MATH2270/MATH2237/MATH2404 Assignment 3

Name
Samay Jain

Student number
S3963844

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. 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.
- ☒ 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 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://samayjain.shinyapps.io/Data_Vis_Assgn_3/

References

Australian Energy Statistics 2023 Table B. Available at: https://www.energy.gov.au/publications/australian-energy-update-2023 (Accessed: 23 October 2023).

Line plots in R. Available at: https://plotly.com/r/line-charts/ (Accessed: 24 October 2023).

Welcome to Shiny. Available at: https://shiny.posit.co/r/getstarted/shiny-basics/lesson1/index.html (Accessed: 24 October 2023).

Assignment Code

```
# Calling the libraries

library(shiny)

library(ggplot2)

library(plotly)

library(zoo)

library(stringr)

library(tidyr)

library(readxl)

library(dplyr)

library(scales)

library(RColorBrewer)

library(rsconnect)

library(readr)
```

```
# Data Preprocessing

setwd("C:/Users/HP/Downloads/Data_Vis_Assgn_3")

energy_data <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 3,range = 'B5:H68')

energy_data1 <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 4,range = 'B5:H68')

energy_data2 <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 5,range = 'B5:H68')

energy_data3 <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 6,range = 'B5:H68')

energy_data4 <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 7,range = 'B5:H68')

energy_data5 <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 8,range = 'B5:H68')

energy_data6 <- read_xlsx("Australian Energy Statistics 2023 Table B.xlsx", sheet = 9,range = 'B5:H68')

colnames(energy_data)[1] <- "Year"

colnames(energy_data)[2] <- "Population"

colnames(energy_data)[3] <- "GSP in Millions"

colnames(energy_data)[4] <- "Energy consumption (PJ)"

colnames(energy_data)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data)[7] <- "Energy productivity ($ Million/PJ)"

energy_data <- energy_data[-(1:30),]

str(energy_data)

energy_data$State <- "NSW"

energy_data$Year <- substr(energy_data$Year, start = 1, stop = 4)

energy_data$Year <- as.double(energy_data$Year)

energy_data$Population <- as.double(energy_data$Population)

energy_data$`GSP in Millions` <- as.double(energy_data$`GSP in Millions`)

energy_data$`Energy consumption (PJ)` <- as.double(energy_data$`Energy consumption (PJ)`)

energy_data$`Energy consumption (GJ/person)` <- as.double(energy_data$`Energy consumption
(GJ/person)`)

energy_data$`Enery Intensity (GJ/$Million)` <- as.double(energy_data$`Enery Intensity (GJ/$Million)`)
```

```
energy_data$`Energy productivity ($ Million/PJ)` <- as.double(energy_data$`Energy productivity ($ Million/PJ)`)

energy_data[is.na(energy_data)] <- 0


colnames(energy_data1)[1] <- "Year"

colnames(energy_data1)[2] <- "Population"

colnames(energy_data1)[3] <- "GSP in Millions"

colnames(energy_data1)[4] <- "Energy consumption (PJ)"

colnames(energy_data1)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data1)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data1)[7] <- "Energy productivity ($ Million/PJ)"

energy_data1 <- energy_data1[-(1:30),]

str(energy_data1)

energy_data1$State <- "VIC"

energy_data1$Year <- substr(energy_data1$Year, start = 1, stop = 4)

energy_data1$Year <- as.double(energy_data1$Year)

energy_data1$Population <- as.double(energy_data1$Population)

energy_data1$`GSP in Millions` <- as.double(energy_data1$`GSP in Millions`)

energy_data1$`Energy consumption (PJ)` <- as.double(energy_data1$`Energy consumption (PJ)`)

energy_data1$`Energy consumption (GJ/person)` <- as.double(energy_data1$`Energy consumption (GJ/person)`)

energy_data1$`Enery Intensity (GJ/$Million)` <- as.double(energy_data1$`Enery Intensity (GJ/$Million)`)

energy_data1$`Energy productivity ($ Million/PJ)` <- as.double(energy_data1$`Energy productivity ($ Million/PJ)`)

