# MATH2270/MATH2237/MATH2404 Assignment 3

## Name

Alve Reza Sazim

## Student number

S4085418

## Assessment declaration checklist

Please carefully read the statements below and check each box if you agree with the declaration. If you do not check all boxes, your assignment will not be marked. If you make a false declaration on any of these points, you may be investigated for academic misconduct. Students found to have breached academic integrity may receive official warnings and/or serious academic penalties. Please read more about academic integrity [here](https://www.rmit.edu.au/students/my-course/assessment-results/academic-integrity). If you are unsure about any of these points or feel your assessment might breach academic integrity, please contact your course coordinator for support. It is important that you DO NOT submit any assessment until you can complete the declaration truthfully.

By checking the boxes below, I declare the following:

|  |  |
| --- | --- |
|  | I have not impersonated, or allowed myself to be impersonated by, any person for the purposes of this assessment |
|  | This assessment is my original work and no part of it has been copied from any other source except where due acknowledgement is made. |
|  | No part of this assessment has been written for me by any other person except where such collaboration has been authorised by the lecturer/teacher concerned. |
|  | I have not used generative “AI” tools for the purposes of this assessment. |
|  | Where this work is being submitted for individual assessment, I declare that it is my original work and that no part has been contributed by, produced by or in conjunction with another student. |
|  | I give permission for my assessment response to be reproduced, communicated, compared and archived for the purposes of detecting plagiarism. |
|  | I give permission for a copy of my assessment to be retained by the university for review and comparison, including review by external examiners. |

I understand that:

|  |  |
| --- | --- |
|  | Plagiarism is the presentation of the work, idea or creation of another person or machine as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to exclusion from the University. Plagiarised material can be drawn from, and presented in, written, graphic and visual form, including electronic data and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited. |
|  | Plagiarism includes the act of assisting or allowing another person to plagiarise or to copy my work. |

I agree and acknowledge that:

|  |  |
| --- | --- |
|  | I have read and understood the Declaration and Statement of Authorship above. |
|  | If I do not agree to the Declaration and Statement of Authorship in this context and all boxes are not checked, the assessment outcome is not valid for assessment purposes and will not be included in my final result for this course. |

## Assignment URL

## <https://01975f51-d7bd-c09b-0fd5-c683ab2c7169.share.connect.posit.cloud/>

## References

## 1.World Health Organization (WHO), & United Nations. (n.d.). Life expectancy and global health indicators (2000–2015) [Data set]. Open Data Bay. <https://www.opendatabay.com/data/government/9078a68f-26d0-4234-b538-08268271e3ac>

## Assignment Code

App.r

# Global Health Analytics Dashboard

# Student: Alve Reza Sazim

# Course: Data Vis with R

# Assignment 3

library(shiny)

library(plotly)

library(ggplot2)

library(dplyr)

library(tidyr)

library(DT)

library(leaflet)

library(sf)

library(maps)

library(mapdata)

library(viridis)

library(shinydashboard)

library(shinyWidgets)

# Screen dimensions (easy to modify)

SCREEN\_WIDTH <- 1920

SCREEN\_HEIGHT <- 1080

# Load and process data

data <- read.csv("lifeexpectancy\_processed.csv", stringsAsFactors = FALSE)

# Get world map data

world\_map <- map\_data("world")

# Country name mapping for consistency

country\_mapping <- c(

"United States of America" = "USA",

"United Kingdom of Great Britain and Northern Ireland" = "UK",

"Russian Federation" = "Russia",

"Republic of Korea" = "South Korea",

"Viet Nam" = "Vietnam",

"Bolivia (Plurinational State of)" = "Bolivia",

"Venezuela (Bolivarian Republic of)" = "Venezuela",

"Iran (Islamic Republic of)" = "Iran",

"United Republic of Tanzania" = "Tanzania",

"Syrian Arab Republic" = "Syria",

"Democratic People's Republic of Korea" = "North Korea",

"Congo" = "Republic of the Congo",

"Côte d'Ivoire" = "Ivory Coast",

"Republic of Moldova" = "Moldova",

"Czechia" = "Czech Republic",

"Lao People's Democratic Republic" = "Laos",

"The former Yugoslav republic of Macedonia" = "Macedonia",

"Trinidad and Tobago" = "Trinidad",

"Brunei Darussalam" = "Brunei",

"Cabo Verde" = "Cape Verde"

)

