

Enhancing Disease Maps with Labels in R

GRAPH Network & WHO, supported by the Global Fund to fight HIV, TB & Malaria

October 2023

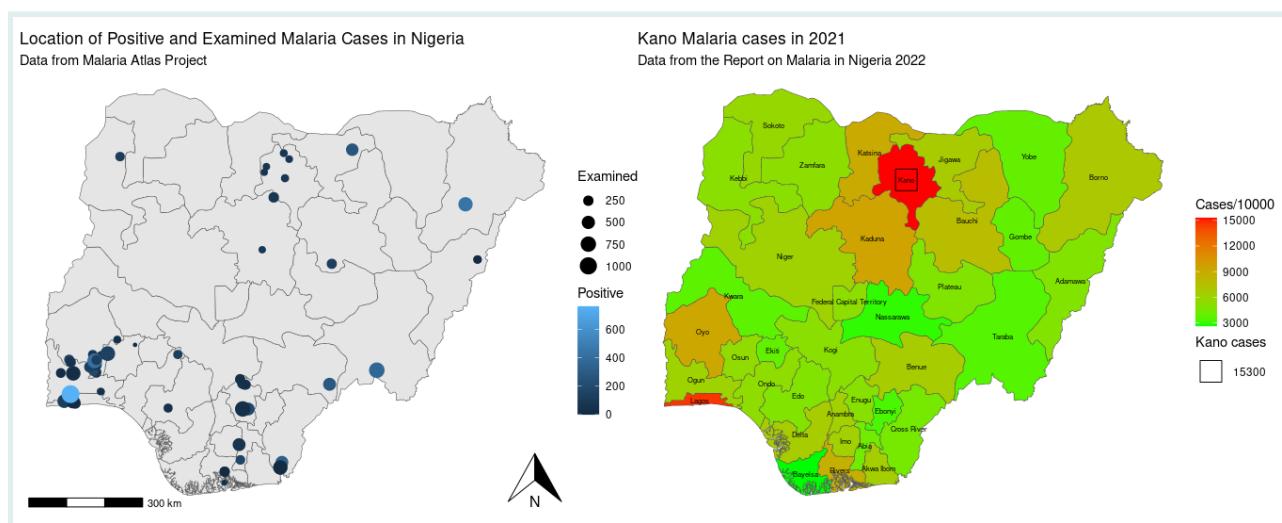
This document is a draft of a lesson made by the GRAPH Network, a non-profit headquartered at the University of Geneva Global Health Institute, in collaboration with the World Health Organization, under a Global Fund 2023 grant to create e-learning modules to build in-country data capacity for epidemiological and impact analysis for National HIV, TB and malaria programs

Introduction
Packages
Data Preparation
Building a Simple Choropleth Map
Adding Continuous Data Indicators to the Choropleth Map
Exploring Malaria Case Increase Rates Using Choropleth Maps
Labeling the Choropleth Map with State Names
Displaying Combined State Names and Increase Rates on the Choropleth Map
Highlighting a Specific Region on the Map while Preserving Context
Labeling Point Locations: Exploring Malaria Positive Rate and Incidence
Final Thoughts
References
WRAP UP!
Answer Key

Introduction

In the geospatial data visualization, maps are powerful storytelling tools. However, a map without clear annotations and labels is like a book without titles or chapter headings. While the story might still be there, it becomes significantly harder to understand, interpret, and appreciate.

In this lesson, we place a special emphasis on the importance of annotating and labeling maps. Proper annotation transforms a simple visualization into an informative guide, allowing viewers to quickly grasp complex spatial data. With precise labeling, areas of interest can be immediately recognized, facilitating a clearer comprehension of the data's narrative.



Learning objectives

Learning Objectives: Advanced Geospatial Visualization Techniques

By the end of this section, you should be able to:

1. Incorporate continuous data indicators into choropleth maps for enhanced granularity.
2. Calculate and visualize the increase rates of malaria cases using choropleth maps.
3. Effectively overlay state names onto choropleth maps, ensuring clarity and readability.
4. Seamlessly integrate state names with increase rates on maps without compromising legibility.
5. Apply techniques to accentuate specific regions on a map while retaining the overall context.
6. Determine optimal point placement strategies and integrate them effectively into geospatial visualizations.