energy_data1[is.na(energy_data1)] <- 0


colnames(energy_data2)[1] <- "Year"

colnames(energy_data2)[2] <- "Population"

colnames(energy_data2)[3] <- "GSP in Millions"

colnames(energy_data2)[4] <- "Energy consumption (PJ)"
```

```
colnames(energy_data2)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data2)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data2)[7] <- "Energy productivity ($ Million/PJ)"

energy_data2 <- energy_data2[-(1:30),]

str(energy_data2)

energy_data2$State <- "QLD"

energy_data2$Year <- substr(energy_data2$Year, start = 1, stop = 4)

energy_data2$Year <- as.double(energy_data2$Year)

energy_data2$Population <- as.double(energy_data2$Population)

energy_data2$`GSP in Millions` <- as.double(energy_data2$`GSP in Millions`)

energy_data2$`Energy consumption (PJ)` <- as.double(energy_data2$`Energy consumption (PJ)`)

energy_data2$`Energy consumption (GJ/person)` <- as.double(energy_data2$`Energy consumption
(GJ/person)`)

energy_data2$`Enery Intensity (GJ/$Million)` <- as.double(energy_data2$`Enery Intensity (GJ/$Million)`)

energy_data2$`Energy productivity ($ Million/PJ)` <- as.double(energy_data2$`Energy productivity ($
Million/PJ)`)

energy_data2[is.na(energy_data2)] <- 0


colnames(energy_data3)[1] <- "Year"

colnames(energy_data3)[2] <- "Population"

colnames(energy_data3)[3] <- "GSP in Millions"

colnames(energy_data3)[4] <- "Energy consumption (PJ)"

colnames(energy_data3)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data3)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data3)[7] <- "Energy productivity ($ Million/PJ)"

energy_data3 <- energy_data3[-(1:30),]

str(energy_data3)

energy_data3$State <- "SA"

energy_data3$Year <- substr(energy_data3$Year, start = 1, stop = 4)
```

```
energy_data3$Year <- as.double(energy_data3$Year)

energy_data3$Population <- as.double(energy_data3$Population)

energy_data3$`GSP in Millions` <- as.double(energy_data3$`GSP in Millions`)

energy_data3$`Energy consumption (PJ)` <- as.double(energy_data3$`Energy consumption (PJ)`)

energy_data3$`Energy consumption (GJ/person)` <- as.double(energy_data3$`Energy consumption (GJ/person)`)

energy_data3$`Enery Intensity (GJ/$Million)` <- as.double(energy_data3$`Enery Intensity (GJ/$Million)`)

energy_data3$`Energy productivity ($ Million/PJ)` <- as.double(energy_data3$`Energy productivity ($ Million/PJ)`)

energy_data3[is.na(energy_data3)] <- 0


colnames(energy_data4)[1] <- "Year"

colnames(energy_data4)[2] <- "Population"

colnames(energy_data4)[3] <- "GSP in Millions"

colnames(energy_data4)[4] <- "Energy consumption (PJ)"

colnames(energy_data4)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data4)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data4)[7] <- "Energy productivity ($ Million/PJ)"

energy_data4 <- energy_data4[-(1:30),]

str(energy_data4)

energy_data4$State <- "WA"

energy_data4$Year <- substr(energy_data4$Year, start = 1, stop = 4)

energy_data4$Year <- as.double(energy_data4$Year)

energy_data4$Population <- as.double(energy_data4$Population)

energy_data4$`GSP in Millions` <- as.double(energy_data4$`GSP in Millions`)

energy_data4$`Energy consumption (PJ)` <- as.double(energy_data4$`Energy consumption (PJ)`)

energy_data4$`Energy consumption (GJ/person)` <- as.double(energy_data4$`Energy consumption (GJ/person)`)

energy_data4$`Enery Intensity (GJ/$Million)` <- as.double(energy_data4$`Enery Intensity (GJ/$Million)`)
```

```
energy_data4$`Energy productivity ($ Million/PJ)` <- as.double(energy_data4$`Energy productivity ($ Million/PJ)`)

energy_data4[is.na(energy_data4)] <- 0


colnames(energy_data5)[1] <- "Year"

colnames(energy_data5)[2] <- "Population"

colnames(energy_data5)[3] <- "GSP in Millions"

colnames(energy_data5)[4] <- "Energy consumption (PJ)"

colnames(energy_data5)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data5)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data5)[7] <- "Energy productivity ($ Million/PJ)"

