index_2015)
```

```
## Warning in .local(object, ...): summary is an estimate based on a sample of 1e+05 cells (24.18% of all cells)
```

```
##         GHA_VIIRS_NL_1k
## Min.                  0
## 1st Qu.               0
## Median                0
## 3rd Qu.               0
## Max.                 63
## NA's             132036
```

```
save(Enhanced_Vegetation_Index_2015,file="Enhanced_Vegetation_Index_2015.RData")

str_name5<-'GHA_precip_1k.tif'
Annual_Precipitation_2015=raster(str_name5)# import raster file for precipitation
plot(Annual_Precipitation_2015)
```

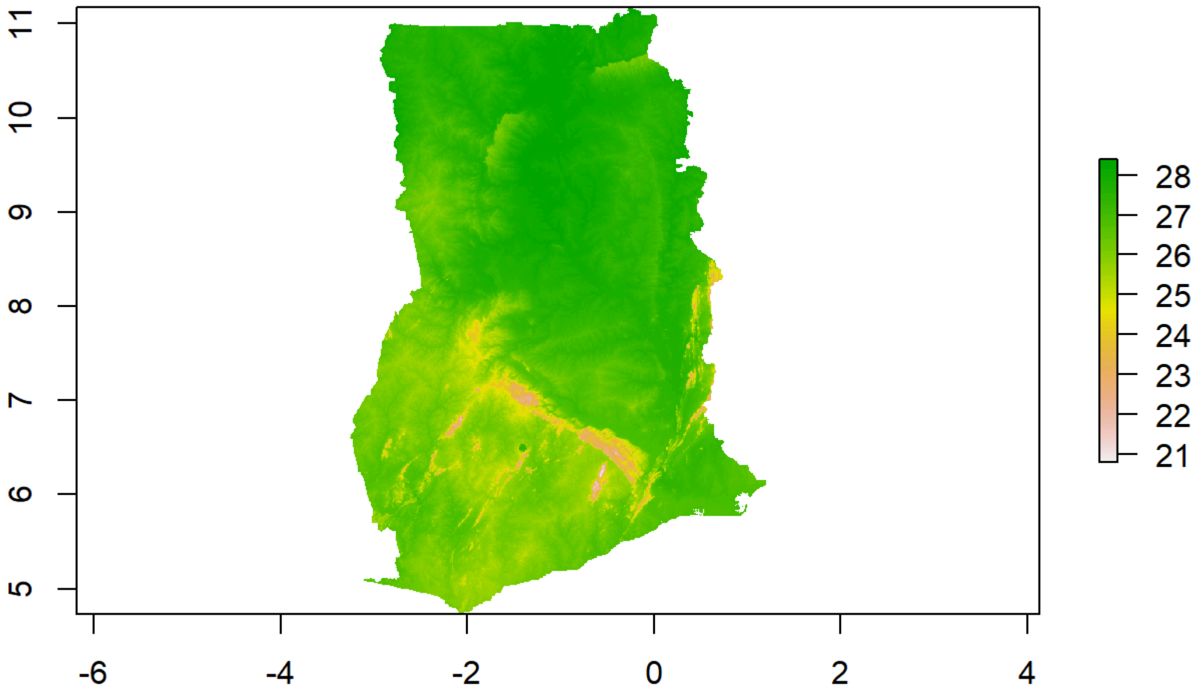



```
summary(Annual_Precipitation_2015)
```

```
## Warning in .local(object, ...): summary is an estimate based on a sample of 1e+05 cells (24.18% of all cells)
```

```
##         GHA_precip_1k
## Min.               63
## 1st Qu.            89
## Median            101
## 3rd Qu.           113
## Max.              177
## NA's           132036
```

```
save(Annual_Precipitation_2015,file="Annual_Precipitation_2015.RData")

str_name6<-'GHA_temp_1k.tif'
Mean_Temperature_2015=raster(str_name6)*0.1 # import raster file for temperature
plot(Mean_Temperature_2015)
```




```
summary(Mean_Temperature_2015)
```

```
##          GHA_temp_1k
## Min.            20.8
## 1st Qu.         26.2
## Median          27.0
## 3rd Qu.         27.7
## Max.            28.4
## NA's       131999.0
```

```
save(Mean_Temperature_2015,file="Mean_Temperature_2015.RData")


names(mal_sp)
```

```
##  [1] "DHSCLUST"                        "DHSID.x"
##  [3] "DHSCC"                           "DHSYEAR"
##  [5] "CCFIPS"                          "ADM1FIPS"
##  [7] "ADM1FIPSNA"                      "ADM1SALBNA"
##  [9] "ADM1SALBCO"                      "ADM1DHS"
## [11] "ADM1NAME"                        "DHSREGCO"
## [13] "DHSREGNA"                        "SOURCE"
## [15] "URBAN_RURA"                      "LATNUM"
## [17] "LONGNUM"                         "ALT_GPS"
## [19] "ALT_DEM"                         "DATUM"
## [21] "hv001"                           "total"
## [23] "mal_count"                       "mal_prev"
## [25] "DHSID.y"                         "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"          "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"              "lon"
## [33] "lat"
```

```
load("mal_sp.RData")

mal_df<-as.data.frame(mal_sp)
#summary(mal_df)
names(mal_df)
```

```
##  [1] "DHSCLUST"                         "DHSID.x"
##  [3] "DHSCC"                            "DHSYEAR"
##  [5] "CCFIPS"                           "ADM1FIPS"
##  [7] "ADM1FIPSNA"                       "ADM1SALBNA"
##  [9] "ADM1SALBCO"                       "ADM1DHS"
## [11] "ADM1NAME"                         "DHSREGCO"
## [13] "DHSREGNA"                         "SOURCE"
## [15] "URBAN_RURA"                       "LATNUM"
## [17] "LONGNUM"                          "ALT_GPS"
## [19] "ALT_DEM"                          "DATUM"
## [21] "hv001"                            "total"
## [23] "mal_count"                        "mal_prev"
## [25] "DHSID.y"                          "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"            "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"                "lon"
## [33] "lat"                              "coords.x1"
## [35] "coords.x2"                        "coords.x3"
```

```
mal_df<-mal_df[-(34:36)]
class(mal_df)
```

```
## [1] "data.frame"
```

```
class(mal_sp2$mal_count)
```

```
## [1] "numeric"
```

```
save(mal_df,file="mal_df.RData")
load("mal_df.RData")
dim(mal_df)
```

```
## [1] 192  33
```

```
## Model without spatial correlation
mod0<-glm(cbind(mal_count,total-mal_count)~ITN_Coverage_2015+Travel_Times_2015+
         Aridity_2015+Enhanced_Vegetation_Index_2015+Annual_Precipitation_2015,
         family = "binomial",data = mal_df)

save(mod0,file="mod0.RData")
summary(mod0)
```

```
##
## Call:
## glm(formula = cbind(mal_count, total - mal_count) ~ ITN_Coverage_2015 +
##     Travel_Times_2015 + Aridity_2015 + Enhanced_Vegetation_Index_2015 +
##     Annual_Precipitation_2015, family = "binomial", data = mal_df)
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   -4.963e+00  9.072e-01  -5.471 4.48e-08 ***
## ITN_Coverage_2015              3.423e+00  4.954e-01   6.910 4.83e-12 ***
## Travel_Times_2015              5.183e-03  9.698e-04   5.344 9.08e-08 ***
## Aridity_2015                  -2.044e-02  3.519e-02  -0.581 0.561265
## Enhanced_Vegetation_Index_2015  2.926e-04  8.066e-05   3.627 0.000286 ***
## Annual_Precipitation_2015      1.088e-02  1.753e-02   0.621 0.534793
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 734.88  on 191  degrees of freedom
## Residual deviance: 496.09  on 186  degrees of freedom
## AIC: 886.45
##
## Number of Fisher Scoring iterations: 4
```

```
exp(cbind(OR=coef(mod0),confint(mod0)))
```

```
## Waiting for profiling to be done...
```

```
##                                        OR        2.5 %      97.5 %
## (Intercept)                   0.006991533  0.001161857  0.04078408
## ITN_Coverage_2015            30.671986725 11.662717947 81.38667151
## Travel_Times_2015             1.005196434  1.003285777  1.00711005
## Aridity_2015                  0.979765678  0.914206455  1.04950584
## Enhanced_Vegetation_Index_2015  1.000292648  1.000135958  1.00045246
## Annual_Precipitation_2015     1.010941944  0.976784571  1.04632239
```

```
## Using INLA
library(INLA)

# Empty model
formula0 <- mal_count ~ 1

fit0 <- inla(formula0, family = "binomial", Ntrials = total,
control.family = list(link = "logit"),
data = mal_df,control.compute=list(dic=TRUE, cpo=TRUE, waic=TRUE))

summary(fit0)
```

```
## Time used:
##     Pre = 0.829, Running = 0.985, Post = 0.428, Total = 2.24
## Fixed effects:
##                mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -1.112 0.044     -1.199   -1.112     -1.025 -1.112   0
##
## Deviance Information Criterion (DIC) ...............: 1115.24
## Deviance Information Criterion (DIC, saturated) ....: 736.78
## Effective number of parameters .....................: 0.999
##
## Watanabe-Akaike information criterion (WAIC) ...: 1118.40
## Effective number of parameters .................: 4.13
##
## Marginal log-Likelihood:  -558.82
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
save(fit0,file="fit0.RData")

load("fit0.RData")
summary(fit0)
```

```
## Time used:
##     Pre = 0.829, Running = 0.985, Post = 0.428, Total = 2.24
## Fixed effects:
##                mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -1.112 0.044     -1.199   -1.112     -1.025 -1.112   0
##
## Deviance Information Criterion (DIC) ...............: 1115.24
## Deviance Information Criterion (DIC, saturated) ....: 736.78
## Effective number of parameters .....................: 0.999
##
## Watanabe-Akaike information criterion (WAIC) ...: 1118.40
## Effective number of parameters .................: 4.13
##
## Marginal log-Likelihood:  -558.82
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
formula0 <- mal_count~ ITN_Coverage_2015+Travel_Times_2015+
         Aridity_2015+Enhanced_Vegetation_Index_2015+
         Annual_Precipitation_2015+Mean_Temperature_2015

formula0 <- mal_count~ ITN_Coverage_2015+Travel_Times_2015+
         Aridity_2015


fit1 <- inla(formula0, family = "binomial", Ntrials = total,
control.family = list(link = "logit"),
data = mal_df,control.compute=list(dic=TRUE, cpo=TRUE, waic=TRUE))

summary(fit1)
```

```
## Time used:
##     Pre = 0.291, Running = 0.947, Post = 0.289, Total = 1.53
## Fixed effects:
##                     mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept)       -4.920 0.398     -5.700   -4.920     -4.140 -4.920   0
## ITN_Coverage_2015  4.047 0.455      3.154    4.047      4.940  4.047   0
## Travel_Times_2015  0.004 0.001      0.002    0.004      0.005  0.004   0
## Aridity_2015       0.048 0.010      0.027    0.048      0.068  0.048   0
##
## Deviance Information Criterion (DIC) ...............: 897.11
## Deviance Information Criterion (DIC, saturated) ....: 518.66
## Effective number of parameters .....................: 3.99
##
## Watanabe-Akaike information criterion (WAIC) ...: 905.09
## Effective number of parameters .................: 11.31
##
## Marginal log-Likelihood:  -469.64
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
save(fit1,file="fit1.RData")

load("fit1.RData")

summary(fit1,digits=4)
```

```
## Time used:
##     Pre = 0.2915, Running = 0.9468, Post = 0.2891, Total = 1.527
## Fixed effects:
##                       mean     sd 0.025quant 0.5quant 0.975quant    mode kld
## (Intercept)        -4.9200 0.3980    -5.7001  -4.9200    -4.1400 -4.9200   0
## ITN_Coverage_2015   4.0469 0.4554     3.1543   4.0469     4.9395  4.0469   0
## Travel_Times_2015   0.0037 0.0009     0.0020   0.0037     0.0054  0.0037   0
## Aridity_2015        0.0476 0.0105     0.0271   0.0476     0.0681  0.0476   0
##
## Deviance Information Criterion (DIC) ...............: 897.11
## Deviance Information Criterion (DIC, saturated) ....: 518.66
## Effective number of parameters .....................: 3.99
##
## Watanabe-Akaike information criterion (WAIC) ...: 905.09
## Effective number of parameters .................: 11.31
##
## Marginal log-Likelihood:  -469.64
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
####################   Spatial model in INLA  ######################

load("mal_df.RData")

coo <- cbind(mal_df$lon, mal_df$lat)
save(coo,file="coo.RData")

#mesh <- inla.mesh.2d(loc = coo,max.edge = c(0.3, 5), cutoff = 0.02)
#mesh$n

mesh <- inla.mesh.2d(loc = coo,max.edge = c(0.3, 5), cutoff = 0.01)
mesh$n
```

```
## [1] 1197
```

```
plot(mesh)