Upon mastering these objectives, you will have the tools and knowledge to create rich, detailed, and informative geospatial visualizations.

Packages

```
# Load packages
if(!require(pacman)) install.packages("pacman")
pacman::p_load(malariaAtlas,
  ggplot2,
  geodata,
  dplyr,
  here,
  readr,
  sf,
  patchwork)

# Unable scientific notation
options(scipen=100000)
```

Data Preparation

Before diving into any visualization or analysis, it's essential to load and preprocess our data. This includes reading in datasets, merging related information, and filtering out unnecessary or irrelevant entries.

```
# Reading the geographical shapefile data
nga_adm1 <- sf::st_read(here::here("data/raw/NGA_adm_shapefile/NGA_adm1.shp"))
%>%
sf::st_simplify(dTolerance = 1000) # simplify shapefile to speed up
visualization
```

```
## Reading layer `NGA_adm1` from data source
##
`/Users/kendavidn/Dropbox/tgc_github_projects/epi_reports_staging/data/raw/NGA_adm_shapef
##   using driver `ESRI Shapefile'
## Simple feature collection with 38 features and 9 fields
## Geometry type: MULTIPOLYGON
## Dimension:      XY
## Bounding box:  xmin: 2.668431 ymin: 4.270418 xmax: 14.67642 ymax: 13.89201
## Geodetic CRS:  WGS 84
```

The geographical shapefile data for Nigeria's administrative boundaries (like states or provinces) is read and stored in the `nga_adm1` object.

```
# Reading the attribute data related to malaria cases
malaria_cases <- read_csv(here::here("data/malaria.csv"))
```

This step loads data related to malaria cases in different states of Nigeria.

```
# Filtering out the 'Water body' entries from the geographical data
nga_adm1 <- filter(nga_adm1, NAME_1 != "Water body")

# Merging the geographical data with the malaria cases data
malaria <- malaria_cases %>%
  left_join(nga_adm1, by = c("state_name" = "NAME_1")) %>%
  st_as_sf()
```

Here, we're combining the geographical boundaries data with the malaria cases data using state names as a reference. This merged data is then converted into a format suitable for geospatial visualizations.

```
# Filtering down to essential columns for our analysis
malaria2 <- malaria %>%
  select(state_name, cases_2000, cases_2006, cases_2010, cases_2015,
  cases_2021, geometry)
```

We're narrowing down our dataset to specific columns, mainly the state names, the malaria cases from various years, and the geographical boundaries of these states (i.e. geometry).

```
# Reading in population data for different regions of Nigeria
population_nigeria <- read_csv(here::here("data/population_nigeria.csv"))
```

This step loads data indicating the population of different states or regions in Nigeria.

```
# Combining the population data with our malaria data
malaria3 <- malaria2 %>%
  left_join(population_nigeria, by = c("state_name")) %>%
  st_as_sf()
```

By merging the population data with our malaria cases data, we enrich our dataset. This combined data allows for more comprehensive visualizations and analyses, such as calculating incidence rates or assessing trends relative to population size.

Building a Simple Choropleth Map



REMINDER

Choropleth maps are powerful visualization tools that display divided geographical areas shaded or patterned in proportion to the value of a variable.

In this example, we'll be using a choropleth map to visualize the distribution of malaria cases across different regions of Nigeria for the year 2021.

```
# Constructing the choropleth map using ggplot2
ggplot(data=malaria3) +
  geom_sf(aes(fill=cases_2021/1000)) + # The fill color is determined by the
  # number of malaria cases in 2021, scaled per 1000
  labs(title = "Nigeria Malaria Distributed Cases in 2021", fill = "Cases per
    1000") + # Adding labels and title to the plot
  scale_fill_continuous(low = "green", high = "red") + # Using a continuous
  # color scale transitioning from green to red
  theme_void() # Using a minimal theme for better visualization of the map

## Error in `calc_limits_bbox()`:
## ! Scale limits cannot be mapped onto spatial coordinates in `coord_sf()`
## i Consider setting `lims_method = "geometry_bbox"` or `default_crs =
NULL`.
```

