

Individual Assignment 8: Maps and Interactive Graph

Atharv Prashant Tungatkar

2025-11-04

Importing Libraries that are generally required

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.4.3
```

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

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

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

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.4.3
```

```
library(plotly)
```

```
## Warning: package 'plotly' was built under R version 4.4.3
```

```
##  
## Attaching package: 'plotly'
```

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

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

```
## The following object is masked from 'package:graphics':  
##  
##      layout
```

```
library(xts)
```

```
## Warning: package 'xts' was built under R version 4.4.3
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.4.3
```

```
##  
## Attaching package: 'zoo'
```

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

```
##  
##      as.Date, as.Date.numeric
```

```
##  
## ##### Warning from 'xts' package #####  
## #  
## # The dplyr lag() function breaks how base R's lag() function is supposed to #  
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or #  
## # source() into this session won't work correctly. #  
## #  
## # Use stats:::lag() to make sure you're not using dplyr:::lag(), or you can add #  
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop #  
## # dplyr from breaking base R's lag() function. #  
## #  
## # Code in packages is not affected. It's protected by R's namespace mechanism #  
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning. #  
## #  
## #####
```

```
##  
## Attaching package: 'xts'
```

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

```
##  
##      first, last
```

```
library(dygraphs)
```

```
## Warning: package 'dygraphs' was built under R version 4.4.3
```

```
library(maps)
```

```

## Warning: package 'maps' was built under R version 4.4.3

library(gapminder)

## Warning: package 'gapminder' was built under R version 4.4.3

library(usmap)

## Warning: package 'usmap' was built under R version 4.4.3

library(tibble)

## Warning: package 'tibble' was built under R version 4.4.2

library(maps)
library(mapproj)

## Warning: package 'mapproj' was built under R version 4.4.3

```

Question 1

Loading the dataset using the given code

```

data("statepop")

## Make the state names Lower-cased
statepop$region <- tolower(statepop$full)
library(ggplot2)
states_map <- map_data("state")
## Merge
population_map <- merge(states_map, statepop, by= "region", all.x=T)

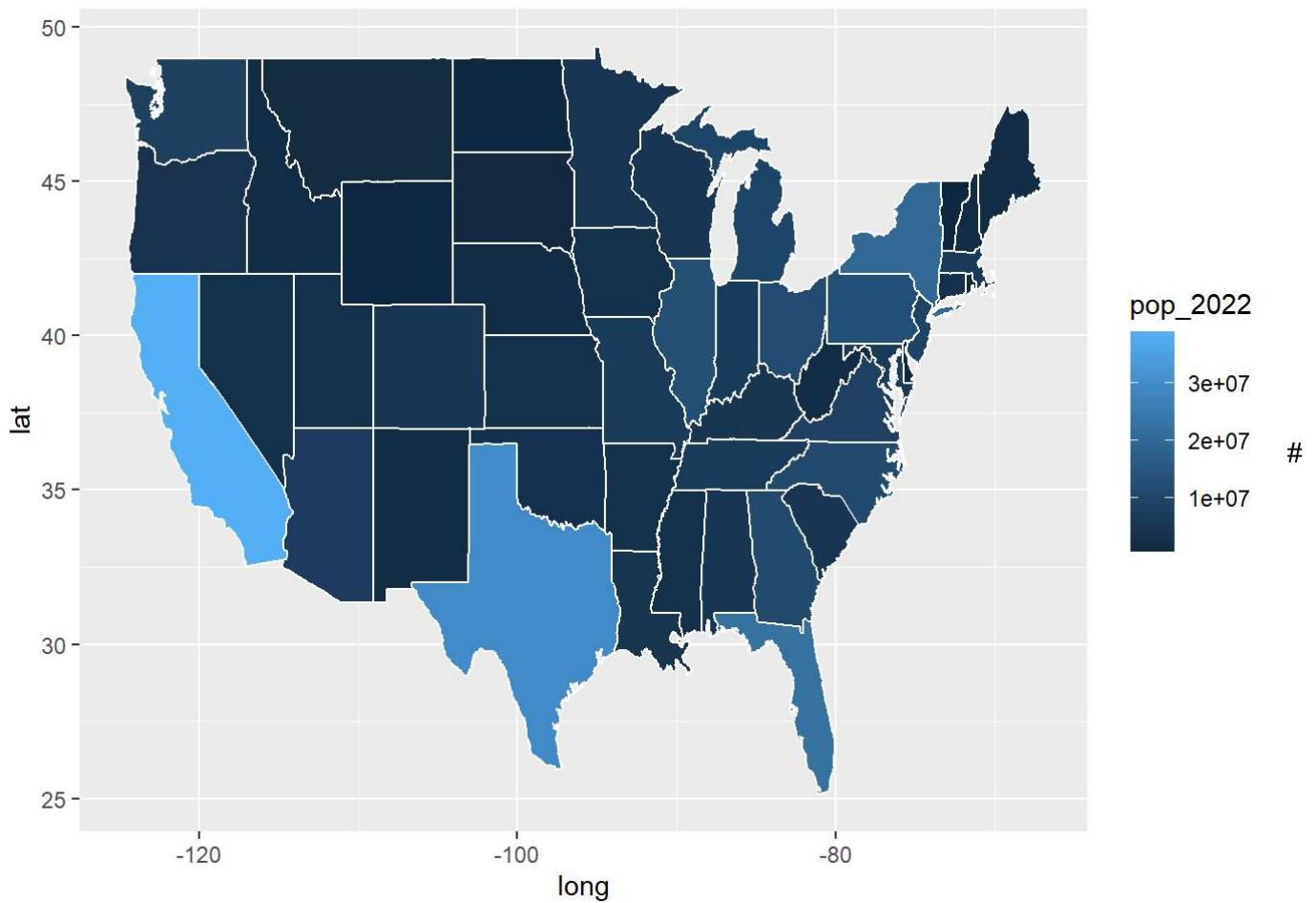
```