#coo <- cbind(d$long, d$lat)
#mesh <- inla.mesh.2d(
#  loc = coo, max.edge = c(0.4, 5),
#  cutoff = 0.01
#)


getwd()
```

```
## [1] "C:/Users/jmkaheto/Documents/Dropbox/Consult/Dr.Jaline Gerardin Northwestern Univ USA/AMMNet_Workshop_Ghan
a_Aheto"
```

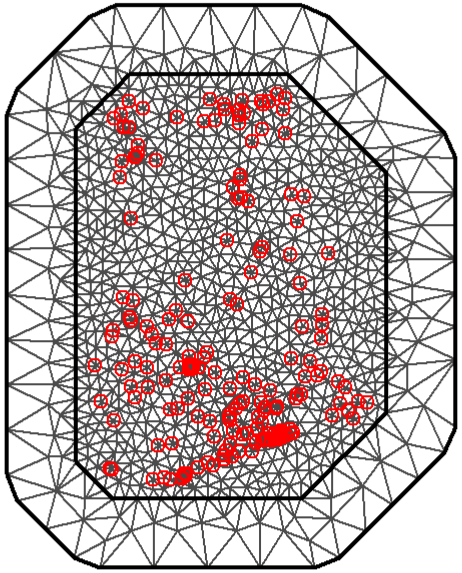
```
save(mesh,file="mesh.RData")

load("mesh.RData")

#The number of vertices is given by mesh$n and we can plot the mesh with plot(mesh).

names(mesh)
```

```
## [1] "meta"     "manifold" "n"        "loc"      "graph"    "segm"     "idx"
## [8] "crs"
```

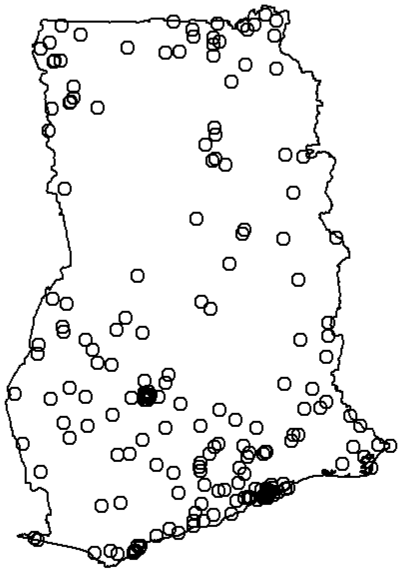
```
mesh$n
```

```
## [1] 1197
```

```
## [1] 669
plot(mesh)
points(coo, col = "red")
```



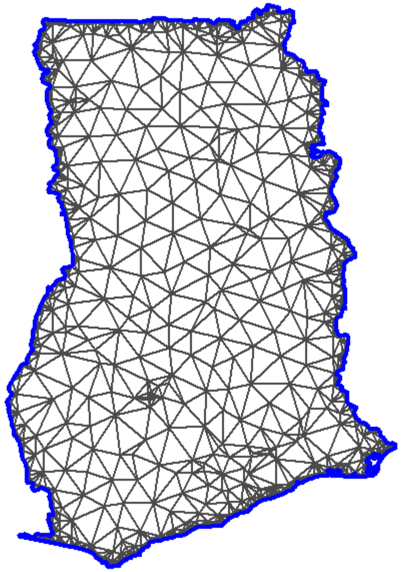
```
#   Alternatively, use the boundary of Nigeria (this is better so use it)
load("gh.border.RData")
gh_shp<-gh.border
plot(gh_shp)
points(mal_df$lon, mal_df$lat)
```



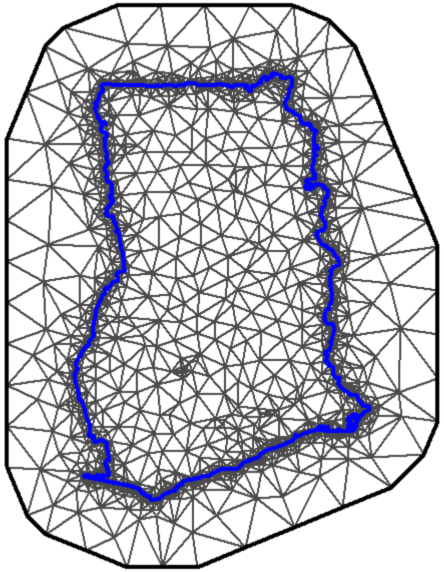
```
bdry <- inla.sp2segment(gh_shp)
names(bdry)
```

```
## [1] "loc"    "idx"    "grp"    "is.bnd" "crs"
```

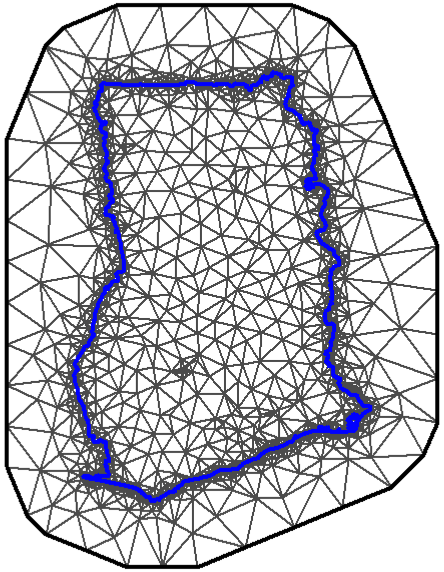
```
bdry$loc <- inla.mesh.map(bdry$loc)


mesh0 <- inla.mesh.2d(loc = coo, boundary = bdry, max.edge=c(0.5))
plot(mesh0)
```



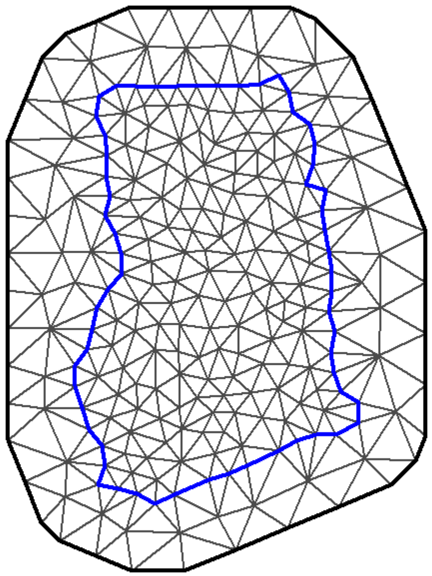
```
mesh1 <- inla.mesh.2d(loc = coo, boundary = bdry, max.edge=c(0.5, 1))
plot(mesh1)
```



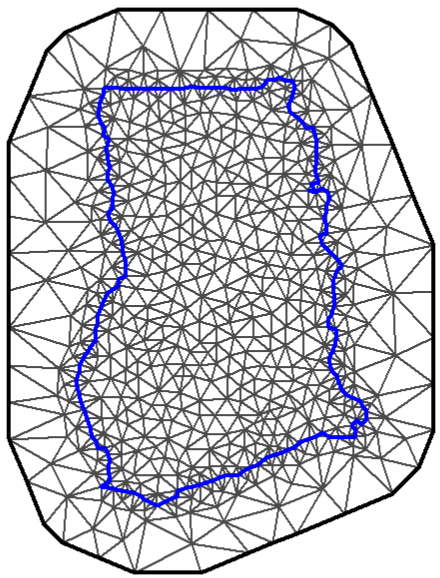
```
mesh2 <- inla.mesh.2d(loc = coo, boundary = bdry, max.edge=c(0.5, 1),
                      offset = c(0.5,1))
plot(mesh2)
```



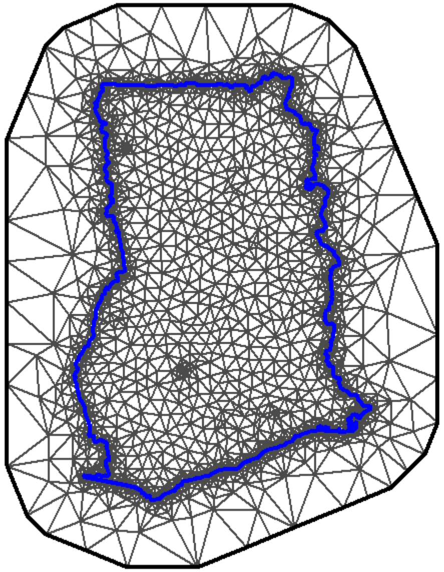
```
mesh3 <- inla.mesh.2d(loc = coo, boundary = bdry, max.edge=c(0.5,1),
                      offset = c(0.5, 1),cutoff = 0.3)
plot(mesh3)
```



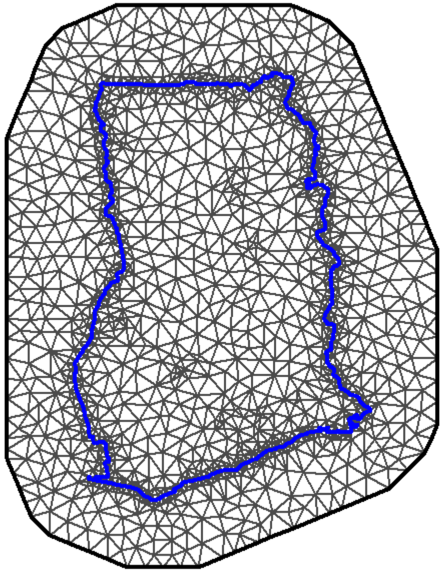
```
mesh4 <- inla.mesh.2d(loc = coo, boundary = bdry, max.edge=c(0.4,1),
                      offset = c(0.4, 1),cutoff = 0.1)
plot(mesh4)
```



```
mesh5 <- inla.mesh.2d(loc = coo,boundary = bdry, max.edge = c(0.3, 5), cutoff = 0.01)
plot(mesh5)
```



```
# Mesh 6
#max.edge = diff(range(coo))/(3*5)
max.edge<-max(diff(range(mal_df$lon)),diff(range(mal_df$lat)))/15

cutoff<-max.edge/10

mesh6 <- inla.mesh.2d(loc = coo, boundary = bdry,max.edge = max.edge,offset=c(0.5,1),cutoff=cutoff)
plot(mesh6)
```



```
png(file="mesh_results.png",pointsize=16,width = 650, height = 600)
par(mfrow=c(3,3))
plot(mesh0);plot(mesh1);plot(mesh2);plot(mesh3);plot(mesh4);plot(mesh5);plot(mesh6)

dev.off()
```

```
## png
##   2
```

```
##################################################################################
# Mesh creation notes
# SPDE Mesh Design

# max.edge: is the largest allowed triangle length; the lower the number the
#           higher the resolution
#
# offset: is defining how far you want to extend your domain
#         (i.e. a secondary boundary box)
#
# cutoff: can be used to avoid building too many small triangles around
#         clustered data locations
#
# ################################################################################



#3.3 Build the SPDE model on the mesh
#Then, we use the inla.spde2.matern() function to build the SPDE model.

#Building the SPDE model on the mesh
mesh<-mesh4



#3.3 Build the SPDE model on the mesh
#Then, we use the inla.spde2.matern() function to build the SPDE model.

#Building the SPDE model on the mesh
spde <- inla.spde2.matern(mesh = mesh, alpha = 2,constr = TRUE)
save(spde,file="spde.RData")

load("spde.RData")

lengths(spde)
```

```
##      model     n.spde    n.theta param.inla          f        BLC       mesh
##          1          1          1         15          6         18          8
```

```
indexs <- inla.spde.make.index("s", spde$n.spde)
lengths(indexs)
```

```
##       s s.group  s.repl
##     688     688     688
```

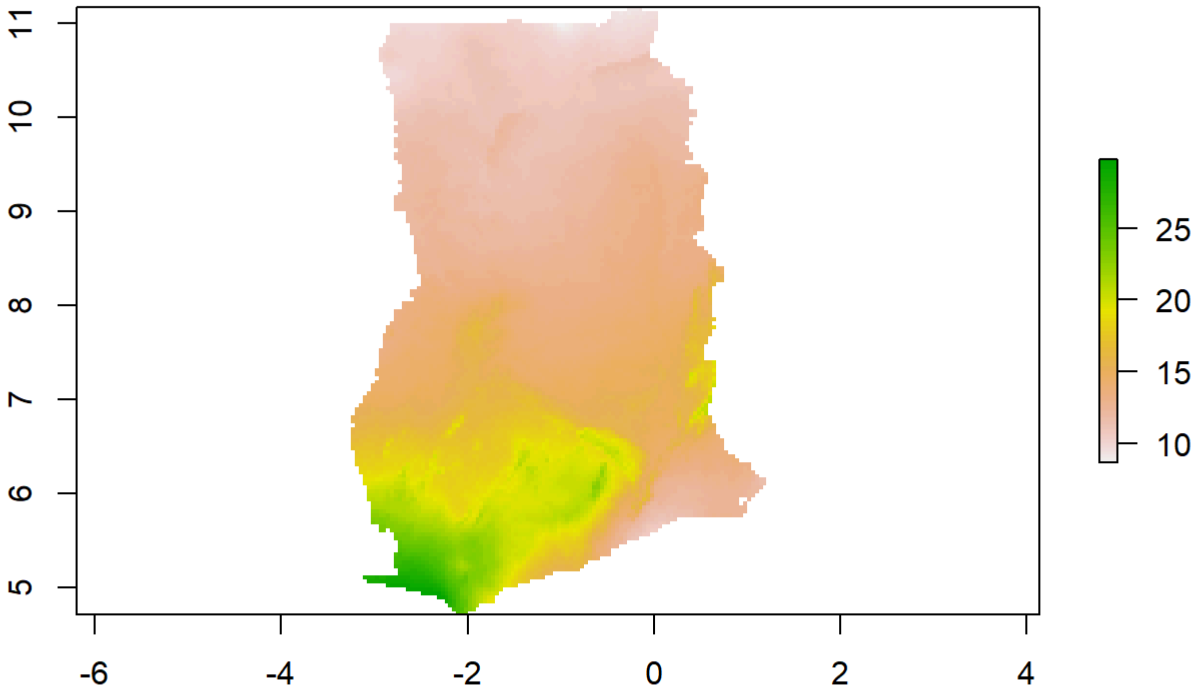
```
#    s    s.group  s.repl
#   1537    1537    1537

save(indexs,file="indexs.RData")
load("indexs.RData")



# Projector matrix
#We need to build a projector matrix A that projects the spatially continuous
#Gaussian random field at the mesh nodes. The projector matrix A can be built
#with the inla.spde.make.A() function passing the mesh and the coordinates.