Here's a detailed explanation of the code:

- `ggplot(data=malaria3)`: Initiates a ggplot object using the `malaria3` dataset.
- `geom_sf(aes(fill=cases_2021/1000))`: Adds the geographical data from `malaria3` and fills each region based on the number of malaria cases in 2021, scaled down by a factor of 1000. This effectively represents the number of cases per thousand people.
- `labs(title = "Nigeria Malaria Distributed Cases in 2021", fill = "Cases per 1000")`: Specifies the title of the plot and the label for the color scale.
- `scale_fill_continuous(low = "green", high = "red")`: Applies a continuous color scale where regions with fewer cases are colored green and regions with more cases are colored red.

The resulting plot provides a clear view of how malaria cases are distributed across Nigeria in 2021, with the color intensity indicating the magnitude of cases in each region.

Practice 1: Modify the provided choropleth map to visualize the distribution of malaria cases in Nigeria for the year 2015.

Instructions

1. Update the data mapping in the `geom_sf()` function to reflect malaria cases for the year 2015.
2. Adjust the title in the `labs()` function to indicate that the visualization pertains to 2015.
3. Change the color gradient in `scale_fill_continuous()` to transition from blue (low cases) to yellow (high cases).

Below a starter code:



```
# Constructing the choropleth map for 2015 using ggplot2
ggplot(data=malaria3) +
  geom_sf(aes(fill=_____)) + # Fill in the correct data
  # column for 2015
  labs(title = "_____", fill = "Cases per
    1000") + # Update the title appropriately
  scale_fill_continuous(_____) + # Modify the color scale
  theme_void()
```

Adding Continuous Data Indicators to the Choropleth Map

When analyzing disease data, it's often useful to look beyond raw case numbers and focus on rates, particularly incidence rates. The incidence rate provides a normalized measure that can account for differing population sizes across regions, making comparisons more meaningful.

Understanding Incidence

The incidence of a disease is a cornerstone of epidemiological research. It quantifies the number of new cases of a disease that occur within a specific time frame, typically a year, relative to a population at risk.

Mathematically, it's given by:

$$\text{Incidence} = \frac{\text{Number of new cases during the time period}}{\text{Population at risk at the start of the period}}$$

In this context, we're looking at the incidence of malaria in different states of Nigeria for the year 2021. Specifically, we'll compute the incidence rate by dividing the number of new malaria cases in 2021 by the population of each state from the last available census in 2019.

The following R code accomplishes this and visualizes the data:

```
# Find the centroid coordinates of each state to position labels
centroid_coords <- st_coordinates(st_centroid(malaria2$geometry))

# Visualizing Malaria Incidence in 2021 using a Choropleth Map
ggplot(data = malaria3) +
  # Filling each state with a color representing the incidence rate in 2021.
  geom_sf(aes(fill = round(cases_2021/population_2019, 2))) +
  # Labeling each state with its incidence rate.
  geom_text(aes(x = centroid_coords[, 1], y = centroid_coords[, 2], label =
    round(cases_2021/population_2019, 2)), size = 2) +
  # Adding title and legend title.
  labs(title = "Nigeria Malaria Incidence in 2021", fill = "Incidence") +
  # Using a continuous color scale from green (low incidence) to red (high
  # incidence).
  scale_fill_continuous(low = "green", high = "red") +
  # Using a minimalistic theme for clearer visualization.
  theme_void()
```

```
## Error in `calc_limits_bbox()`:
## ! Scale limits cannot be mapped onto spatial coordinates in `coord_sf()`
## i Consider setting `lims_method = "geometry_bbox"` or `default_crs =
NULL`.
```

Here's a brief explanation of the visualization:

- The `geom_sf()` function creates a choropleth map where each state's color represents its malaria incidence rate in 2021.
- The `geom_text()` function labels each state with its specific incidence rate, positioning each label at the state's centroid.
- The color scale, transitioning from green to red, visually emphasizes regions with higher incidence rates.