Plotting Map with the population of 2022 for states.

```

population_states<-ggplot(population_map, aes(x = long, y = lat, group = group, fill = pop_20
22)) +
  geom_polygon(colour = "white")
population_states

```



Question 2

Loading Dataset

```
marketing_data<-read.csv("E:\\MBA-MAR--657-Visual-Analytics\\Datasets\\marketing_campaign.csv")
head(marketing_data,2)
```

```
##      ID Year_Birth Education Marital_Status Income Kidhome Teenhome Dt_Customer
## 1 5524      1957 Graduation       Single  58138      0      0 9/4/2012
## 2 2174      1954 Graduation       Single  46344      1      1 3/8/2014
##   Recency MntWines MntFruits MntMeatProducts MntFishProducts MntSweetProducts
## 1      58     635      88            546          172           88
## 2      38      11       1             6            2            1
##   MntGoldProds NumDealsPurchases NumWebPurchases NumCatalogPurchases
## 1        88                  3                  8                 10
## 2         6                  2                  1                 1
##   NumStorePurchases NumWebVisitsMonth AcceptedCmp3 AcceptedCmp4 AcceptedCmp5
## 1            4                  7                  0                  0                 0
## 2            2                  5                  0                  0                 0
##   AcceptedCmp1 AcceptedCmp2 Complain Response
## 1          0          0        0        1
## 2          0          0        0        0
```