# Prepare data for analysis

country\_health\_stats <- data %>%

group\_by(country) %>%

summarise(

lifeexpectancy = round(mean(lifeexpectancy, na.rm = TRUE), 1),

adultmortality = round(mean(adultmortality, na.rm = TRUE), 1),

infant\_mortality\_rate = round(mean(infant\_mortality\_rate, na.rm = TRUE), 1),

bmi = round(mean(bmi, na.rm = TRUE), 1),

.groups = 'drop'

) %>%

mutate(region = ifelse(country %in% names(country\_mapping),

country\_mapping[country],

country))

# Merge with map data

map\_data\_merged <- world\_map %>%

left\_join(country\_health\_stats, by = "region")

# Check unmatched regions (debugging only)

# unmatched <- anti\_join(world\_map, country\_health\_stats, by = "region")

# print(unique(unmatched$region))

# Time series data

health\_intervention\_data <- data %>%

group\_by(year) %>%

summarise(

hepatitisb = round(mean(hepatitisb, na.rm = TRUE), 1),

measles = round(mean(measles, na.rm = TRUE), 0),

polio = round(mean(polio, na.rm = TRUE), 1),

diphtheria = round(mean(diphtheria, na.rm = TRUE), 1),

hivaids = round(mean(hivaids, na.rm = TRUE), 2),

.groups = 'drop'

)

development\_comparison <- data %>%

group\_by(year, status) %>%

summarise(avg\_life\_exp = round(mean(lifeexpectancy, na.rm = TRUE), 1), .groups = 'drop')

bmi\_health\_analysis <- data %>%

filter(!is.na(bmi\_category) & bmi\_category != "Unknown" & bmi\_category != "") %>%

group\_by(year, bmi\_category) %>%

summarise(

avg\_life\_exp = round(mean(lifeexpectancy, na.rm = TRUE), 1),

count = n(),

.groups = 'drop'

) %>%

filter(count >= 5)

# UI

ui <- dashboardPage(

dashboardHeader(title = "Global Health Analytics",

titleWidth = 300,

tags$li(class = "dropdown",

tags$style(HTML(paste0("

.main-header {

max-height: 50px !important;

}

.main-header .navbar {

margin-left: 300px;

height: 50px !important;

}

.main-header .logo {

height: 50px !important;

font-size: 18px !important;

line-height: 50px !important;

}

"))))),

dashboardSidebar(

width = 300,

collapsed = TRUE,

sidebarMenu(

id = "tabs",

menuItem("Data Exploration", tabName = "exploration", icon = icon("search")),

menuItem("Global Patterns", tabName = "map", icon = icon("globe")),

menuItem("Temporal Trends", tabName = "trends", icon = icon("chart-line"))

),

tags$style(HTML(paste0("

.main-sidebar {

height: ", SCREEN\_HEIGHT, "px !important;

}

.sidebar-menu li a {

font-size: 16px;

padding: 15px 5px 15px 15px;

}

")))

),

dashboardBody(

tags$head(

tags$style(HTML(paste0("

/\* Reset and base styles \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, sans-serif;

overflow: hidden;

background: #f5f5f5;

}

/\* Fixed dimensions \*/

.content-wrapper, .right-side {

height: ", SCREEN\_HEIGHT - 50, "px !important;

overflow: hidden !important;

}

.content {

padding: 15px !important;

height: 100% !important;

overflow-y: auto !important;

overflow-x: hidden !important;

}

/\* Typography \*/

h2 {

font-size: 26px;

font-weight: 300;

margin-bottom: 8px;

color: #333;

}

h3 {

font-size: 18px;

font-weight: 400;

margin: 15px 0 10px 0;

color: #555;

}

p {

font-size: 13px;

line-height: 1.5;

color: #666;

margin-bottom: 10px;

}

strong {

font-weight: 600;

color: #444;

}

/\* Containers \*/

.page-container {

height: 100%;

display: flex;

flex-direction: column;

padding: 0 15px;

overflow-y: auto;

overflow-x: hidden;

}

.viz-container {

flex: 1;

height: 100%;

margin: 0;

padding: 10px;

background: white;

border-radius: 6px;

box-shadow: 0 1px 3px rgba(0,0,0,0.1);

transition: all 0.3s ease;

overflow: hidden;

}

/\* Data table styles \*/

.dataTables\_wrapper {

font-size: 13px;

height: 100%;

overflow: hidden;