A <- inla.spde.make.A(mesh = mesh, loc = coo)
save(A,file="A.RData")
load("A.RData")

# Prediction data
# Here we specify the locations where we want to predict the prevalence. We set
# the prediction locations to the locations of the the covariate raster data.
# We can get the coordinates of the raster r with the function rasterToPoints()
# of the raster package. This function returns a matrix with the coordinates and
# values of the raster that do not have NA values. We see there are 12964 points.



library(raster)
#r <- getData(name = "alt", country = "GHA", mask = TRUE)
#plot(r)



#r<- getData(name = "alt", country = "GHA", mask = TRUE)



res(Aridity_2015)# resolution = 0.008333333
```

```
## [1] 0.008333333 0.008333333
```

```
# In this example, we want to use fewer prediction points. We can lower the
# resolution of the raster by using aggregate() of the  raster package.
# The arguments of the function are:

ra <- aggregate(Aridity_2015, fact = 5, fun = mean)
dim(ra)
```

```
## [1] 155 107   1
```

```
plot(ra)
```




```
res(ra)
```

```
## [1] 0.04166667 0.04166667
```

```
save(ra,file="ra.RData")

load("ra.RData")

dp <- rasterToPoints(ra)
dim(dp)
```

```
## [1] 11552     3
```

```
#r<- getData(name = "alt", country = "GHA", mask = TRUE)
#class(r)

dp<-as.data.frame(dp)
names(dp)
```

```
## [1] "x"                    "y"                    "GHA_CGIAR_AI_1k_final"
```

```
names(mal_df)
```

```
##  [1] "DHSCLUST"                        "DHSID.x"
##  [3] "DHSCC"                           "DHSYEAR"
##  [5] "CCFIPS"                          "ADM1FIPS"
##  [7] "ADM1FIPSNA"                      "ADM1SALBNA"
##  [9] "ADM1SALBCO"                      "ADM1DHS"
## [11] "ADM1NAME"                        "DHSREGCO"
## [13] "DHSREGNA"                        "SOURCE"
## [15] "URBAN_RURA"                      "LATNUM"
## [17] "LONGNUM"                         "ALT_GPS"
## [19] "ALT_DEM"                         "DATUM"
## [21] "hv001"                           "total"
## [23] "mal_count"                       "mal_prev"
## [25] "DHSID.y"                         "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"           "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"               "lon"
## [33] "lat"
```

```
summary(dp)
```

```
##        x                 y          GHA_CGIAR_AI_1k_final
##  Min.   :-3.2375   Min.   : 4.737   Min.   : 8.648
##  1st Qu.:-2.0708   1st Qu.: 6.487   1st Qu.:11.843
##  Median :-1.2375   Median : 7.862   Median :13.566
##  Mean   :-1.2073   Mean   : 7.964   Mean   :14.555
##  3rd Qu.:-0.3625   3rd Qu.: 9.404   3rd Qu.:16.317
##  Max.   : 1.1792   Max.   :11.154   Max.   :29.820
```

```
colnames(dp)<-c("lon","lat","Aridity_2015")
names(dp)
```

```
## [1] "lon"          "lat"          "Aridity_2015"
```

```
#dp$Aridity_2015 <- extract(Aridity_MEAN, d[, c("long", "lat")])

dp$ITN_Coverage_2015 <- extract(ITN_Coverage_2015, dp[, c("lon", "lat")])
dp$Mean_Temperature_2015 <- extract(Mean_Temperature_2015, dp[, c("lon", "lat")])
dp$Annual_Precipitation_2015 <- extract(Annual_Precipitation_2015, dp[, c("lon", "lat")])
dp$Travel_Times_2015 <- extract(Travel_Times_2015, dp[, c("lon", "lat")])
dp$Enhanced_Vegetation_Index_2015 <- extract(Enhanced_Vegetation_Index_2015, dp[, c("lon", "lat")])

summary(dp)
```

```
##       lon                lat          Aridity_2015    ITN_Coverage_2015
##  Min.   :-3.2375   Min.   : 4.737   Min.   : 8.648   Min.   :0.0000
##  1st Qu.:-2.0708   1st Qu.: 6.487   1st Qu.:11.843   1st Qu.:0.5294
##  Median :-1.2375   Median : 7.862   Median :13.566   Median :0.5928
##  Mean   :-1.2073   Mean   : 7.964   Mean   :14.555   Mean   :0.5909
##  3rd Qu.:-0.3625   3rd Qu.: 9.404   3rd Qu.:16.317   3rd Qu.:0.6754
##  Max.   : 1.1792   Max.   :11.154   Max.   :29.820   Max.   :0.7868
##                                                      NA's   :17
##  Mean_Temperature_2015 Annual_Precipitation_2015 Travel_Times_2015
##  Min.   :21.30         Min.   : 65.0             Min.   :  0.00
##  1st Qu.:26.20         1st Qu.: 89.0             1st Qu.: 19.70
##  Median :27.00         Median :101.0             Median : 36.40
##  Mean   :26.89         Mean   :103.4             Mean   : 43.81
##  3rd Qu.:27.70         3rd Qu.:113.0             3rd Qu.: 60.00
##  Max.   :28.40         Max.   :177.0             Max.   :241.20
##  NA's   :290           NA's   :290               NA's   :290
##  Enhanced_Vegetation_Index_2015
##  Min.   : 0.000
##  1st Qu.: 0.000
##  Median : 0.000
##  Mean   : 1.017
##  3rd Qu.: 0.000
##  Max.   :63.000
##  NA's   :290
```

```
#dp$ALT_DEM <- extract(ALT_DEM, dp[, c("x", "y")])

dim(dp)
```

```
## [1] 11552     8
```

```
class(dp)
```

```
## [1] "data.frame"
```

```
names(dp)
```

```
## [1] "lon"                            "lat"
## [3] "Aridity_2015"                   "ITN_Coverage_2015"
## [5] "Mean_Temperature_2015"          "Annual_Precipitation_2015"
## [7] "Travel_Times_2015"              "Enhanced_Vegetation_Index_2015"
```

```
summary(dp)
```

```
##       lon               lat          Aridity_2015    ITN_Coverage_2015
##  Min.   :-3.2375   Min.   : 4.737   Min.   : 8.648   Min.   :0.0000
##  1st Qu.:-2.0708   1st Qu.: 6.487   1st Qu.:11.843   1st Qu.:0.5294
##  Median :-1.2375   Median : 7.862   Median :13.566   Median :0.5928
##  Mean   :-1.2073   Mean   : 7.964   Mean   :14.555   Mean   :0.5909
##  3rd Qu.:-0.3625   3rd Qu.: 9.404   3rd Qu.:16.317   3rd Qu.:0.6754
##  Max.   : 1.1792   Max.   :11.154   Max.   :29.820   Max.   :0.7868
##                                                      NA's   :17
##  Mean_Temperature_2015 Annual_Precipitation_2015 Travel_Times_2015
##  Min.   :21.30         Min.   : 65.0             Min.   :  0.00
##  1st Qu.:26.20         1st Qu.: 89.0             1st Qu.: 19.70
##  Median :27.00         Median :101.0             Median : 36.40
##  Mean   :26.89         Mean   :103.4             Mean   : 43.81
##  3rd Qu.:27.70         3rd Qu.:113.0             3rd Qu.: 60.00
##  Max.   :28.40         Max.   :177.0             Max.   :241.20
##  NA's   :290           NA's   :290               NA's   :290
##  Enhanced_Vegetation_Index_2015
##  Min.   : 0.000
##  1st Qu.: 0.000
##  Median : 0.000
##  Mean   : 1.017
##  3rd Qu.: 0.000
##  Max.   :63.000
##  NA's   :290
```

```
dp<-na.omit(dp)
dim(dp)
```

```
## [1] 11262     8
```

```
save(dp,file="dp.RData")
load("dp.RData")

#library(mice)

#tempData <- mice(dp,m=5,maxit=50,meth='pmm',seed=500)#
#summary(tempData)

names(dp)
```

```
## [1] "lon"                          "lat"
## [3] "Aridity_2015"                 "ITN_Coverage_2015"
## [5] "Mean_Temperature_2015"        "Annual_Precipitation_2015"
## [7] "Travel_Times_2015"            "Enhanced_Vegetation_Index_2015"
```

```
coop <- as.matrix(dp[, c("lon", "lat")])
dim(coop)
```

```
## [1] 11262     2
```

```
save(coop,file="coop.RData")

load("coop.RData")

# Projector matrix
# We also construct the matrix that projects the locations where we will do the
# predictions.

Ap <- inla.spde.make.A(mesh = mesh, loc = coop)
save(Ap,file="Ap.RData")
load("Ap.RData")

names(mal_df)
```

```
##  [1] "DHSCLUST"                         "DHSID.x"
##  [3] "DHSCC"                            "DHSYEAR"
##  [5] "CCFIPS"                           "ADM1FIPS"
##  [7] "ADM1FIPSNA"                       "ADM1SALBNA"
##  [9] "ADM1SALBCO"                       "ADM1DHS"
## [11] "ADM1NAME"                         "DHSREGCO"
## [13] "DHSREGNA"                         "SOURCE"
## [15] "URBAN_RURA"                       "LATNUM"
## [17] "LONGNUM"                          "ALT_GPS"
## [19] "ALT_DEM"                          "DATUM"
## [21] "hv001"                            "total"
## [23] "mal_count"                        "mal_prev"
## [25] "DHSID.y"                          "Aridity_2015"
## [27] "Enhanced_Vegetation_Index_2015" "ITN_Coverage_2015"
## [29] "Mean_Temperature_2015"            "Annual_Precipitation_2015"
## [31] "Travel_Times_2015"                "lon"
## [33] "lat"
```

```
stk.e <- inla.stack(tag = "est",
data = list(y = mal_df$mal_count, numtrials = mal_df$total),
A = list(1, A),
effects = list(data.frame(b0 = 1,mal_df[,26:31]), s = indexs))

save(stk.e,file="stk.e.RData")
load("stk.e.RData")

#stack for prediction stk.p
names(dp)
```

```
## [1] "lon"                            "lat"
## [3] "Aridity_2015"                   "ITN_Coverage_2015"
## [5] "Mean_Temperature_2015"          "Annual_Precipitation_2015"
## [7] "Travel_Times_2015"              "Enhanced_Vegetation_Index_2015"
```

```
stk.p <- inla.stack(tag = "pred",
data = list(y = NA, numtrials = NA),
A = list(1, Ap),
effects = list(data.frame(b0 = 1,dp[,3:8]), s = indexs))

save(stk.p,file="stk.p.RData")
load("stk.p.RData")



#stk.full has stk.e and stk.p
stk.full <- inla.stack(stk.e, stk.p)
save(stk.full,file="stk.full.RData")

load("stk.full.RData")



## Bayesian model without covariates: spatial



formula0 <- y ~ 0 + b0+
         f(s, model = spde)



start<-Sys.time()# Save start time

fit00 <- inla(formula0, family = "binomial", Ntrials = numtrials,
control.family = list(link = "logit"),
data = inla.stack.data(stk.full),
control.compute=list(dic=TRUE, cpo=TRUE, waic=TRUE),
control.predictor = list(compute = TRUE, link = 1, A = inla.stack.A(stk.full)))

end<-Sys.time()# Save end time
end-start # time taken to run the model
```

```
## Time difference of 5.440297 secs
```

```
summary(fit00)
```

```
## Time used:
##     Pre = 1.43, Running = 3.62, Post = 0.268, Total = 5.31
## Fixed effects:
##      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## b0 -1.164 0.132     -1.428   -1.163     -0.908 -1.163   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -3.57 0.324      -4.22    -3.57      -2.94 -3.56
## Theta2 for s  2.13 0.269       1.61     2.13       2.66  2.12
##
## Deviance Information Criterion (DIC) ...............: 708.88
## Deviance Information Criterion (DIC, saturated) ....: 330.42
## Effective number of parameters .....................: 95.15
##
## Watanabe-Akaike information criterion (WAIC) ...: 712.33
## Effective number of parameters .................: 74.42
##
## Marginal log-Likelihood:  -409.87
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
save(fit00,file="fit00.RData")

load("fit00.RData")
summary(fit00)
```

```
## Time used:
##     Pre = 1.43, Running = 3.62, Post = 0.268, Total = 5.31
## Fixed effects:
##      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## b0 -1.164 0.132     -1.428   -1.163     -0.908 -1.163   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                 mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -3.57 0.324      -4.22    -3.57      -2.94 -3.56
## Theta2 for s  2.13 0.269       1.61     2.13       2.66  2.12
##
## Deviance Information Criterion (DIC) ...............: 708.88
## Deviance Information Criterion (DIC, saturated) ....: 330.42
## Effective number of parameters .....................: 95.15
##
## Watanabe-Akaike information criterion (WAIC) ...: 712.33
## Effective number of parameters .................: 74.42
##
## Marginal log-Likelihood:  -409.87
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
# Compare empty model without
summary(fit0) # WAIC= 1118.41
```