This visualization offers an immediate grasp of the malaria situation in Nigeria, revealing areas of high incidence that might need more focused public health interventions.

Exploring Malaria Case Increase Rates Using Choropleth Maps

Understanding the change in the number of disease cases over time can provide insights into the effectiveness of interventions, the progression of the disease, and areas where increased resources might be needed. In this context, we're looking to visualize the percentage increase in malaria cases from 2015 to 2021 across different states in Nigeria.

Computing the Increase Rate

The increase rate for each state is computed as:

$$\text{Increase Rate} = \left(\frac{\text{Cases in 2021} - \text{Cases in 2015}}{\text{Cases in 2015}} \right) \times 100\%$$

This formula provides the percentage growth (or decrease) in malaria cases from 2015 to 2021.

Visualization of Increase Rates

Let's delve into the code that accomplishes this visualization:

```
# Calculate the increase rate for each state
malaria3 %>%
  mutate(increase_rate = round(((cases_2021 - cases_2015) / cases_2015) *
    100)) -> malaria3

# Visualizing the increase rates using a choropleth map
ggplot(data = malaria3) +
  # Coloring each state based on its increase rate
  geom_sf(aes(fill = increase_rate), color="white", size = 0.2) +
  # Using a continuous color scale to represent increase rates, transitioning
  # from green to red
  scale_fill_continuous(name="Increase rate (%)", limits=c(0,70), low =
    "green", high = "red", breaks=c(0, 20, 40, 60))+

  # Labeling each state with its increase rate percentage
  geom_text(aes(x = centroid_coords[, 1], y = centroid_coords[, 2], label =
    paste0(increase_rate, "%")), size = 2)+

  # Adding a title to the plot
  labs(title = "Nigeria Malaria Increase rate in 2021 compared with 2015")+
  theme_void()

## Error in `calc_limits_bbox()`:
## ! Scale limits cannot be mapped onto spatial coordinates in `coord_sf()`
## i Consider setting `lims_method = "geometry_bbox"` or `default_crs =
NULL`.
```

In this visualization:

- The `geom_sf()` function creates the choropleth map where each state's color intensity represents its malaria increase rate.
- `scale_fill_continuous()` sets the color scale for the increase rates, making areas of higher increase more prominent.
- `geom_text()` adds labels to each state, providing exact percentage values.
- The resulting map allows us to quickly identify regions with significant growth in malaria cases over the selected period.

Through this visualization, stakeholders can pinpoint regions where malaria is on the rise and potentially allocate resources more effectively.

Labeling the Choropleth Map with State Names

In a choropleth map, while color gradients offer a visual cue to understand the distribution of a variable across regions, adding labels can significantly enhance the clarity of the visualization. This is especially true when audiences might not be familiar with all geographical boundaries shown. In the case of our malaria dataset, adding state names to the map makes the data more accessible and understandable.

Let's break down the code:

```
# Constructing a choropleth map with state names
ggplot(data = malaria3) +
  # Fill each state based on the number of malaria cases in 2021, scaled per
  # 10,000
  geom_sf(aes(fill = cases_2021/10000)) +
  # Add the name of each state to its centroid
  geom_text(aes(x = centroid_coords[, 1], y = centroid_coords[, 2], label =
    state_name), size = 2, check_overlap = TRUE) +
  # Add titles and labels
  labs(title = "Nigeria Malaria cases in 2021", fill = "Cases/10000") +
  # Use a continuous color scale from green (low case numbers) to red (high
  # case numbers)
  scale_fill_continuous(low = "green", high = "red") +
  theme_void()
```

```
## Error in `calc_limits_bbox()`:
## ! Scale limits cannot be mapped onto spatial coordinates in `coord_sf()`
## i Consider setting `lims_method = "geometry_bbox"` or `default_crs =
NULL`.
```

Here's a detailed explanation of the visualization:

- `geom_sf(aes(fill = cases_2021/10000))`: This creates the choropleth map where each state's color represents the number of malaria cases in 2021, scaled down by a factor of 10,000.
- `geom_text(aes(...))`: This function places text on the plot. In this instance, it's used to add the name of each state to its geographical centroid. The `check_overlap = TRUE` parameter ensures that ggplot checks for overlapping text labels and tries to prevent labels from overlapping.
- `labs(title = "Nigeria Malaria cases in 2021", fill = "Cases/10000")`: This function adds a title to the plot and a label to the color scale.