}

.dataTables\_scrollBody {

border: 1px solid #e0e0e0;

overflow-y: auto !important;

overflow-x: auto !important;

}

table.dataTable thead th {

background: #fafafa;

font-weight: 600;

text-align: left;

padding: 10px !important;

}

/\* Controls \*/

.control-group {

display: flex;

gap: 20px;

align-items: center;

margin-bottom: 15px;

}

.control-group label {

font-weight: 500;

color: #555;

min-width: 120px;

}

.selectize-control {

flex: 1;

max-width: 300px;

}

/\* Remove all box styling \*/

.box {

border: none !important;

box-shadow: none !important;

background: transparent !important;

}

.box-body {

padding: 0 !important;

}

/\* Subtle animations \*/

.viz-container {

transition: all 0.3s ease;

}

/\* Scrollbar styling \*/

::-webkit-scrollbar {

width: 8px;

height: 8px;

}

::-webkit-scrollbar-track {

background: #f1f1f1;

}

::-webkit-scrollbar-thumb {

background: #888;

border-radius: 4px;

}

::-webkit-scrollbar-thumb:hover {

background: #555;

}

")))

),

tabItems(

# Page 1: Data Exploration

tabItem(

tabName = "exploration",

div(class = "page-container", style = "display: flex; flex-direction: row; gap: 30px; align-items: center; justify-content: space-between;",

# Left Column - Text Content

div(style = "width: 40%; padding: 10px; overflow-y: auto; max-height: 100%;",

h3("A few days ago..."),

p(em("I got a call from home. My grandmother was sick. Really sick. She is about 75 years old and quite weak because of Diabetes, Bone decay and all the hard jobs she has done when she was younger. However, when I was in Hong Kong on my transit, I got a phone call that she is now in the ICU under life support. By the time I arrived, her condition was critical. She came back this time. This whole scenerio deeply moved me. Thats why the question I am looking the answer to is..")),

h2("What Determines How Long We Live?"),

p("Around the world, people are living longer than ever before — but what determines how long and how well we live? This dashboard explores global health data spanning from 2000 to 2015, sourced from the WHO and United Nations. The aim is to discover how variables like GDP, education, mortality, vaccination, and nutrition interact to shape life expectancy."),

h3("Understanding the Data Interface"),

p("Understanding global health outcomes requires more than just numbers — it requires ", em("context"), ". The Data Exploration interface allows users to interactively sift through a rich dataset of over 15 years, sourced from the World Health Organization and United Nations."),

p("This section enables deep dives into:"),

tags$ul(style = "font-size: 13px; line-height: 1.5; margin-left: 20px;",

tags$li(tags$strong("Trends in life expectancy")),

tags$li(tags$strong("Differences between developed and developing countries")),

tags$li(tags$strong("The influence of education, income, and healthcare on survival rates"))

),

p("With filters for country, development status, time range, and life expectancy bands, users can isolate specific factors and compare across regions. This targeted analysis helps uncover correlations and disparities often hidden in large datasets."),

p("The visualized metrics include:"),

tags$ul(style = "font-size: 13px; line-height: 1.5; margin-left: 20px; columns: 2; -webkit-columns: 2; -moz-columns: 2;",

tags$li("Life Expectancy"),

tags$li("Adult & Infant Mortality"),

tags$li("Education Levels"),

tags$li("GDP per Capita"),

tags$li("Alcohol Use"),

tags$li("BMI"),

tags$li("Vaccination Coverage"),

tags$li("Measles Incidence")

),

p("Whether you're investigating a single country's progress or contrasting multiple regions, this interface offers the flexibility needed for meaningful analysis.")

),

# Right Column - Data Table

div(style = "width: 55%; display: flex; align-items: center; justify-content: center;",

div(class = "viz-container", style = "width: 100%; overflow-x: auto;",

h3("Data Exploration Interface"),

p("Filter and explore the raw dataset by country, development status, and key health indicators:"),

fluidRow(style = "margin-bottom: 5px;",

column(3, selectInput("filter\_country", "Country:",

choices = c("All", unique(data$country)),

selected = "All", width = "100%")),

column(3, selectInput("filter\_status", "Development Status:",

choices = c("All", "Developed", "Developing"),

selected = "All", width = "100%")),

column(3, sliderInput("filter\_year", "Year Range:",

min = 2000, max = 2015,

value = c(2000, 2015), step = 1, width = "100%")),

column(3, sliderInput("filter\_life\_exp", "Life Expectancy:",

min = 35, max = 90,

value = c(35, 90), step = 1, width = "100%"))