```
## Time used:
##     Pre = 0.829, Running = 0.985, Post = 0.428, Total = 2.24
## Fixed effects:
##                mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -1.112 0.044     -1.199   -1.112     -1.025 -1.112   0
##
## Deviance Information Criterion (DIC) ...............: 1115.24
## Deviance Information Criterion (DIC, saturated) ....: 736.78
## Effective number of parameters .....................: 0.999
##
## Watanabe-Akaike information criterion (WAIC) ...: 1118.40
## Effective number of parameters .................: 4.13
##
## Marginal log-Likelihood:  -558.82
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
summary(fit00) # WAIC= 698.39
```

```
## Time used:
##     Pre = 1.43, Running = 3.62, Post = 0.268, Total = 5.31
## Fixed effects:
##      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## b0 -1.164 0.132     -1.428   -1.163     -0.908 -1.163   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -3.57 0.324      -4.22    -3.57      -2.94 -3.56
## Theta2 for s  2.13 0.269       1.61     2.13       2.66  2.12
##
## Deviance Information Criterion (DIC) ...............: 708.88
## Deviance Information Criterion (DIC, saturated) ....: 330.42
## Effective number of parameters .....................: 95.15
##
## Watanabe-Akaike information criterion (WAIC) ...: 712.33
## Effective number of parameters .................: 74.42
##
## Marginal log-Likelihood:  -409.87
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
fit00$dic$dic
```

```
## [1] 708.8817
```

```
fit00$waic$waic
```

```
## [1] 712.3322
```

```
####################  Compare models #########################
summary(fit0) # WAIC= 1118.41 # Bayesian non-spatial model without covarites
```

```
## Time used:
##     Pre = 0.829, Running = 0.985, Post = 0.428, Total = 2.24
## Fixed effects:
##                mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -1.112 0.044     -1.199   -1.112     -1.025 -1.112   0
##
## Deviance Information Criterion (DIC) ...............: 1115.24
## Deviance Information Criterion (DIC, saturated) ....: 736.78
## Effective number of parameters .....................: 0.999
##
## Watanabe-Akaike information criterion (WAIC) ...: 1118.40
## Effective number of parameters .................: 4.13
##
## Marginal log-Likelihood:  -558.82
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
summary(fit1) # WAIC= 898.99 # Bayesian non-spatial model with covariates
```

```
## Time used:
##     Pre = 0.291, Running = 0.947, Post = 0.289, Total = 1.53
## Fixed effects:
##                      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept)        -4.920 0.398     -5.700   -4.920     -4.140 -4.920   0
## ITN_Coverage_2015  4.047 0.455      3.154    4.047      4.940  4.047   0
## Travel_Times_2015  0.004 0.001      0.002    0.004      0.005  0.004   0
## Aridity_2015       0.048 0.010      0.027    0.048      0.068  0.048   0
##
## Deviance Information Criterion (DIC) ...............: 897.11
## Deviance Information Criterion (DIC, saturated) ....: 518.66
## Effective number of parameters .....................: 3.99
##
## Watanabe-Akaike information criterion (WAIC) ...: 905.09
## Effective number of parameters .................: 11.31
##
## Marginal log-Likelihood:  -469.64
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
summary(fit00) # WAIC= 699.08 # Bayesian empty spatial model
```

```
## Time used:
##     Pre = 1.43, Running = 3.62, Post = 0.268, Total = 5.31
## Fixed effects:
##      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## b0 -1.164 0.132     -1.428   -1.163     -0.908 -1.163   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                 mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -3.57 0.324      -4.22    -3.57      -2.94 -3.56
## Theta2 for s  2.13 0.269       1.61     2.13       2.66  2.12
##
## Deviance Information Criterion (DIC) ...............: 708.88
## Deviance Information Criterion (DIC, saturated) ....: 330.42
## Effective number of parameters .....................: 95.15
##
## Watanabe-Akaike information criterion (WAIC) ...: 712.33
## Effective number of parameters .................: 74.42
##
## Marginal log-Likelihood:  -409.87
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
#summary(fit2) # WAIC= 699.08 # Bayesian spatial model with covariates



#fit00$summary.random
#fit00$summary.hyperpar

plot(fit00, plot.prior = TRUE)

#fit00$summary.fitted.values


index <- inla.stack.index(stack = stk.full, tag = "pred")$data
save(index,file="index.RData")
load("index.RData")


######## Estimate and convert the estimates to percentages #########

prev_mean0 <- (fit00$summary.fitted.values[index, "mean"])*100
summary(prev_mean0)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.082  23.812  27.200  27.979  31.599  67.943
```

```
prev_l0 <- (fit00$summary.fitted.values[index, "0.025quant"])*100
prev_u0 <- (fit00$summary.fitted.values[index, "0.975quant"])*100

sd0 <- (fit00$summary.fitted.values[index, "sd"])*100# standard deviation/error
summary(sd0)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.288  10.010  14.238  13.211  16.203  22.134
```

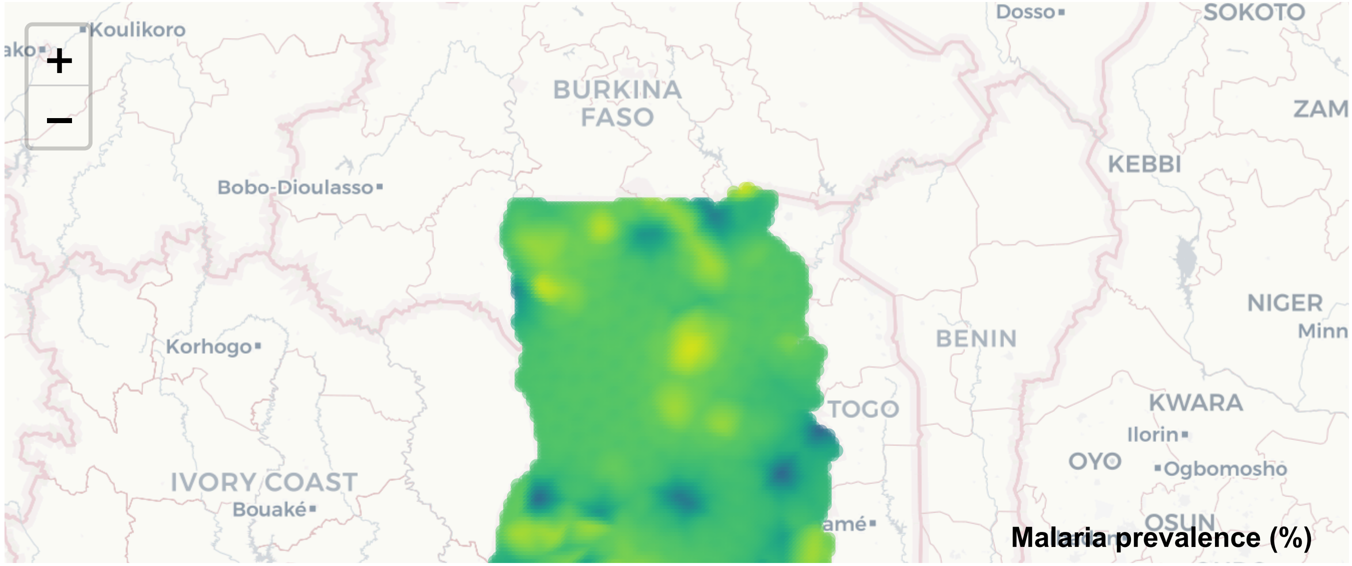
```
prev_mean0 <- (fit00$summary.fitted.values[index, "mean"])*100
summary(prev_mean0)
```

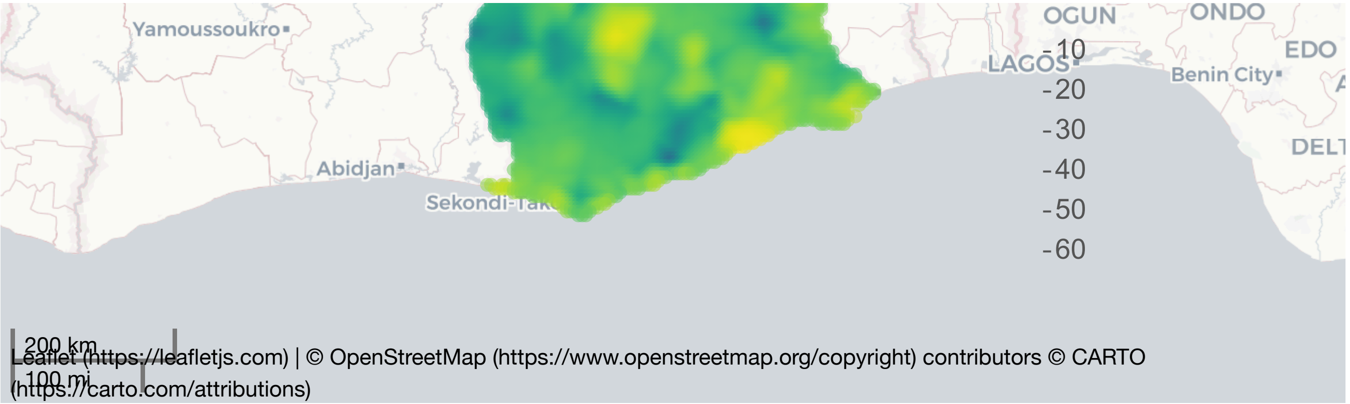
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.082  23.812  27.200  27.979  31.599  67.943
```

```
save(prev_mean0,file="prev_mean0.RData")
load("prev_mean0.RData")
summary(prev_mean0)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.082  23.812  27.200  27.979  31.599  67.943
```

```
library(leaflet)


################ Map the predicted prevalence (%) ##############

# Using leaflet
pal <- colorNumeric("viridis", c(0, 100),reverse=T, na.color = "transparent")

#pal <- colorBin("RdYlBu", bins = c(0, 25, 50, 75, 100),reverse=T)
#pal <- colorBin("RdYlBu", c(0, 100), na.color = "transparent",reverse=T)

leaflet() %>%
addProviderTiles(providers$CartoDB.Positron) %>%
addCircles(
lng = coop[, 1], lat = coop[, 2],
color = pal(prev_mean0)
) %>%
addLegend("bottomright",
pal = pal, values = prev_mean0,
title = "Malaria prevalence (%)"
) %>%
addScaleBar(position = c("bottomleft"))
```

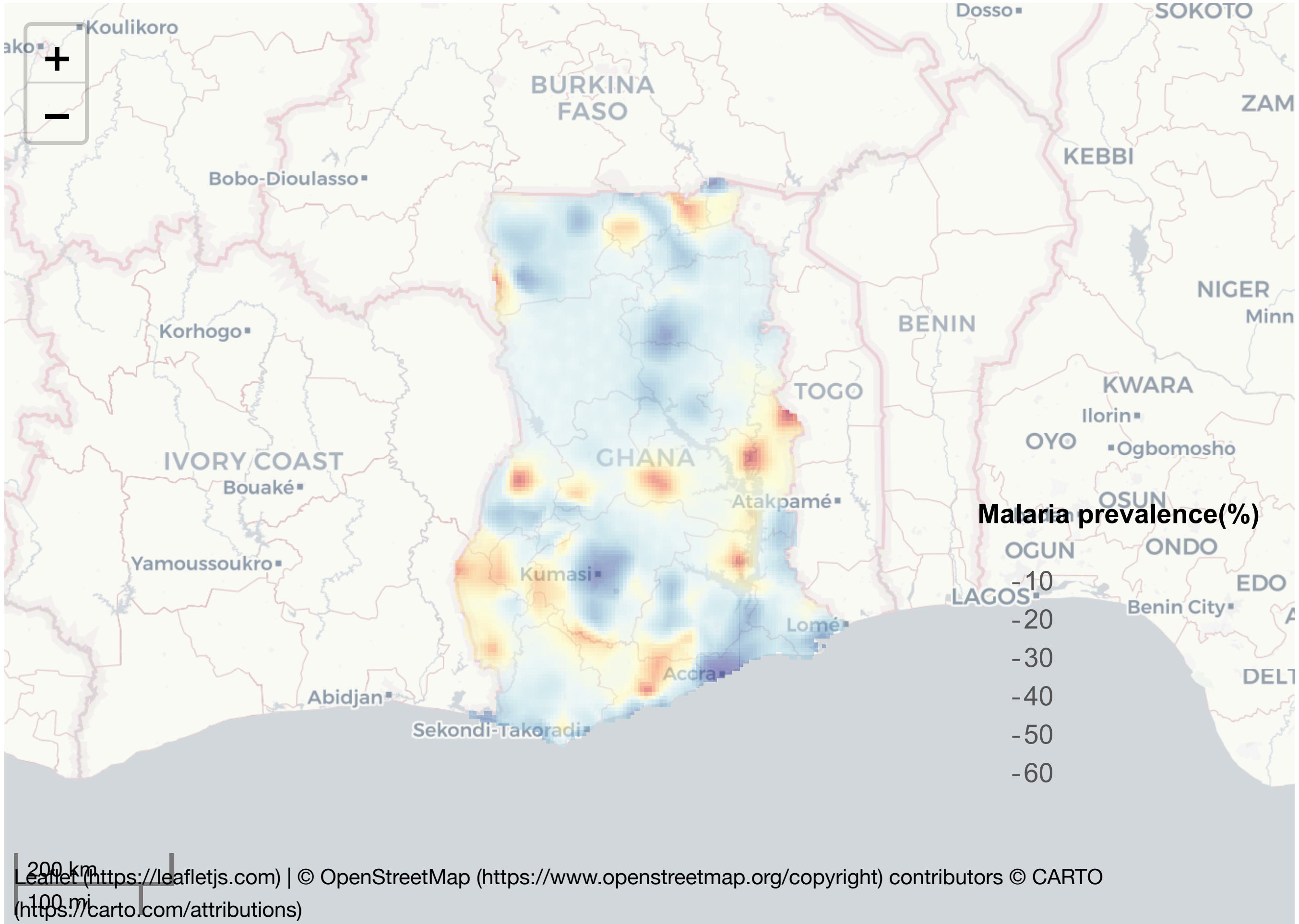




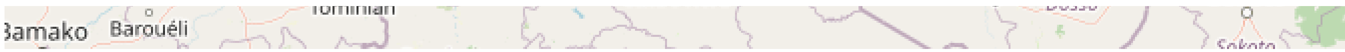

```
r_prev_mean0 <- rasterize(x = coop, y = ra, field = prev_mean0, fun = mean)

summary(r_prev_mean0)
```