- `scale_fill_continuous(low = "green", high = "red")`: This sets the color scale for the map, transitioning from green for states with fewer cases to red for those with more cases.

The resulting visualization is a clear and informative map of malaria cases across Nigeria in 2021, with each state labeled for easy reference.

Displaying Combined State Names and Increase Rates on the Choropleth Map

Visualizations can convey a wealth of information when they incorporate multiple data points in an intuitive manner. By pairing state names with their corresponding increase rates, we can provide a richer, more detailed view of the data without overwhelming the audience.

Let's delve into this visualization:

We want to present a choropleth map showcasing the malaria cases per 10,000 residents across Nigerian states in 2021, with labels that combine state names and their respective increase rates from 2015.

```
# Combine the state names and their respective increase rates into a single
# label
malaria3$label_text <- paste(malaria3$state_name, malaria3$increase_rate, "%")

# Calculate the centroid of each state to serve as reference points for the
# labels
centroid_coords <- st_coordinates(st_centroid(malaria3$geometry))

# Visualize the data
ggplot(data = malaria3) +
  # Create a choropleth map shaded based on the number of malaria cases in
  # 2021 per 10,000 residents
  geom_sf(aes(fill = cases_2021/10000)) +
  # Add combined labels (state name and increase rate) to each state's
  # centroid
  geom_text(aes(x = centroid_coords[, 1],
                y = centroid_coords[, 2],
                label = label_text), size = 2) +
  # Add titles and customize the color legend
  labs(title = "Nigeria Malaria cases in 2021", fill = "Cases/10000") +
  scale_fill_continuous(low = "green", high = "red") +  # Set color gradient
  theme_void()  # Apply a minimal theme for clarity
```

```
## Error in `calc_limits_bbox()`:
## ! Scale limits cannot be mapped onto spatial coordinates in `coord_sf()` 
## i Consider setting `lims_method = "geometry_bbox"` or `default_crs =
NULL`.
```

In this visualization:

- The line `malaria3$label_text <- paste(malaria3$state_name, malaria3$increase_rate, "%")` constructs our combined labels by concatenating the state name with its increase rate, followed by a percentage sign.
- `geom_text()` adds these combined labels to the plot. It positions each label at the centroid of the corresponding state, ensuring the labels are centered and easily associated with their respective regions.
- `scale_fill_continuous(low = "green", high = "red")` assigns a color gradient based on the number of malaria cases. States with fewer cases will be colored green, transitioning to red for states with more cases.

This approach allows us to efficiently communicate two vital pieces of information (cases per 10,000 and increase rate) within the same visualization while keeping the name of states for easy identification.

Practice 2: Create a single choropleth map that showcases the malaria increase rates across different states in Nigeria with each state labeled by its name

Instructions

1. Start by setting up the base plot using the `malaria3` dataset.
2. Create a choropleth map where the fill color represents the increase rate from 2015 to 2021.
3. Label each state with its name using the centroids.
4. Customize the color scale to transition from green (low increase) to red (high increase).
5. Add appropriate titles and labels to the plot.