),

DTOutput("data\_table", height = "450px")

)

)

)

),

# Page 2: World Map

tabItem(

tabName = "map",

div(class = "page-container",

h2("Health Outcomes Are Not Evenly Distributed"),

fluidRow(

column(

width = 5,

p("This map-based visualisation presents a global perspective on four key health indicators: Life Expectancy, Adult Mortality, Child Mortality, and Average BMI. The goal is to show how geography, wealth, and infrastructure play a vital role in human longevity."),

div(style = "margin: 15px 0;",

selectInput("map\_indicator", "Select Health Indicator:",

choices = list(

"Life Expectancy (Years)" = "lifeexpectancy",

"Adult Mortality (per 1000)" = "adultmortality",

"Child Mortality (per 1000)" = "infant\_mortality\_rate",

"Average BMI" = "bmi"

),

selected = "lifeexpectancy", width = "100%")

),

h3("Visualisation Explanation"),

p("Each map uses a choropleth colour scale, where darker shades represent higher values. Toggle between the four metrics to compare regional patterns."),

p(em("This visual representation helps uncover the intersection of public health and geopolitics. Nations with stable governance, robust healthcare infrastructure, and higher investment in education often score better in life expectancy and BMI metrics. Meanwhile, regions affected by conflict or economic disparity tend to experience higher mortality rates.")),

p(em("The stark differences highlighted in this visualization align closely with Sustainable Development Goal 3 (Good Health and Well-being), emphasizing the need for equitable access to health services globally. This map not only reveals disparities but also serves as a call for targeted interventions.")),

h3("Observations and Patterns"),

tags$ul(style = "font-size: 13px; line-height: 1.5;",

tags$li(tags$strong("Life Expectancy:"), " Peaks in Western Europe, Canada, Australia, and parts of Asia; lowest in parts of Central and Sub-Saharan Africa."),

tags$li(tags$strong("Adult Mortality:"), " Starkly high in war-torn and low-income countries, especially where healthcare access is minimal."),

tags$li(tags$strong("Child Mortality:"), " Closely follows patterns of poverty and instability."),

tags$li(tags$strong("BMI:"), " Developed countries show higher average BMI; emerging economies are now seeing rises.")

),

p("These maps highlight both progress and persistent inequality. Even as global life expectancy increases, many countries are still burdened by preventable deaths.")

),

column(

width = 7,

div(class = "viz-container", style = "height: 100%; min-height: 600px; padding: 10px;",

plotlyOutput("world\_map", height = "550px")

)

)

)

)

),

# Page 3: Temporal Trends

tabItem(

tabName = "trends",

div(class = "page-container",

h2("A Timeline of Transformation"),

p("How have things changed over time? This final tab zooms in on longitudinal trends. Explore the relationship between BMI and longevity, the closing development gap, and the impact of public health interventions."),

fluidRow(

column(5,

div(class = "control-group", style = "margin-bottom: 15px;",

selectInput("trend\_analysis", "Analysis Focus:",

choices = list(

"Disease Prevention Progress" = "disease",

"Development Gap Analysis" = "dev\_vs\_developing",

"BMI and Longevity Relationship" = "bmi\_life"

),

selected = "disease", width = "100%"),

conditionalPanel(

condition = "input.trend\_analysis == 'disease'",

selectInput("intervention\_type", "Health Intervention:",

choices = list(

"Hepatitis B Vaccination" = "hepatitisb",

"Measles Control" = "measles",

"Polio Eradication" = "polio",

"Diphtheria Prevention" = "diphtheria",

"HIV/AIDS Response" = "hivaids"

),

selected = "hepatitisb", width = "100%")

)

),

div(style = "margin-top: 20px;",

h3("Visualisation Breakdown"),

tags$ul(style = "font-size: 13px; line-height: 1.5;",

tags$li(tags$strong("BMI vs Life Expectancy:"), " Shows that individuals with a 'Normal' BMI live shorter lives than those classified as overweight — reflecting a paradox observed in many epidemiological studies."),

tags$li(tags$strong("Developed vs Developing Nations:"), " While life expectancy is rising globally, the gap between rich and poor countries remains wide — narrowing only slowly."),

tags$li(tags$strong("Vaccination Trends:"), " The increase in coverage for hepatitis B, polio, and measles maps almost directly to improvements in life expectancy.")