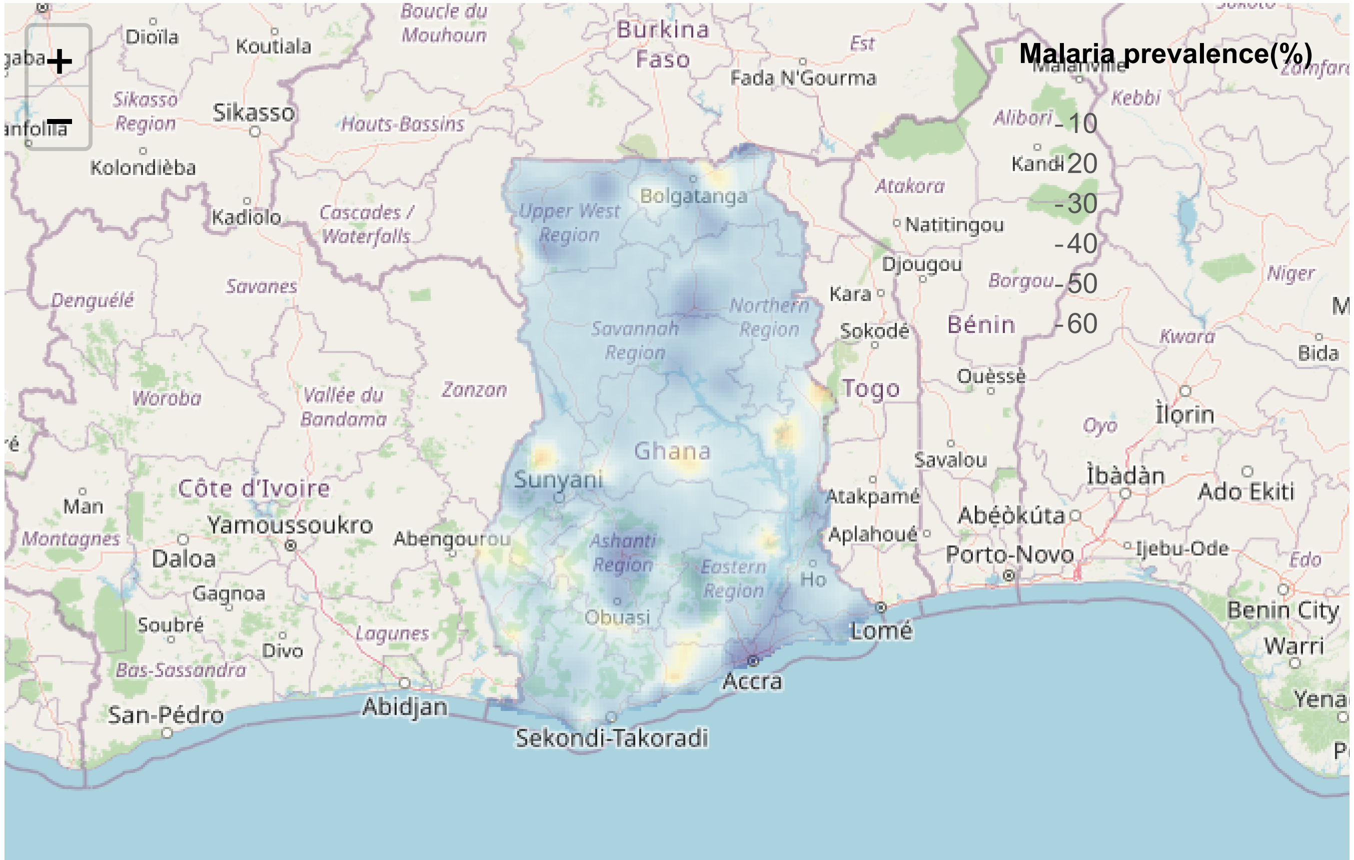
```
##               layer
## Min.       2.082355
## 1st Qu.   23.812237
## Median    27.200235
## 3rd Qu.   31.599109
## Max.      67.943218
## NA's    5323.000000
```

```
# Write raster files and save as GeoTIFF file
writeRaster(r_prev_mean0, "r_prev_mean0.tif", filetype = "GTiff", overwrite = TRUE)

# Mapping predicted malaria prevalence
pal <- colorNumeric("RdYlBu", values(r_prev_mean0),reverse=T, na.color = "transparent")

prev0_map<-leaflet() %>% addProviderTiles(providers$CartoDB.Positron) %>%
  addRasterImage(r_prev_mean0, colors = pal, opacity = 0.5) %>%
  addLegend("bottomright", pal = pal, values = values(r_prev_mean0), title = "Malaria prevalence(%)") %>%
  addScaleBar(position = c("bottomleft"))

prev0_map
```




```
save(prev0_map,file="prev0_map.RData")

# Another short form of mapping raster files
pal2 <- colorNumeric("RdYlBu", c(0, 100), reverse=T, na.color = "transparent")

leaflet() %>% addTiles() %>%
  addRasterImage(r_prev_mean0, colors = pal2, opacity = 0.5) %>%
  addLegend(pal = pal2, values = values(r_prev_mean0),
            title = "Malaria prevalence(%)")
```

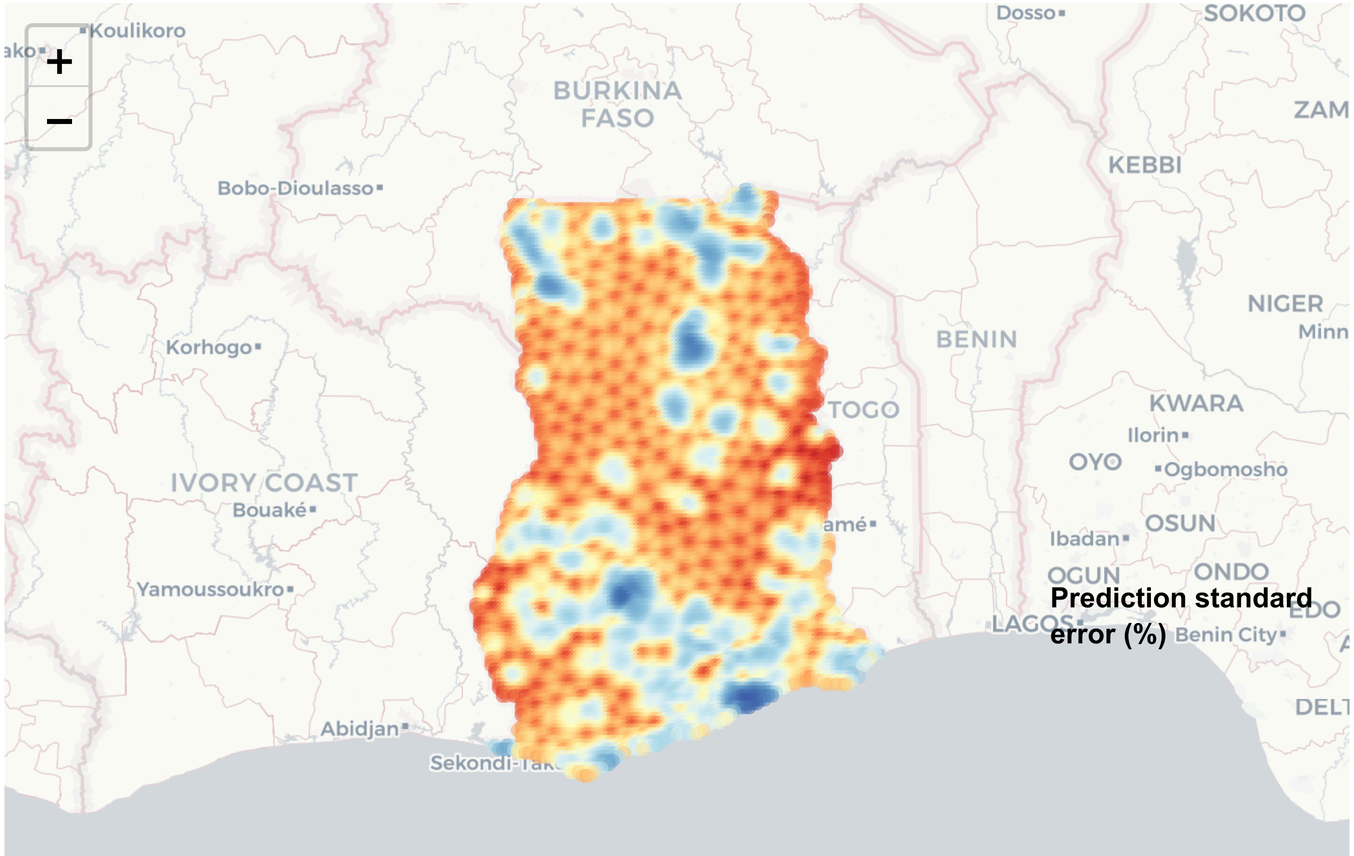


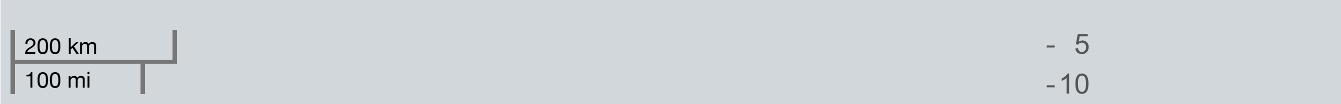


Leaflet (https://leafletjs.com) | © OpenStreetMap (https://openstreetmap.org/copyright/), ODbL (https://opendatacommons.org/licenses/odbl/)

Map the prediction standard error in leaflet

```
#pal <- colorNumeric("viridis", c(0, 100), na.color = "transparent")
pal <- colorNumeric("RdYlBu", c(0,23),reverse=T, na.color = "transparent")

leaflet() %>%
addProviderTiles(providers$CartoDB.Positron) %>%
addCircles(
lng = coop[, 1], lat = coop[, 2],
color = pal(sd0)
) %>%
addLegend("bottomright",
pal = pal, values = sd0,
title = "Prediction standard<br> error (%)"
) %>%
addScaleBar(position = c("bottomleft"))
```