Below a starter code:



```
# Combining visualization of increase rates with state names
ggplot(data = malaria3) +
  _____ + # Fill in the code to generate the choropleth map
  _____ + # Add state names to each region
  _____ + # Specify the color scale for increase rates
  _____ # Add titles and labels
```

Highlighting a Specific Region on the Map while Preserving Context

Sometimes, you might want to draw attention to a particular area or region on your map, without omitting details from the surrounding areas. By using specific graphical

elements, like larger markers or distinctive colors, you can emphasize certain regions while still showcasing the broader data. In this example, we're focusing on the "Kano" region of Nigeria.

```
# Calculate centroid coordinates for labeling
centroid_coords <- st_coordinates(st_centroid(malaria3$geometry))

# Visualize the malaria cases across Nigeria with an emphasis on Kano
ggplot(data = malaria3) +
  # Create a choropleth map with color based on malaria cases in 2021
  geom_sf(aes(fill = cases_2021/1000)) +
  # Set a continuous color gradient from green to red
  scale_fill_continuous(low = "green", high = "red") +
  
  # Overlay a point on Kano to emphasize it. The size of the point corresponds
  # to the number of cases
  geom_point(data = subset(malaria3, state_name == "Kano"),
             aes(x = st_coordinates(st_centroid(geometry))[1],
                  y = st_coordinates(st_centroid(geometry))[2], size =
round(cases_2021/1000)),
             color = "black", shape = 22, fill = "transparent") +
  
  # Label each region with its name
  geom_text(aes(x = centroid_coords[, 1], y = centroid_coords[, 2], label =
state_name), size = 2, check_overlap = TRUE) +
  
  # Customize the scale of the size of the emphasized point
  scale_size_continuous(range = c(1, 12)) +
  
  # Add title and legends
  labs(title = "Kano Malaria cases in 2021", subtitle = "Data from the Report
on Malaria in Nigeria 2022", size = "Kano cases", fill =
"Cases/10000",) +
  
  # Apply a minimal theme for clarity
  theme_void()
```

```
## Error in `calc_limits_bbox()`:
## ! Scale limits cannot be mapped onto spatial coordinates in `coord_sf()`
## i Consider setting `lims_method = "geometry_bbox"` or `default_crs =
NULL`.
```

In this visualization:

- The `geom_point()` function is utilized to lay a circle over the Kano region. The size of the circle signifies the number of cases in Kano in 2021. The circle is designed transparent (`fill = "transparent"`) with a bold black boundary (`color = "black"`) to set it apart.
- `geom_text()` adds the names of all regions to the map. It places each label near the geographical center of the respective region. The parameter `check_overlap = TRUE` prevents labels from overlapping when possible, ensuring readability.

RECAP

While “Kano” is emphasized, all other regions are also displayed with their respective color shading based on malaria cases. This offers a holistic view of the situation across Nigeria, enabling viewers to compare Kano with other regions.

Such an approach is invaluable when you wish to spotlight specific details or areas of interest without sidelining the broader dataset, enriching your data presentations.

Labeling Point Locations: Exploring Malaria Positive Rate and Incidence

Mapping and visualizing specific data points on a geographical map can provide crucial insights, especially when dealing with epidemiological data. Let's delve into the code to understand the processes and the visualization we're aiming to achieve:

```
# Data Retrieval
# The malariaAtlas package provides the `getPR` function to access the
# parasite rate (PR).
# PR is an essential indicator of malaria prevalence.

# Fetching data for Nigeria for both malaria species
nigeria_pr <- malariaAtlas::getPR(ISO = "NGA", species = "both") %>%
  # Filtering out records with missing PR values
  filter(!is.na(pr)) %>%
  # Removing any rows with missing longitude or latitude
  drop_na(longitude, latitude) %>%
  # Converting the data into a spatial dataframe to facilitate mapping
  st_as_sf(coords = c("longitude", "latitude"), crs = 4326)
```