energy_data5 <- energy_data5[-(1:30),]

str(energy_data5)

energy_data5$State <- "TAS"

energy_data5$Year <- substr(energy_data5$Year, start = 1, stop = 4)

energy_data5$Year <- as.double(energy_data5$Year)

energy_data5$Population <- as.double(energy_data5$Population)

energy_data5$`GSP in Millions` <- as.double(energy_data5$`GSP in Millions`)

energy_data5$`Energy consumption (PJ)` <- as.double(energy_data5$`Energy consumption (PJ)`)

energy_data5$`Energy consumption (GJ/person)` <- as.double(energy_data5$`Energy consumption (GJ/person)`)

energy_data5$`Enery Intensity (GJ/$Million)` <- as.double(energy_data5$`Enery Intensity (GJ/$Million)`)

energy_data5$`Energy productivity ($ Million/PJ)` <- as.double(energy_data5$`Energy productivity ($ Million/PJ)`)

energy_data5[is.na(energy_data5)] <- 0


colnames(energy_data6)[1] <- "Year"

colnames(energy_data6)[2] <- "Population"

colnames(energy_data6)[3] <- "GSP in Millions"

colnames(energy_data6)[4] <- "Energy consumption (PJ)"
```

```
colnames(energy_data6)[5] <- "Energy consumption (GJ/person)"

colnames(energy_data6)[6] <- "Enery Intensity (GJ/$Million)"

colnames(energy_data6)[7] <- "Energy productivity ($ Million/PJ)"

energy_data6 <- energy_data6[-(1:30),]

str(energy_data6)

energy_data6$State <- "NT"

energy_data6$Year <- substr(energy_data6$Year, start = 1, stop = 4)

energy_data6$Year <- as.double(energy_data6$Year)

energy_data6$Population <- as.double(energy_data6$Population)

energy_data6$`GSP in Millions` <- as.double(energy_data6$`GSP in Millions`)

energy_data6$`Energy consumption (PJ)` <- as.double(energy_data6$`Energy consumption (PJ)`)

energy_data6$`Energy consumption (GJ/person)` <- as.double(energy_data6$`Energy consumption
(GJ/person)`)

energy_data6$`Enery Intensity (GJ/$Million)` <- as.double(energy_data6$`Enery Intensity (GJ/$Million)`)

energy_data6$`Energy productivity ($ Million/PJ)` <- as.double(energy_data6$`Energy productivity ($
Million/PJ)`)

energy_data6[is.na(energy_data6)] <- 0


final_data <- rbind(energy_data, energy_data1, energy_data2, energy_data3, energy_data4, energy_data5,
energy_data6)

# Data preprocessing ends here .....

final_data <- read.csv('final_data.csv')

# Define UI for application that draws a histogram

ui <- fluidPage(

  tags$head(

    tags$style(

      HTML("

        .footer {

          position: fixed;

          bottom: 0;
```

```
        right: 0;
        width: 100%;
        padding: 10px;
        font-size: 10px;
        text-align: right;
        z-index: 100;
      }
    ")
  )
),
titlePanel("Energy Trends and Insights: Australian State Analysis (1989-2021)"),
mainPanel(
  tabsetPanel(
    tabPanel("Energy Consumption",
             plotlyOutput("lineplot")),
    tabPanel("Population v/s energy consumption/capita",
             plotlyOutput("st_population", width = "1000px", height = "300px"),
             plotlyOutput("consump_per_capita", width = "1000px", height = "300px")),
    tabPanel("Energy Productivity and Intensity",
             plotlyOutput("Energy_productivity", width = "1000px", height = "300px"),
             plotlyOutput("Energy_intensity", width = "1000px", height = "300px"),
             div(class = "footer",
                 p(
                   "References:",
                   br(),
                   "Australian Energy Statistics 2023 Table B. Available at:
https://www.energy.gov.au/publications/australian-energy-update-2023 (Accessed: 23 October 2023).",
                   br(),
                   "Line plots in R. Available at: https://plotly.com/r/line-charts/ (Accessed: 24 October 2023).",
```

```
                br(),
                "Welcome to Shiny. Available at: https://shiny.posit.co/r/getstarted/shiny-basics/lesson1/index.html (Accessed: 24 October 2023)."
              )
            )
          )
        )
      )
    )

server <- function(input, output) {
  output$lineplot <- renderPlotly({
    updatemenus <- list(