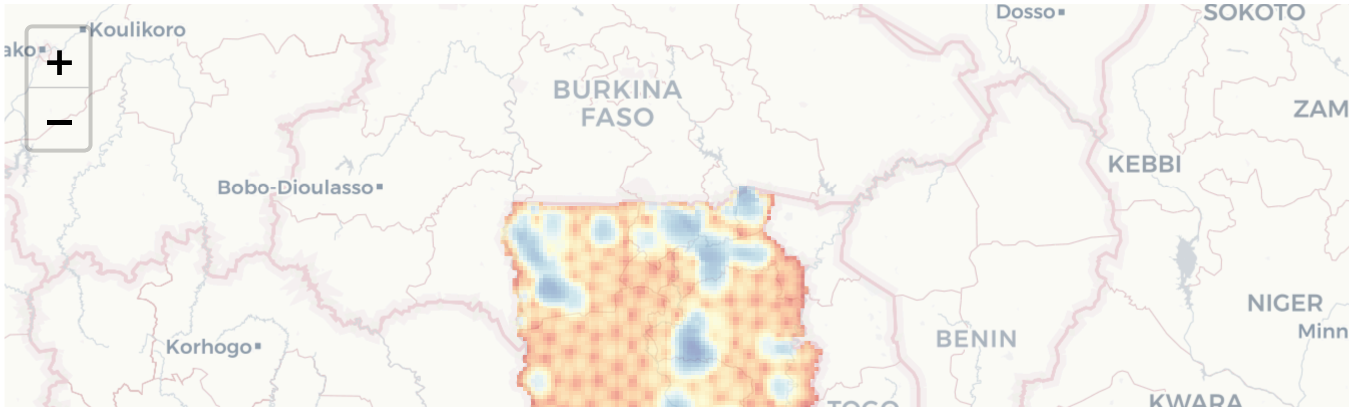


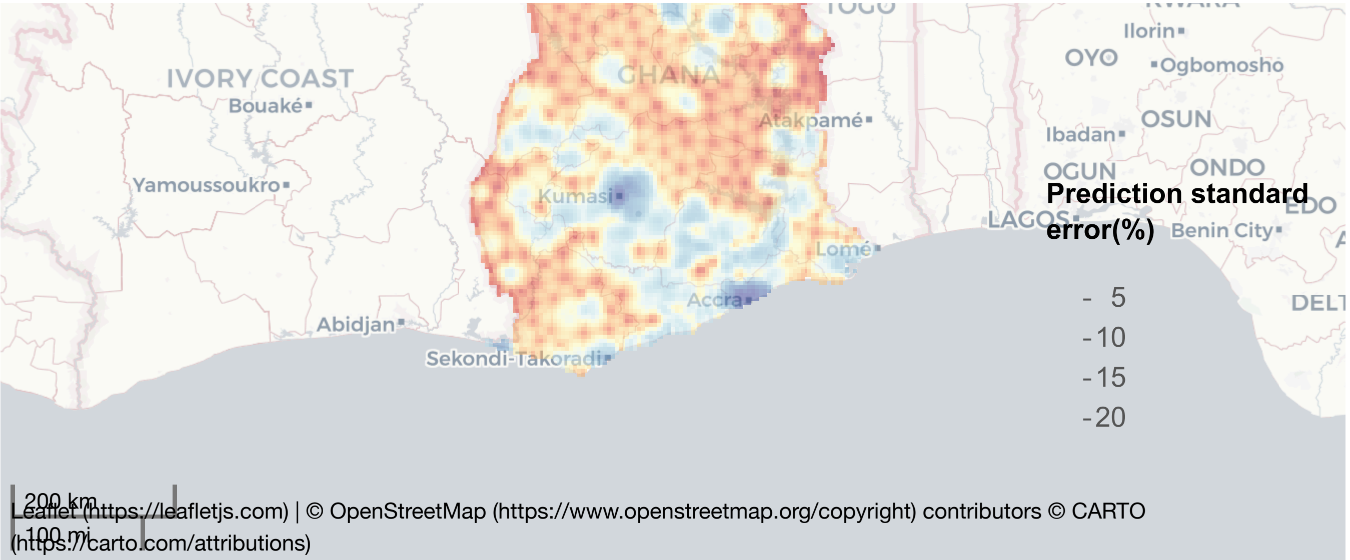



```
##########   Mapping prediction standard errors (%)   #############

sd0 <- (fit00$summary.fitted.values[index, "sd"])*100# standard deviation/error
summary(sd0)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.288  10.010  14.238  13.211  16.203  22.134
```

```
r_sd0 <- rasterize(x = coop, y = ra, field = sd0, fun = mean)

# Write raster files and save as GeoTIFF file
writeRaster(r_sd0, "r_sd0.tif", filetype = "GTiff", overwrite = TRUE)

pal <- colorNumeric("RdYlBu", values(r_sd0),reverse=T, na.color = "transparent")

sd0_map<-leaflet() %>% addProviderTiles(providers$CartoDB.Positron) %>%
  addRasterImage(r_sd0, colors = pal, opacity = 0.5) %>%
  addLegend("bottomright", pal = pal, values = values(r_sd0), title = "Prediction standard<br> error(%)") %>%
  addScaleBar(position = c("bottomleft"))

sd0_map
```





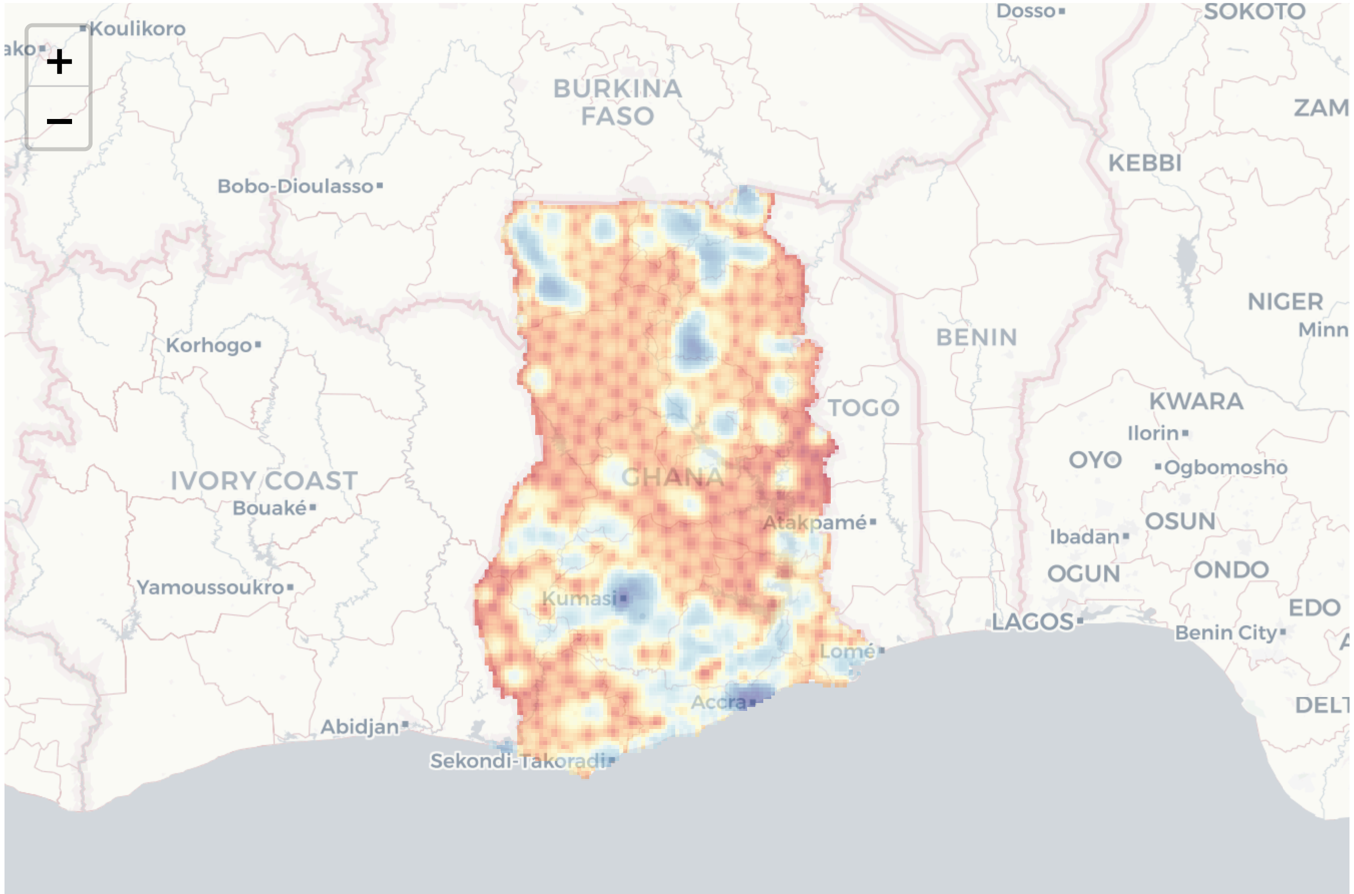

```
save(sd0_map,file="sd0_map.RData")
```

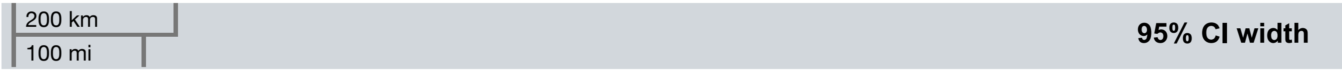
Mapping 95% ci width

```
prev_l0 <- (fit00$summary.fitted.values[index, "0.025quant"])*100
prev_u0 <- (fit00$summary.fitted.values[index, "0.975quant"])*100

wi0<-prev_u0-prev_l0
summary(wi0)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.889  38.581  54.108  49.985  60.961  79.039
```

```
r_wi0 <- rasterize(x = coop, y = ra, field = wi0, fun = mean)

# Write raster files and save as GeoTIFF file
writeRaster(r_wi0, "r_wi0.tif", filetype = "GTiff", overwrite = TRUE)

pal <- colorNumeric("RdYlBu", values(r_wi0),reverse=T, na.color = "transparent")

wi0_map<-leaflet() %>% addProviderTiles(providers$CartoDB.Positron) %>%
  addRasterImage(r_wi0, colors = pal, opacity = 0.5) %>%
  addLegend("bottomright", pal = pal, values = values(r_wi0), title = "95% CI width") %>%
  addScaleBar(position = c("bottomleft"))

wi0_map
```






```
save(wi0_map,file="wi0_map.RData")
```

Extracting the variance and other posterior estimates

```
# Back to extracting our random term variance now.

inla.emarginal(function(x) 1/x, fit00$marginals.hyper[[1]])
```

```
## [1] -0.2822867
```

```
# In order to extract the relevant information on the spatial field we will need to use the inla.spde2.result() f
unction

Mod_p1.field <- inla.spde2.result(inla = fit00,
                                 name = "s", spde = spde,
                                 do.transf = T)     # This will transform the results back from the internal mode
l scale


names(Mod_p1.field) # check the component of Mod_p1.field
```

```
##  [1] "summary.values"                "marginals.values"
##  [3] "summary.hyperpar"              "summary.theta"
##  [5] "summary.log.tau"               "summary.log.kappa"
##  [7] "summary.log.variance.nominal"  "summary.log.range.nominal"
##  [9] "marginals.theta"               "marginals.log.tau"
## [11] "marginals.log.kappa"           "marginals.log.variance.nominal"
## [13] "marginals.log.range.nominal"   "marginals.tau"
## [15] "marginals.kappa"               "marginals.variance.nominal"
## [17] "marginals.range.nominal"
```

```
#The two most important things we can extract here are the range parameter
#(kappa), the nominal variance (sigma) and the range (r, radius where
#autocorrelation falls below 0.1)). These are important parameters of the
#spatial autocorrelation: the higher the Kappa, the smoother the spatial
#autocorrelation structure (and the highest the range). Shorter range
#indicates a sharp increase of autocorrelation between closely located points
#and a stronger autocorrelation effect.

inla.emarginal(function(x) x, Mod_p1.field$marginals.kappa[[1]])             #posterior mean for kappa
```

```
## [1] 8.708368
```

```
inla.hpdmarginal(0.95, Mod_p1.field$marginals.kappa[[1]])                    # credible intervarls for Kappa
```

```
##                  low     high
## level:0.95 4.557976 13.49308
```

```
inla.emarginal(function(x) x, Mod_p1.field$marginals.variance.nominal[[1]])  #posterior mean for variance
```

```
## [1] 1.460353
```

```
inla.hpdmarginal(0.95, Mod_p1.field$marginals.variance.nominal[[1]])         # CI for variance
```

```
##                   low     high
## level:0.95 0.9026486 2.078478
```

```
inla.emarginal(function(x) x, Mod_p1.field$marginals.range.nominal[[1]])     #posterior mean for r (in coordinate
s units)
```

```
## [1] 0.3487868
```

```
inla.hpdmarginal(0.95, Mod_p1.field$marginals.range.nominal[[1]])          # CI for r
```

```
##                    low      high
## level:0.95 0.1810647 0.5367121
```