This chunk fetches malaria prevalence data for Nigeria. After retrieval, the data is cleansed by filtering out missing values. It's then transformed into a format suitable for geospatial visualization (`sf` object).

```

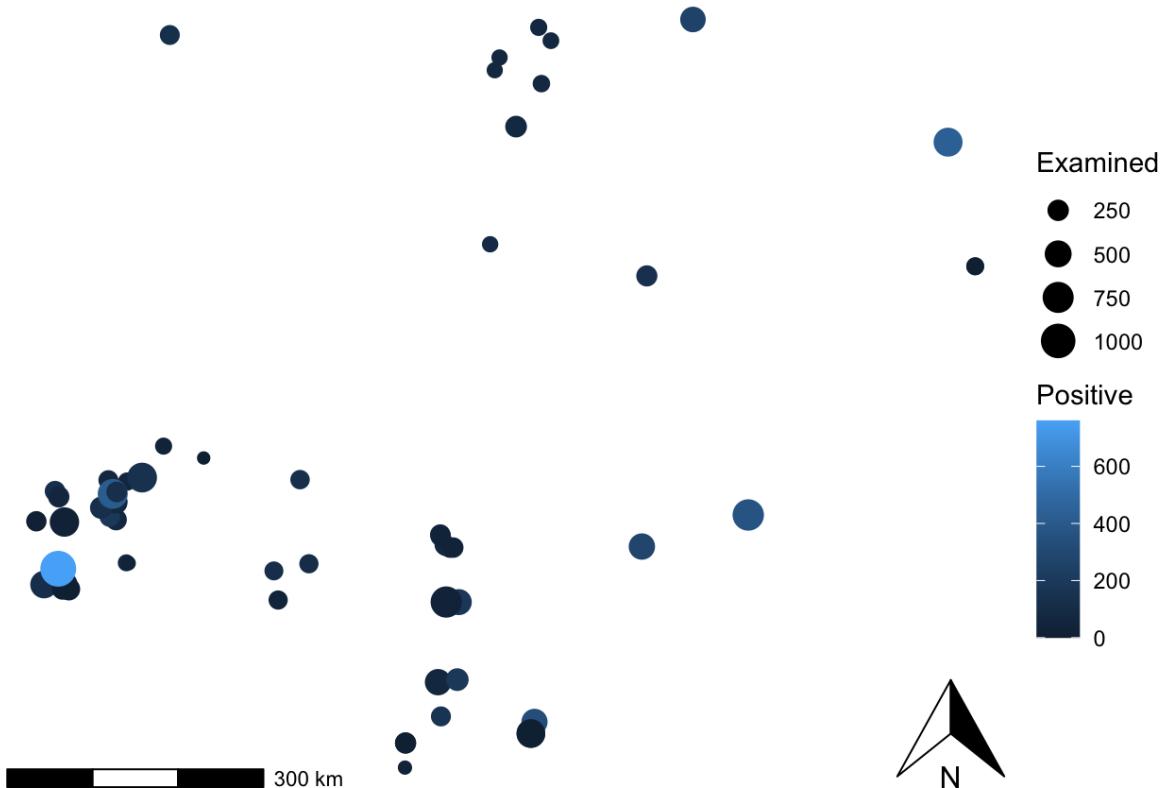
# Setting up a geospatial visualization using ggplot2

# Starting the plot
ggplot() +
  # Plotting the administrative boundaries of Nigeria
  geom_sf(data = nga_adm1) +
  # Adding points for each testing location
  # The color of each point indicates if the test was positive, and the size
  # represents the number of people tested
  geom_sf(aes(size = examined, color = positive), data = nigeria_pr) +
  # Setting titles, axis labels, and legends for the plot
  labs(title = "Location of Positive and Examined Malaria Cases in Nigeria",
       subtitle = "Data from Malaria Atlas Project",
       color = "Positive",
       size = "Examined") +
  # Adding labels for longitude and latitude
  xlab("Longitude") +
  ylab("Latitude") +
  # Incorporating a north arrow to provide orientation
  ggspatial::annotation_north_arrow(location = "br") +
  # Incorporating a scale bar to assist in distance interpretation
  ggspatial::annotation_scale(location = "bl") +
  # Applying a minimalistic theme for visual clarity
  theme_void()

```

Location of Positive and Examined Malaria Cases in Nigeria

Data from Malaria Atlas Project



RECAP

This code chunk visualizes the malaria data on a map. It highlights regions based on the number of malaria tests and their outcomes. Specific tools from the `ggspatial` package are used to add cartographic elements, ensuring the map is informative.