),

h3("Analysis"),

p("The data tells a hopeful, yet complex story. Medical advancements and global health initiatives have clearly improved outcomes — but inequity persists. This dashboard demonstrates how data visualisation can illuminate both success and struggle."),

)

),

column(7,

div(class = "viz-container", style = "height: 450px;",

plotlyOutput("trends\_chart", height = "100%")

)

)

),

div(style = "margin-top: 40px; width: 100%;",

h3("Conclusion"),

p("Ten years ago, critical care like life support, ICU admission, or even timely diagnosis would have been out of reach for families like mine—financially, technologically, and geographically. Today, my grandmother’s survival was not just a miracle, but a testament to how far medical science and public health have come. We are witnessing a revolution—where once-impossible treatments are becoming routine, and where hope is no longer a privilege of the wealthy."),

p("She is still here because the systems around her improved—because researchers pushed boundaries, because data illuminated needs, and because someone, somewhere, used insights like those in this dashboard to plan smarter interventions. It made me realize that behind every row in a dataset is a life that could be saved. That’s why I built this dashboard—not just to analyze numbers, but to recognize lives, like hers, that deserve a chance.")

)

)

)

)

)

)

# Server

server <- function(input, output, session) {

# Reactive filtered data

filtered\_data <- reactive({

df <- data

if (input$filter\_country != "All") {

df <- df %>% filter(country == input$filter\_country)

}

if (input$filter\_status != "All") {

df <- df %>% filter(status == input$filter\_status)

}

df <- df %>%

filter(year >= input$filter\_year[1] & year <= input$filter\_year[2]) %>%

filter(lifeexpectancy >= input$filter\_life\_exp[1] & lifeexpectancy <= input$filter\_life\_exp[2])

df

})

# Data table output

output$data\_table <- renderDT({

df <- filtered\_data() %>%

select(

Country = country,

Year = year,

Status = status,

`Life Expectancy` = lifeexpectancy,

`Adult Mortality` = adultmortality,

`Infant Mortality` = infant\_mortality\_rate,

`GDP per Capita` = gdp,

`Education (Years)` = schooling,

`BMI` = bmi,

`Alcohol Consumption` = alcohol,

`Hepatitis B Coverage` = hepatitisb,

`Measles Cases` = measles

)

datatable(

df,

options = list(

scrollY = "350px",

scrollX = TRUE,

paging = FALSE,

searching = TRUE,

dom = 't',

columnDefs = list(

list(className = 'dt-left', targets = 0:2),

list(className = 'dt-center', targets = 3:11)

)

),

class = 'compact stripe hover',

rownames = FALSE,

filter = 'top'

) %>%

formatRound(columns = c('Life Expectancy', 'GDP per Capita', 'BMI',

'Alcohol Consumption', 'Education (Years)'), digits = 1) %>%

formatPercentage(columns = c('Hepatitis B Coverage'), digits = 0)

})

# World map with zoom capability

output$world\_map <- renderPlotly({

selected\_metric <- input$map\_indicator

# Color configuration - fixed ranges for all data

color\_config <- switch(selected\_metric,

"lifeexpectancy" = list(

colors = c("white", "#2d8f2d"),

title = "Life Expectancy<br>(Years)",

range = c(40, 90)

),

"adultmortality" = list(

colors = c("white", "#d32f2f"),

title = "Adult Mortality<br>(per 1000)",

range = c(0, 700)

),

"infant\_mortality\_rate" = list(

colors = c("white", "#ff5722"),

title = "Infant Mortality<br>(per 1000)",

range = c(0, 120)

),

"bmi" = list(

colors = c("white", "#7b1fa2"),

title = "Average BMI<br>(kg/m²)",

range = range(map\_data\_merged$bmi, na.rm = TRUE)

)

)

plot\_data <- map\_data\_merged %>%

mutate(display\_label = paste0(

region, "<br>",

switch(input$map\_indicator,

"lifeexpectancy" = "Life Expectancy: ",

"adultmortality" = "Adult Mortality: ",

"infant\_mortality\_rate" = "Infant Mortality: ",

"bmi" = "Average BMI: "),

round(get(input$map\_indicator), 1)

))