Bayesian model with covariates: spatial

```
###########   Final   #####################################

formula <- y~ ITN_Coverage_2015+Travel_Times_2015+Aridity_2015+
              f(s, model = spde)



start<-Sys.time()# Save start time

fit2 <- inla(formula, family = "binomial", Ntrials = numtrials,
control.family = list(link = "logit"),
data = inla.stack.data(stk.full),
control.compute=list(dic=TRUE, cpo=TRUE, waic=TRUE),
control.predictor = list(compute = TRUE, link = 1, A = inla.stack.A(stk.full)))
```

```
## Warning in inla.core(formula = formula, family = family, contrasts = contrasts, : The A-matrix in the predicto
r (see ?control.predictor) is specified
##   but an intercept is in the formula. This will likely result
##   in the intercept being applied multiple times in the model, and is likely
##   not what you want. See ?inla.stack for more information.
##   You can remove the intercept adding ``-1'' to the formula.
```

```
end<-Sys.time()# Save end time
end-start # time taken to run the model
```

```
## Time difference of 4.441126 secs
```

```
summary(fit2,digits=4)
```

```
## Time used:
##     Pre = 0.7619, Running = 3.418, Post = 0.1997, Total = 4.38
## Fixed effects:
##                      mean     sd 0.025quant 0.5quant 0.975quant    mode kld
## (Intercept)       -3.1125 0.5930    -4.3808  -3.0799    -2.0370 -2.9921   0
## ITN_Coverage_2015  4.9283 1.2420     2.5813   4.8940     7.4703  4.8942   0
## Travel_Times_2015  0.0065 0.0021     0.0024   0.0064     0.0107  0.0064   0
## Aridity_2015       0.0650 0.0319     0.0063   0.0635     0.1330  0.0589   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -2.95 0.401      -3.75    -2.95      -2.17 -2.94
## Theta2 for s  1.79 0.378       1.05     1.79       2.54  1.78
##
## Deviance Information Criterion (DIC) ...............: 698.10
## Deviance Information Criterion (DIC, saturated) ....: 319.64
## Effective number of parameters .....................: 80.62
##
## Watanabe-Akaike information criterion (WAIC) ...: 707.98
## Effective number of parameters .................: 69.65
##
## Marginal log-Likelihood:  -404.09
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
save(fit2,file="fit2.RData")

load("fit2.RData")
summary(fit2)
```

```
## Time used:
##     Pre = 0.762, Running = 3.42, Post = 0.2, Total = 4.38
## Fixed effects:
##                     mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept)       -3.113 0.593     -4.381   -3.080     -2.037 -2.992   0
## ITN_Coverage_2015  4.928 1.242      2.581    4.894      7.470  4.894   0
## Travel_Times_2015  0.006 0.002      0.002    0.006      0.011  0.006   0
## Aridity_2015       0.065 0.032      0.006    0.064      0.133  0.059   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                 mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -2.95 0.401      -3.75    -2.95      -2.17 -2.94
## Theta2 for s  1.79 0.378       1.05     1.79       2.54  1.78
##
## Deviance Information Criterion (DIC) ...............: 698.10
## Deviance Information Criterion (DIC, saturated) ....: 319.64
## Effective number of parameters .....................: 80.62
##
## Watanabe-Akaike information criterion (WAIC) ...: 707.98
## Effective number of parameters .................: 69.65
##
## Marginal log-Likelihood:  -404.09
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
fit2$dic$dic
```

```
## [1] 698.0967
```

```
fit2$waic$waic
```

```
## [1] 707.9754
```

```
# Compare empty model without covariates and
summary(fit0) # WAIC= 1118.40
```

```
## Time used:
##     Pre = 0.829, Running = 0.985, Post = 0.428, Total = 2.24
## Fixed effects:
##               mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -1.112 0.044     -1.199   -1.112     -1.025 -1.112   0
##
## Deviance Information Criterion (DIC) ...............: 1115.24
## Deviance Information Criterion (DIC, saturated) ....: 736.78
## Effective number of parameters .....................: 0.999
##
## Watanabe-Akaike information criterion (WAIC) ...: 1118.40
## Effective number of parameters .................: 4.13
##
## Marginal log-Likelihood:  -558.82
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
summary(fit00) # WAIC= 712.33
```

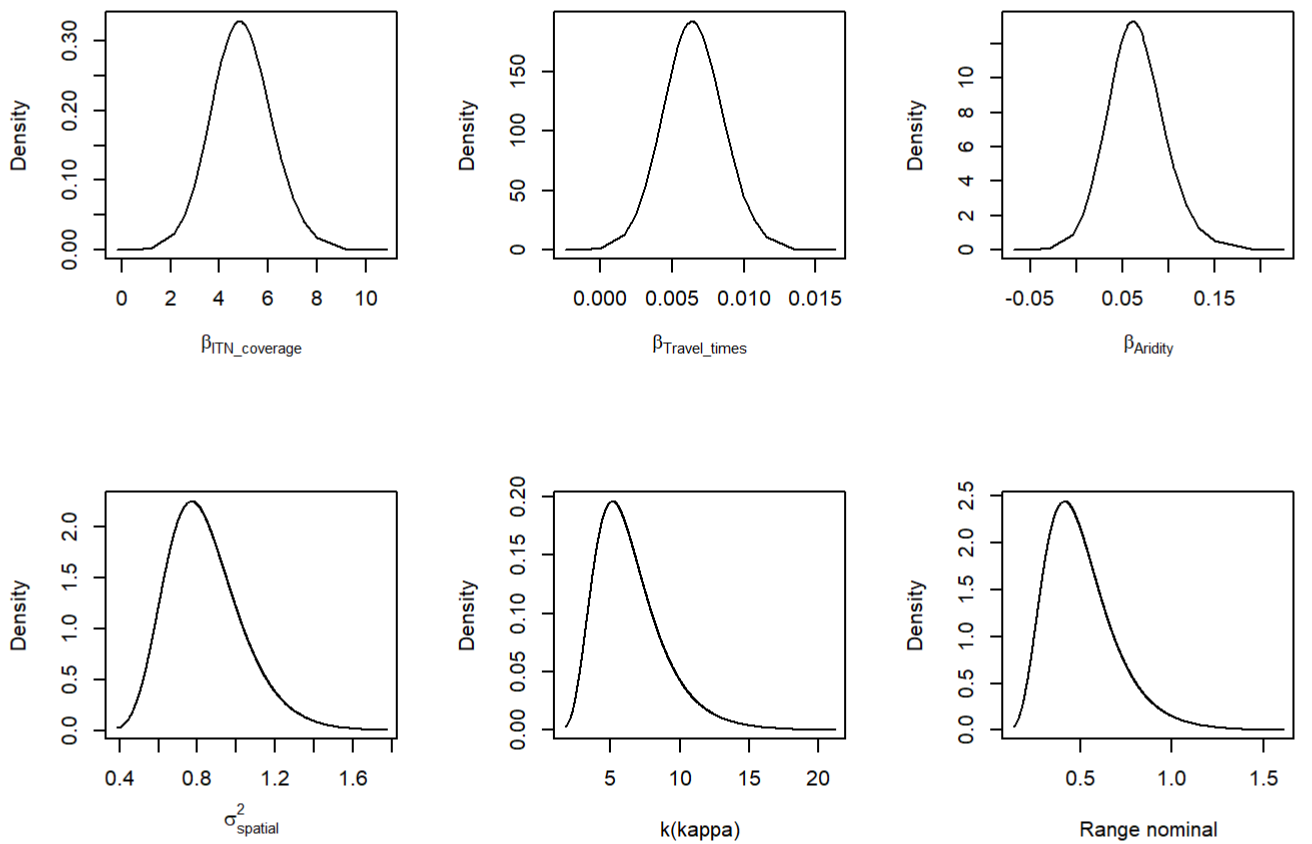
```
## Time used:
##     Pre = 1.43, Running = 3.62, Post = 0.268, Total = 5.31
## Fixed effects:
##      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## b0 -1.164 0.132     -1.428   -1.163     -0.908 -1.163   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                 mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -3.57 0.324      -4.22    -3.57      -2.94 -3.56
## Theta2 for s  2.13 0.269       1.61     2.13       2.66  2.12
##
## Deviance Information Criterion (DIC) ...............: 708.88
## Deviance Information Criterion (DIC, saturated) ....: 330.42
## Effective number of parameters .....................: 95.15
##
## Watanabe-Akaike information criterion (WAIC) ...: 712.33
## Effective number of parameters .................: 74.42
##
## Marginal log-Likelihood:  -409.87
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

```
summary(fit2) # WAIC= 707.98
```

```
## Time used:
##     Pre = 0.762, Running = 3.42, Post = 0.2, Total = 4.38
## Fixed effects:
##                      mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept)        -3.113 0.593     -4.381   -3.080     -2.037 -2.992   0
## ITN_Coverage_2015   4.928 1.242      2.581    4.894      7.470  4.894   0
## Travel_Times_2015   0.006 0.002      0.002    0.006      0.011  0.006   0
## Aridity_2015        0.065 0.032      0.006    0.064      0.133  0.059   0
##
## Random effects:
##   Name     Model
##     s SPDE2 model
##
## Model hyperparameters:
##                mean    sd 0.025quant 0.5quant 0.975quant  mode
## Theta1 for s -2.95 0.401      -3.75    -2.95      -2.17 -2.94
## Theta2 for s  1.79 0.378       1.05     1.79       2.54  1.78
##
## Deviance Information Criterion (DIC) ...............: 698.10
## Deviance Information Criterion (DIC, saturated) ....: 319.64
## Effective number of parameters .....................: 80.62
##
## Watanabe-Akaike information criterion (WAIC) ...: 707.98
## Effective number of parameters .................: 69.65
##
## Marginal log-Likelihood:  -404.09
## CPO, PIT is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

observe the plots for fixed and hyper parameters combined

```
par(mfrow=c(2,3))

#Fixed effects
#plot(fit2$marginals.fixed[[1]], ty = "l", xlab = expression(beta):Intercept", ylab = "Density")
plot(fit2$marginals.fixed[[2]], ty = "l", xlab = expression(beta[ITN_coverage]), ylab = "Density")
plot(fit2$marginals.fixed[[3]], ty = "l", xlab = expression(beta[Travel_times]), ylab = "Density")
plot(fit2$marginals.fixed[[4]], ty = "l", xlab = expression(beta[Aridity]), ylab = "Density")

# Random effects
fit2.res<-inla.spde2.result(fit2, 's', spde, do.transf=TRUE)

plot(fit2.res$marginals.var[[1]], ty = "l", xlab = expression(sigma[spatial]^2), ylab = "Density")
plot(fit2.res$marginals.kap[[1]], type = "l", xlab = "k(kappa)", ylab = "Density")
plot(fit2.res$marginals.range[[1]], type = "l", xlab = "Range nominal", ylab = "Density")
```


Density
0.00
0.10
0.20
0.30
0
2
4
6
8
10
βITN_coverage
Density
0
50
100
150
0.000
0.005
0.010
0.015
βTravel_times
Density
0
2
4
6
8
10
-0.05
0.05
0.15
βAridity
Density
0.0
0.5
1.0
1.5
2.0
0.4
0.8
1.2
1.6
σ2spatial
Density
0.00
0.05
0.10
0.15
0.20
5
10
15
20
k(kappa)
Density
0.0
0.5
1.0
1.5
2.0
2.5
0.5
1.0
1.5
Range nominal

```
dev.off()
```

```
## pdf
##   3
```

```
# Save as png for publication
png(file="Result_Plots.png",pointsize = 16)
par(mfrow=c(2,3))

#Fixed effects
#plot(fit2$marginals.fixed[[1]], ty = "l", xlab = " $ \beta $:Intercept", ylab = "Density")
plot(fit2$marginals.fixed[[2]], ty = "l", xlab = expression(beta[ITN_coverage]), ylab = "Density")
plot(fit2$marginals.fixed[[3]], ty = "l", xlab = expression(beta[Travel_times]), ylab = "Density")
plot(fit2$marginals.fixed[[4]], ty = "l", xlab = expression(beta[Aridity]), ylab = "Density")

# Random effects
fit2.res<-inla.spde2.result(fit2, 's', spde, do.transf=TRUE)

plot(fit2.res$marginals.var[[1]], ty = "l", xlab = expression(sigma[spatial]^2), ylab = "Density")
plot(fit2.res$marginals.kap[[1]], type = "l", xlab = "k(kappa)", ylab = "Density")
plot(fit2.res$marginals.range[[1]], type = "l", xlab = "Range nominal", ylab = "Density")

dev.off()
```

```
## pdf
##   3
```

Estimate and convert the estimates to percentages

```
prev_mean1 <- (fit2$summary.fitted.values[index, "mean"])*100
summary(prev_mean1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.482   9.510  13.638  15.153  19.752  50.635
```

```
save(prev_mean1,file="prev_mean1.RData")
load("prev_mean1.RData")

prev_l1 <- (fit2$summary.fitted.values[index, "0.025quant"])*100
prev_u1 <- (fit2$summary.fitted.values[index, "0.975quant"])*100

wi1<-prev_u1- prev_l1
summary(wi1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.665  22.243  30.537  31.036  39.779  72.374
```

```
save(wi1,file="wi1.RData")
load("wi1.RData")

sd1 <- (fit2$summary.fitted.values[index, "sd"])*100 # standard deviation/error
summary(sd1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.4557  5.8312  7.9960  8.1369 10.3896 19.6907
```

```
save(sd1,file="sd1.RData")
load("sd1.RData")
```