Practice 3: Reflecting the positive rate by size and color**CHALLENGE**

Create a new visualization that represents the positive rate (number of positive cases divided by the number examined) for each location. Modify the size of the points to reflect the positive rate.

Final Thoughts

Geospatial data visualization is more than just plotting data on a map. It's about narrating a story that's rooted in location and space. This lesson delved deep into a layers-based narrative approach, from the importance of clear annotations to the integration of diverse data types for a richer understanding.

References

- [ggplot2: Elegant Graphics for Data Analysis \(3e\)](#)
- [R for Data Science](#)
- [Geocomputation with R](#)

WRAP UP!

In this lesson on maps labeling, explored in depth the following points:

RECAP

- **Annotations & Labels:** Clear annotations transform maps, making complex spatial data easily comprehensible.
- **Advanced Mapping Techniques:** We explored how to integrate continuous indicators, overlay labels, emphasize regions, and



determine optimal point placements on maps.

- **Practical Application:** Using a malaria dataset, we demonstrated data preparation, visualization, and interpretation in R.
- **R packages:** Packages like `ggplot2`, `sf`, and `ggspatial` facilitate advanced geospatial visualizations.
- **Highlight with Context:** It's essential to provide a holistic view, even when emphasizing specific areas.

Answer Key

Solution for practice 1

To visualize the distribution of malaria cases in Nigeria for the year 2015, follow the instructions provided in the exercise. Here's the complete solution:

```
# Constructing the choropleth map for 2015 using ggplot2
ggplot(data = malaria3) +
  geom_sf(aes(fill = cases_2015 / 1000)) + # Updated data column to reflect
  # 2015
  labs(title = "Nigeria Malaria Distributed Cases in 2015", fill = "Cases per
    1000") + # Updated title for 2015
  scale_fill_continuous(low = "blue", high = "yellow") + # Modified color
  # scale to blue-to-yellow gradient
  theme_void()
```

Solution for practice 2

To combine the visualization of increase rates with state names, follow the steps below:

```

# Combining visualization of increase rates with state names
ggplot(data = malaria3) +
  # Create a choropleth map with fill color based on the increase rate
  geom_sf(aes(fill = increase_rate), color="white", size = 0.2) +
  
  # Label each region with its state name
  geom_text(aes(x = centroid_coords[, 1],
                y = centroid_coords[, 2],
                label = state_name), size = 2, check_overlap = TRUE) +
  
  # Specify the color scale for increase rates
  scale_fill_continuous(name="Increase rate (%)",
                        limits=c(0,70),
                        low = "green",
                        high = "red",
                        breaks=c(0, 20, 40, 60)) +
  
  # Add a title and legend to the plot
  labs(title = "Nigeria Malaria Increase Rate from 2015 to 2021",
       fill = "Increase Rate (%)") +
  
  theme_void() # Apply a minimalistic theme for clarity

```

In this solution, the choropleth map is created based on the increase rate of malaria cases from 2015 to 2021. Each state in Nigeria is labeled by its name, and the color gradient (from green to red) showcases the magnitude of the increase rate. The map is enhanced with a title and a legend to ensure clarity and comprehension.

Solution for practice 3

Create a visualization that represents the positive rate:

```

# Adding a positive rate column
nigeria_pr$positive_rate <- (nigeria_pr$positive / nigeria_pr$examined) * 100

ggplot() +
  geom_sf(data = nga_adm1) +
  geom_sf(aes(size = positive_rate, color = positive_rate), data =
    nigeria_pr) +
  labs(title = "Location and Positive Rate of Malaria Cases in Nigeria",
       subtitle = "Data from Malaria Atlas Project",
       color = "Positive Rate (%)",
       size = "Positive Rate (%)") +
  xlab("Longitude") +
  ylab("Latitude") +
  ggspatial::annotation_north_arrow(location = "br") +
  ggspatial::annotation_scale(location = "bl") +
  theme_void()

```

Contributors

The following team members contributed to this lesson:



IMAD EL BADISY

Data Science Education Officer
Deeply interested in health data



JOY VAZ

R Developer and Instructor, the GRAPH Network
Loves doing science and teaching science