# Create plot with border styling

p <- ggplot(plot\_data, aes(x = long, y = lat, group = group)) +

geom\_polygon(aes\_string(fill = selected\_metric, text = "display\_label"), color = "#333333", size = 0.2) +

scale\_fill\_gradient(

low = color\_config$colors[1],

high = color\_config$colors[2],

limits = color\_config$range,

na.value = "#e0e0e0",

name = color\_config$title

) +

theme\_void() +

theme(

legend.position = "right",

legend.title = element\_text(size = 10),

legend.text = element\_text(size = 9),

plot.background = element\_rect(fill = "white", color = "#cccccc", size = 1),

panel.background = element\_rect(fill = "#f5f5f5", color = NA)

) +

coord\_fixed(1.3)

ggplotly(p, tooltip = "text", height = 550) %>%

layout(

hoverlabel = list(bgcolor = "white", font = list(size = 12)),

margin = list(l = 10, r = 10, t = 10, b = 10),

dragmode = "zoom",

showlegend = TRUE

) %>%

config(

displayModeBar = TRUE,

modeBarButtonsToRemove = c("select2d", "lasso2d", "autoScale2d"),

displaylogo = FALSE

)

})

# Trends chart

output$trends\_chart <- renderPlotly({

req(input$trend\_analysis)

# Create the base plot based on analysis type

plot\_result <- if (input$trend\_analysis == "disease") {

req(input$intervention\_type)

selected\_intervention <- input$intervention\_type

y\_label <- switch(selected\_intervention,

"hepatitisb" = "Hepatitis B Coverage (%)",

"measles" = "Measles Cases",

"polio" = "Polio Coverage (%)",

"diphtheria" = "Diphtheria Coverage (%)",

"hivaids" = "HIV/AIDS Deaths (per 100,000)"

)

plot\_ly(health\_intervention\_data,

x = ~year,

y = ~get(selected\_intervention),

type = 'scatter',

mode = 'lines+markers',

line = list(color = '#2e7d32', width = 3),

marker = list(color = '#2e7d32', size = 8),

hovertemplate = paste0("<b>", y\_label, ": %{y}</b><br>Year: %{x}<extra></extra>")) %>%

layout(

xaxis = list(title = "Year", gridcolor = "#f0f0f0"),

yaxis = list(title = y\_label, gridcolor = "#f0f0f0"),

plot\_bgcolor = "#fafafa",

paper\_bgcolor = "white",

legend = list(x = 0.02, y = 0.98),

margin = list(l = 60, r = 20, t = 20, b = 60),

hovermode = 'closest'

)

} else if (input$trend\_analysis == "dev\_vs\_developing") {

plot\_ly() %>%

add\_trace(data = development\_comparison %>% filter(status == "Developed"),

x = ~year, y = ~avg\_life\_exp, name = "Developed Countries",

type = 'scatter', mode = 'lines+markers',

line = list(color = '#1976d2', width = 3),

marker = list(color = '#1976d2', size = 8)) %>%

add\_trace(data = development\_comparison %>% filter(status == "Developing"),

x = ~year, y = ~avg\_life\_exp, name = "Developing Countries",

type = 'scatter', mode = 'lines+markers',

line = list(color = '#d32f2f', width = 3),

marker = list(color = '#d32f2f', size = 8)) %>%

layout(

xaxis = list(title = "Year", gridcolor = "#f0f0f0"),

yaxis = list(title = "Life Expectancy (Years)", gridcolor = "#f0f0f0"),

plot\_bgcolor = "#fafafa",

paper\_bgcolor = "white",

legend = list(x = 0.02, y = 0.98),

margin = list(l = 60, r = 20, t = 20, b = 60),

hovermode = 'closest'

)

} else {

# BMI analysis

bmi\_colors <- c("Normal" = "#4caf50", "Underweight" = "#ff9800",

"Overweight" = "#2196f3", "Obese" = "#9c27b0")

plot\_data <- plot\_ly()

for (bmi\_cat in names(bmi\_colors)) {

category\_data <- bmi\_health\_analysis %>% filter(bmi\_category == bmi\_cat)

if (nrow(category\_data) > 0) {

plot\_data <- plot\_data %>% add\_trace(

data = category\_data,

x = ~year,

y = ~avg\_life\_exp,

name = bmi\_cat,

type = 'scatter',

mode = 'lines+markers',

line = list(color = bmi\_colors[[bmi\_cat]], width = 3),

marker = list(color = bmi\_colors[[bmi\_cat]], size = 8)

)