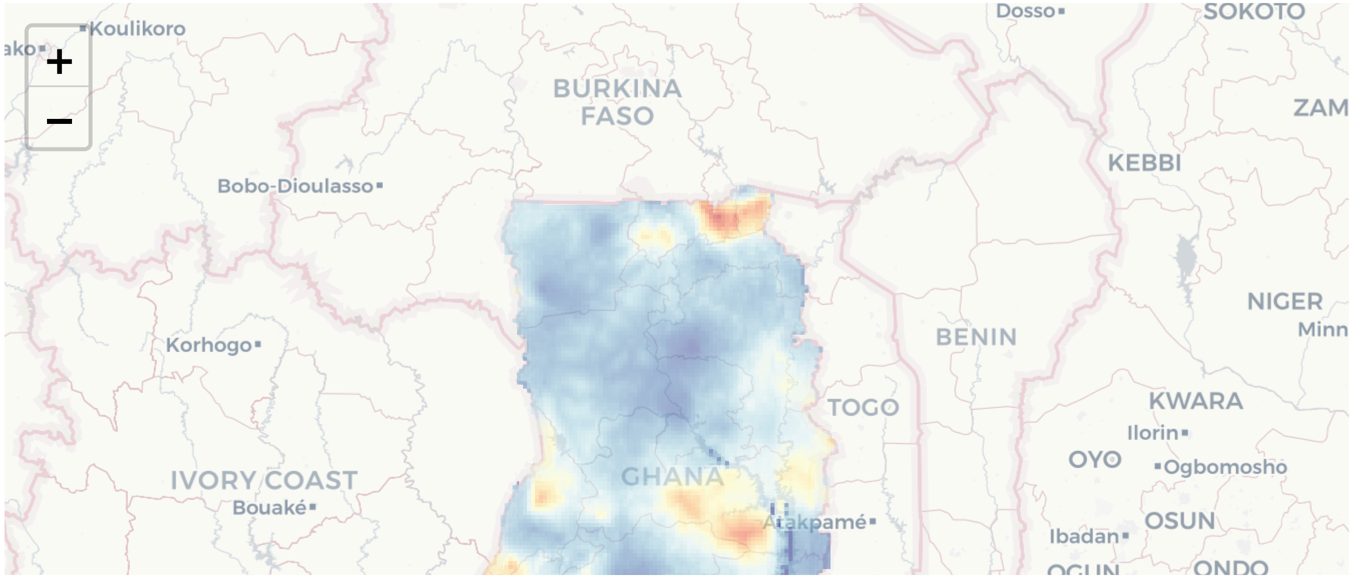
Mapping the predicted values

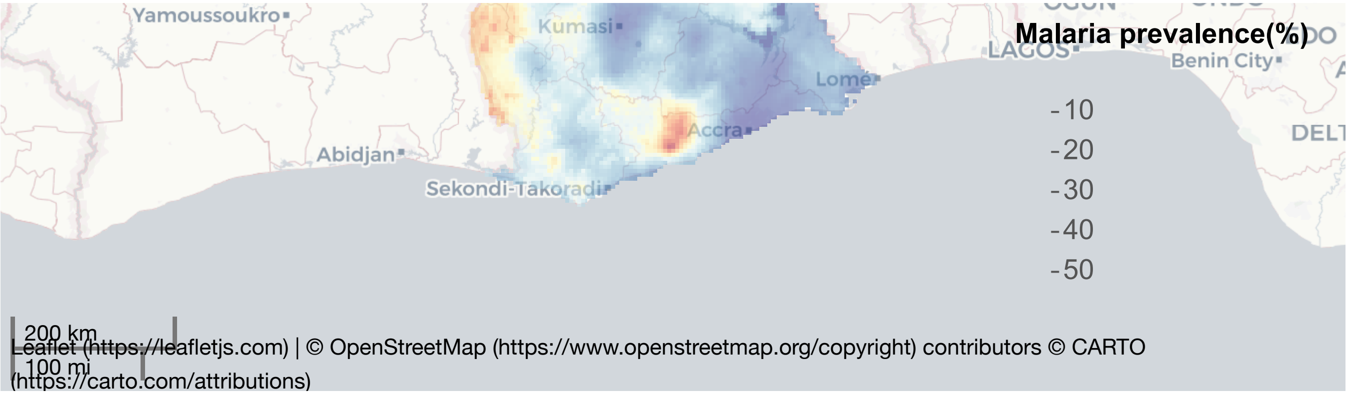
Mapping the predicted malaria prevalence

```
r_prev_mean1 <- rasterize(x = coop, y = ra, field = prev_mean1, fun = mean)

summary(r_prev_mean1)
```

```
##              layer
## Min.       0.481997
## 1st Qu.    9.509797
## Median    13.637992
## 3rd Qu.   19.751902
## Max.      50.634835
## NA's    5323.000000
```

```
# Write raster files and save as GeoTIFF file
writeRaster(r_prev_mean1, "r_prev_mean1.tif", filetype = "GTiff", overwrite = TRUE)

# Mapping predicted malaria prevalence
pal <- colorNumeric("RdYlBu", values(r_prev_mean1),reverse=T, na.color = "transparent")


prev1_map<-leaflet() %>% addProviderTiles(providers$CartoDB.Positron) %>%
  addRasterImage(r_prev_mean1, colors = pal, opacity = 0.5) %>%
  addLegend("bottomright", pal = pal, values = values(r_prev_mean1), title = "Malaria prevalence(%)") %>%
  addScaleBar(position = c("bottomleft"))

prev1_map
```



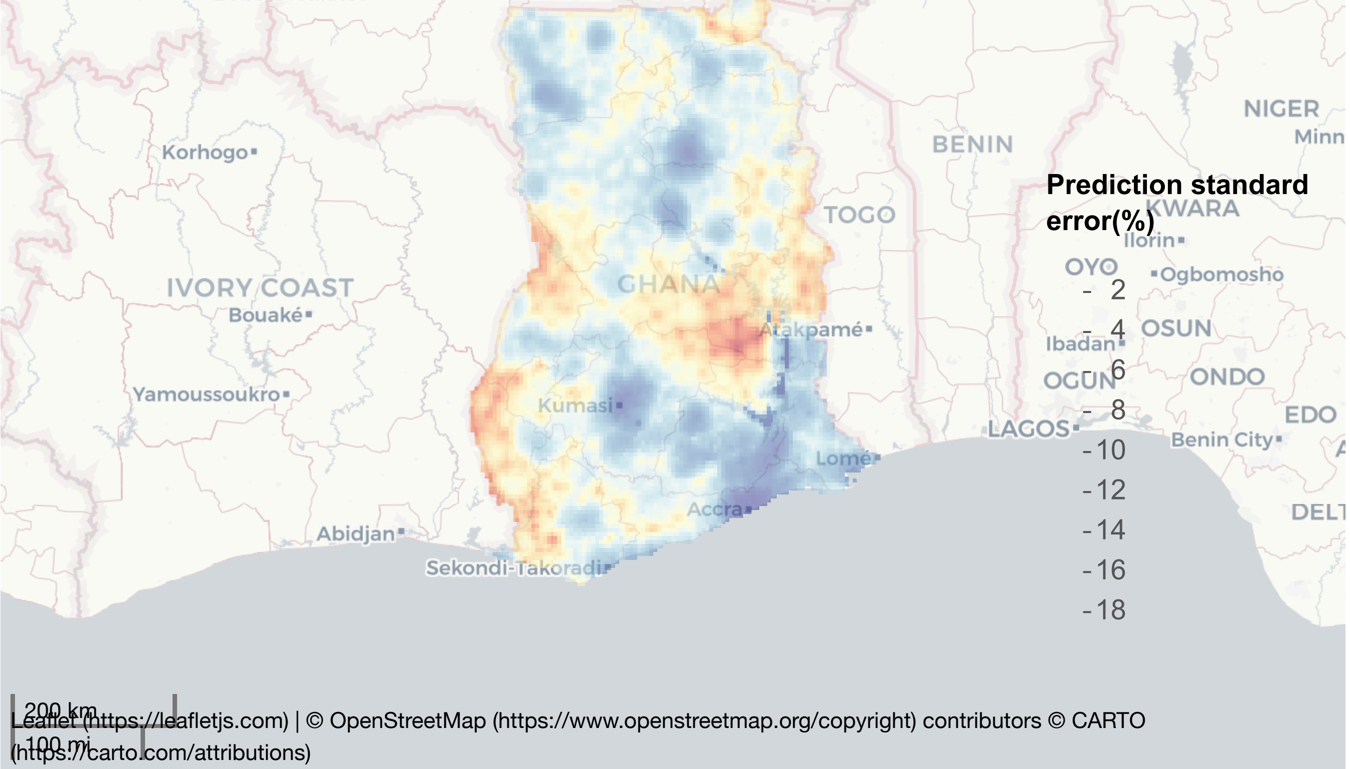



```
save(prev1_map,file="prev1_map.RData")
```

Mapping the predicted standard error

Mapping prediction standard errors (%)

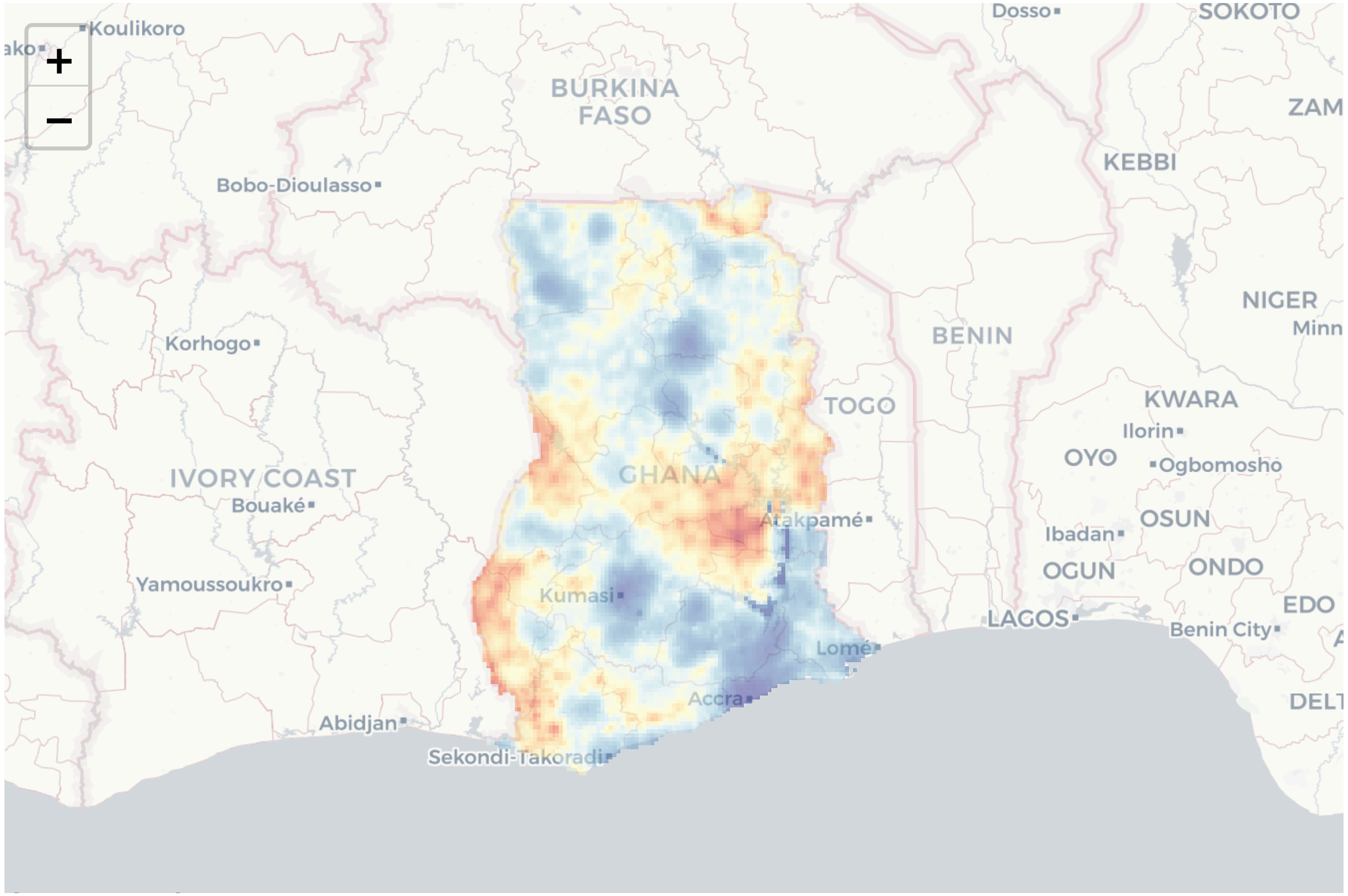
```
r_sd1 <- rasterize(x = coop, y = ra, field = sd1, fun = mean)

# Write raster files and save as GeoTIFF file
writeRaster(r_sd1, "r_sd1.tif", filetype = "GTiff", overwrite = TRUE)

pal <- colorNumeric("RdYlBu", values(r_sd1),reverse=T, na.color = "transparent")

sd1_map<-leaflet() %>% addProviderTiles(providers$CartoDB.Positron) %>%
  addRasterImage(r_sd1, colors = pal, opacity = 0.5) %>%
  addLegend("bottomright", pal = pal, values = values(r_sd1), title = "Prediction standard<br> error(%)") %>%
  addScaleBar(position = c("bottomleft"))

sd1_map
```

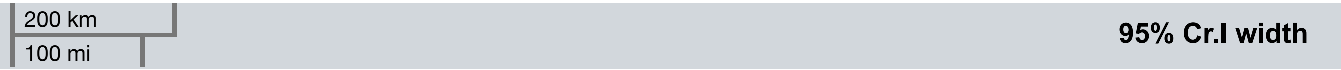



```
save(sd1_map,file="sd1_map.RData")
```

Mapping the 95% Credible interval width

```
r_wi1 <- rasterize(x = coop, y = ra, field = wi1, fun = mean)

# Write raster files and save as GeoTIFF file
writeRaster(r_wi1, "r_wi1.tif", filetype = "GTiff", overwrite = TRUE)

pal <- colorNumeric("RdYlBu", values(r_wi1),reverse=T, na.color = "transparent")

wi1_map<-leaflet() %>% addProviderTiles(providers$CartoDB.Positron) %>%
  addRasterImage(r_wi1, colors = pal, opacity = 0.5) %>%
  addLegend("bottomright", pal = pal, values = values(r_wi1), title = "95% Cr.I width") %>%
  addScaleBar(position = c("bottomleft"))

wi1_map
```






```
save(wi1_map,file="wi1_map.RData")
```

End of Geospatial modelling