      list(
        active = 0,
        x = -.3,
        type= 'buttons',
        buttons = list(
          list(
            label = "NSW",
            method = "update",
            args = list(list(visible = c(TRUE, "legendonly", "legendonly", "legendonly", "legendonly", "legendonly", "legendonly")))),
          list(
            label = "VIC",
            method = "update",
            args = list(list(visible = c("legendonly", "legendonly", "legendonly", "legendonly", "legendonly",TRUE, "legendonly")))),
          list(
            label = "QLD",
```

```
          method = "update",

          args = list(list(visible = c("legendonly", "legendonly", TRUE,"legendonly", "legendonly",
"legendonly", "legendonly")))),

        list(

          label = "SA",

          method = "update",

          args = list(list(visible = c("legendonly", "legendonly", "legendonly", TRUE, "legendonly",
"legendonly", "legendonly")))),

        list(

          label = "WA",

          method = "update",

          args = list(list(visible = c("legendonly", "legendonly", "legendonly", "legendonly", "legendonly",
"legendonly", TRUE)))),

        list(

          label = "TAS",

          method = "update",

          args = list(list(visible = c("legendonly", "legendonly", "legendonly", "legendonly", TRUE,
"legendonly","legendonly")))),

        list(

          label = "NT",

          method = "update",

          args = list(list(visible = c("legendonly", TRUE, "legendonly", "legendonly", "legendonly",
"legendonly", "legendonly"))))

      )

    )

  )

  plot_ <- plot_ly(data = final_data , x = final_data$Year, y = final_data$Energy_consumption_.PJ., color
= final_data$State, type = 'scatter', mode = 'line') %>%

    layout(

      title = "State wise energy consumption during 1989-2021",

      xaxis = list(zeroline = FALSE, title = "Year", tickangle = -45),
```

```
      yaxis = list(zeroline = FALSE, title = "Energy Consumption(PJ)"),
      updatemenus=updatemenus)
  plot_
})
output$st_population <- renderPlotly({
  plot1 <- plot_ly(data = final_data , x = final_data$Year, y = final_data$Population, color = final_data$State, type = 'scatter', mode = 'line') %>%
    layout(
      title = "Population in different states during 1989-2021",
      xaxis = list(zeroline = FALSE, title = "Year", tickangle = -45),
      yaxis = list(zeroline = FALSE, title = "Population"),
      updatemenus=updatemenus)
  plot1
})
output$consump_per_capita <- renderPlotly({
  plot2 <- plot_ly(data = final_data , x = final_data$Year, y = final_data$Energy_consumption_.GJ.person., color = final_data$State, type = 'scatter', mode = 'line') %>%
    layout(
      title = "Energy consumption per capita during 1989-2021",
      xaxis = list(zeroline = FALSE, title = "Year", tickangle = -45),
      yaxis = list(zeroline = FALSE, title = "Energy consumption(GJ/person)"),
      updatemenus=updatemenus)
  plot2
})
output$Energy_productivity <- renderPlotly({
  plot3 <- plot_ly(data = final_data , x = final_data$Year, y = final_data$Energy_productivity_...Million.PJ., color = final_data$State, type = 'scatter', mode = 'line') %>%
    layout(
      title = "Changes in Energy productivity during 1989-2021",
```

```
      xaxis = list(zeroline = FALSE, title = "Year", tickangle = -45),
      yaxis = list(zeroline = FALSE, title = "Energy productivity($Million/PJ)"),
      updatemenus=updatemenus)
    plot3
  })
  output$Energy_intensity <- renderPlotly({
    plot4 <- plot_ly(data = final_data , x = final_data$Year, y = final_data$Enery_Intensity_.GJ..Million.,
color = final_data$State, type = 'scatter', mode = 'line') %>%
      layout(
      title = "Changes in Energy intensity during 1989-2021",
      xaxis = list(zeroline = FALSE, title = "Year", tickangle = -45),
      yaxis = list(zeroline = FALSE, title = "Energy intensity (GJ/$Million"),
      updatemenus=updatemenus)
    plot4
  })
}

shinyApp(ui, server)
```