}

}

plot\_data %>%

layout(

xaxis = list(title = "Year", gridcolor = "#f0f0f0"),

yaxis = list(title = "Life Expectancy (Years)", gridcolor = "#f0f0f0"),

plot\_bgcolor = "#fafafa",

paper\_bgcolor = "white",

legend = list(x = 0.02, y = 0.98),

margin = list(l = 60, r = 20, t = 20, b = 60),

hovermode = 'closest'

)

}

# Apply config to the final plot

plot\_result %>% config(displayModeBar = FALSE)

})

}

# Run the app

shinyApp(ui = ui, server = server)  
  
  
Pre processing Code:

library(dplyr)

library(tidyr)

data <- read.csv("Life Expectancy Data.csv", stringsAsFactors = FALSE)

names(data) <- gsub(" ", "", names(data))

names(data) <- gsub("-", "", names(data))

names(data) <- gsub("/", "", names(data))

names(data) <- tolower(names(data))

# Check for missing population data

data$population\_missing <- is.na(data$population) | data$population == 0

# Calculate infant mortality rate (per 1000 live births)

data$infant\_mortality\_rate <- ifelse(

!is.na(data$infantdeaths) & !is.na(data$population) & data$population > 0,

(data$infantdeaths / data$population) \* 1000,

NA

)

# Calculate under-5 mortality rate (per 1000 live births)

data$under5\_mortality\_rate <- ifelse(

!is.na(data$underfivedeaths) & !is.na(data$population) & data$population > 0,

(data$underfivedeaths / data$population) \* 1000,

NA

)

# Calculate health expenditure per capita

data$health\_exp\_per\_capita <- ifelse(

!is.na(data$percentageexpenditure) & !is.na(data$gdp) & data$gdp > 0,

(data$percentageexpenditure / 100) \* data$gdp,

NA

)

# Normalize key indicators and create composite score

data <- data %>%

mutate(

# Normalize each indicator to 0-1 scale

norm\_life\_exp = (lifeexpectancy - min(lifeexpectancy, na.rm = TRUE)) /

(max(lifeexpectancy, na.rm = TRUE) - min(lifeexpectancy, na.rm = TRUE)),

norm\_gdp = (gdp - min(gdp, na.rm = TRUE)) /

(max(gdp, na.rm = TRUE) - min(gdp, na.rm = TRUE)),

norm\_schooling = (schooling - min(schooling, na.rm = TRUE)) /

(max(schooling, na.rm = TRUE) - min(schooling, na.rm = TRUE)),

norm\_income\_comp = (incomecompositionofresources - min(incomecompositionofresources, na.rm = TRUE)) /

(max(incomecompositionofresources, na.rm = TRUE) - min(incomecompositionofresources, na.rm = TRUE)),

# Calculate development score as average of normalized indicators

development\_score = rowMeans(

cbind(norm\_life\_exp, norm\_gdp, norm\_schooling, norm\_income\_comp),

na.rm = TRUE

)

) %>%

select(-starts\_with("norm\_"))

# BMI categories based on WHO standards

data$bmi\_category <- cut(

data$bmi,

breaks = c(-Inf, 18.5, 25, 30, Inf),

labels = c("Underweight", "Normal", "Overweight", "Obese"),

include.lowest = TRUE

)

data$bmi\_category[is.na(data$bmi)] <- "Unknown"

# Year groups for temporal analysis

data$year\_group <- cut(

data$year,

breaks = c(1999, 2005, 2010, 2015),

labels = c("2000-2005", "2006-2010", "2011-2015"),

include.lowest = TRUE

)

# Life expectancy categories

data$life\_exp\_category <- cut(

data$lifeexpectancy,

breaks = c(-Inf, 60, 70, 80, Inf),

labels = c("Low (<60)", "Medium (60-70)", "High (70-80)", "Very High (>80)"),

include.lowest = TRUE

)

# Create a simplified region mapping based on country names

data$region <- case\_when(

data$country %in% c("United States of America", "Canada", "Mexico",

"Brazil", "Argentina", "Chile", "Colombia", "Peru",

"Venezuela", "Ecuador", "Bolivia", "Paraguay", "Uruguay",

"Guatemala", "Cuba", "Haiti", "Dominican Republic",

"El Salvador", "Honduras", "Nicaragua", "Costa Rica",

"Panama", "Jamaica", "Trinidad and Tobago", "Barbados",

"Saint Lucia", "Saint Vincent and the Grenadines",

"Grenada", "Antigua and Barbuda", "Guyana", "Suriname",

"Belize", "Bahamas") ~ "Americas",

data$country %in% c("United Kingdom", "Germany", "France", "Italy",

"Spain", "Poland", "Romania", "Netherlands", "Belgium",

"Greece", "Portugal", "Czech Republic", "Hungary",

"Sweden", "Austria", "Switzerland", "Bulgaria", "Denmark",

"Finland", "Slovakia", "Norway", "Ireland", "Croatia",

"Bosnia and Herzegovina", "Albania", "Lithuania", "Slovenia",

"Latvia", "Estonia", "Macedonia", "Moldova", "Luxembourg",

"Malta", "Iceland", "Montenegro", "Serbia", "Cyprus") ~ "Europe",

data$country %in% c("China", "India", "Indonesia", "Pakistan", "Bangladesh",

"Japan", "Philippines", "Vietnam", "Turkey", "Iran",

"Thailand", "Myanmar", "South Korea", "Iraq", "Afghanistan",

"Saudi Arabia", "Uzbekistan", "Malaysia", "Nepal", "Yemen",

"North Korea", "Sri Lanka", "Cambodia", "Jordan", "Azerbaijan",

"United Arab Emirates", "Tajikistan", "Israel", "Laos",

"Lebanon", "Singapore", "Oman", "Kuwait", "Georgia",

"Mongolia", "Armenia", "Qatar", "Bahrain", "Timor-Leste",

"Maldives", "Brunei", "Bhutan") ~ "Asia",

data$country %in% c("Nigeria", "Ethiopia", "Egypt", "Democratic Republic of Congo",

"South Africa", "Kenya", "Uganda", "Algeria", "Sudan",

"Morocco", "Angola", "Ghana", "Mozambique", "Madagascar",

"Cameroon", "Ivory Coast", "Niger", "Burkina Faso", "Mali",

"Malawi", "Zambia", "Senegal", "Somalia", "Chad", "Zimbabwe",

"Guinea", "Rwanda", "Benin", "Tunisia", "Burundi", "South Sudan",

"Togo", "Sierra Leone", "Libya", "Liberia", "Mauritania",

"Eritrea", "Gambia", "Botswana", "Namibia", "Gabon",

"Lesotho", "Guinea-Bissau", "Equatorial Guinea", "Mauritius",

"Swaziland", "Djibouti", "Comoros", "Cape Verde",

"Sao Tome and Principe", "Seychelles") ~ "Africa",

data$country %in% c("Australia", "Papua New Guinea", "New Zealand", "Fiji",

"Solomon Islands", "Vanuatu", "Samoa", "Kiribati",

"Micronesia", "Tonga", "Palau", "Cook Islands", "Nauru",

"Tuvalu", "Marshall Islands") ~ "Oceania",

TRUE ~ "Other"

)

# Ensure numeric columns are properly typed

numeric\_cols <- c("lifeexpectancy", "adultmortality", "infantdeaths", "alcohol",

"percentageexpenditure", "hepatitisb", "measles", "bmi",

"underfivedeaths", "polio", "totalexpenditure", "diphtheria",

"hivaids", "gdp", "population", "thinness119years",

"thinness59years", "incomecompositionofresources", "schooling",

"infant\_mortality\_rate", "under5\_mortality\_rate",

"health\_exp\_per\_capita", "development\_score")

for(col in numeric\_cols) {

if(col %in% names(data)) {

data[[col]] <- as.numeric(data[[col]])

}

}

# Ensure integer columns

data$adultmortality <- as.integer(data$adultmortality)

data$infantdeaths <- as.integer(data$infantdeaths)

data$measles <- as.integer(data$measles)

data$underfivedeaths <- as.integer(data$underfivedeaths)

data <- data %>%

select(

country, year, status, lifeexpectancy, adultmortality, infantdeaths,

alcohol, percentageexpenditure, hepatitisb, measles, bmi, underfivedeaths,

polio, totalexpenditure, diphtheria, hivaids, gdp, population,

thinness119years, thinness59years, incomecompositionofresources, schooling,

population\_missing, infant\_mortality\_rate, under5\_mortality\_rate,

health\_exp\_per\_capita, development\_score, bmi\_category, year\_group,

life\_exp\_category, region

)

write.csv(data, "lifeexpectancy\_processed.csv", row.names = FALSE)