Interactive Graph 1

Lets first examine the relationship between spending on fish products vs spending on sweet across different education levels with number of store purchases as size.

```

sweet_vs_fish_plot <- plot_ly(
  marketing_data,
  x = ~MntSweetProducts,
  y = ~MntFishProducts,
  color = ~Education,
  size = ~NumStorePurchases,
  type = 'scatter',
  mode = 'markers',
  hoverinfo = 'text',
  text = ~paste(
    "Sweet Spend: $", MntSweetProducts,
    "<br>Fish Spend: $", MntFishProducts,
    "<br>Store Purchases: ", NumStorePurchases,
    "<br>Education:", Education
  )
) %>%
  layout(
    title = "Sweet vs Fish Product Spending by Education and Marital Status",
    xaxis = list(title = "Spending on Sweet Products (USD)"),
    yaxis = list(title = "Spending on Fish Products (USD)"),
    legend = list(title = list(text='Education'))
  )
)

sweet_vs_fish_plot

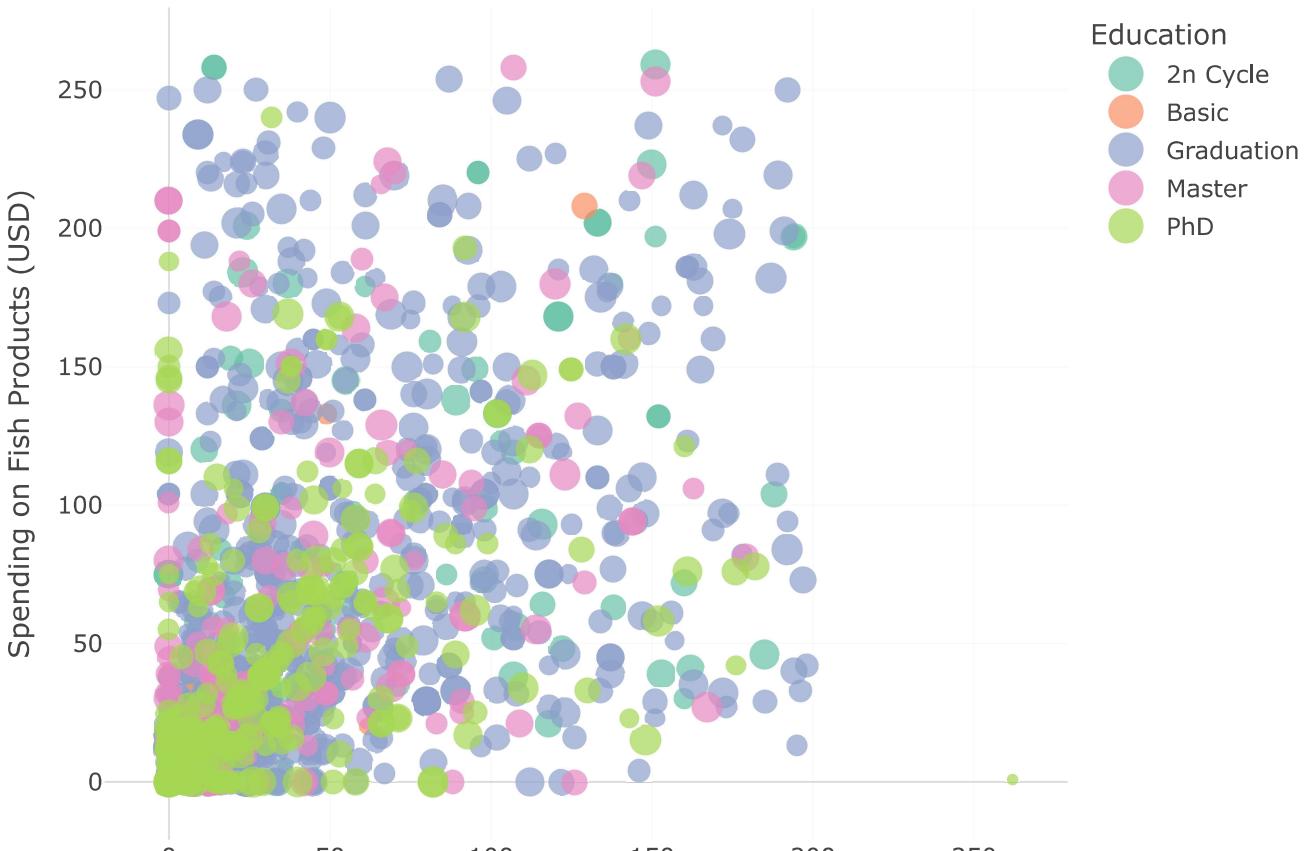
```

```

## Warning: `line.width` does not currently support multiple values.
## Warning: `line.width` does not currently support multiple values.
## Warning: `line.width` does not currently support multiple values.
## Warning: `line.width` does not currently support multiple values.
## Warning: `line.width` does not currently support multiple values.

```

Sweet vs Fish Product Spending by Education and Marital Status



U

D

100

150

200

250

Spending on Sweet Products (USD)

It can be seen that people having a phd are more likely to spend less on sweets and fish.

Interactive Graph 2

Lets examine the relationship between average spending across average income for different education levels at different ages

```

marketing_campaign <- marketing_data %>%
  mutate(Age = 2025 - Year_Birth,
        AgeGroup = cut(Age, breaks = c(18,30,40,50,60,70,100),
                      labels = c("18-30","31-40","41-50","51-60","61-70","70+")),
        Total_Spend = MntWines + MntFruits + MntMeatProducts +
                      MntFishProducts + MntSweetProducts + MntGoldProds)

edu_spend <- marketing_campaign %>%
  group_by(Education, AgeGroup) %>%
  summarise(Avg_Spend = mean(Total_Spend, na.rm = TRUE),
            Avg_Income = mean(Income, na.rm = TRUE),
            .groups = 'drop')

anim_spend <- ggplot(edu_spend, aes(x = Avg_Income, y = Avg_Spend,
                                       color = Education,
                                       frame = AgeGroup)) +
  geom_point(alpha = 2) +
  labs(title = "Animated Average Spending by Education Level and Age Group",
       x = "Average Income (USD)", y = "Average Spending (USD)")

anim_spend <- ggplotly(anim_spend)

anim_spend

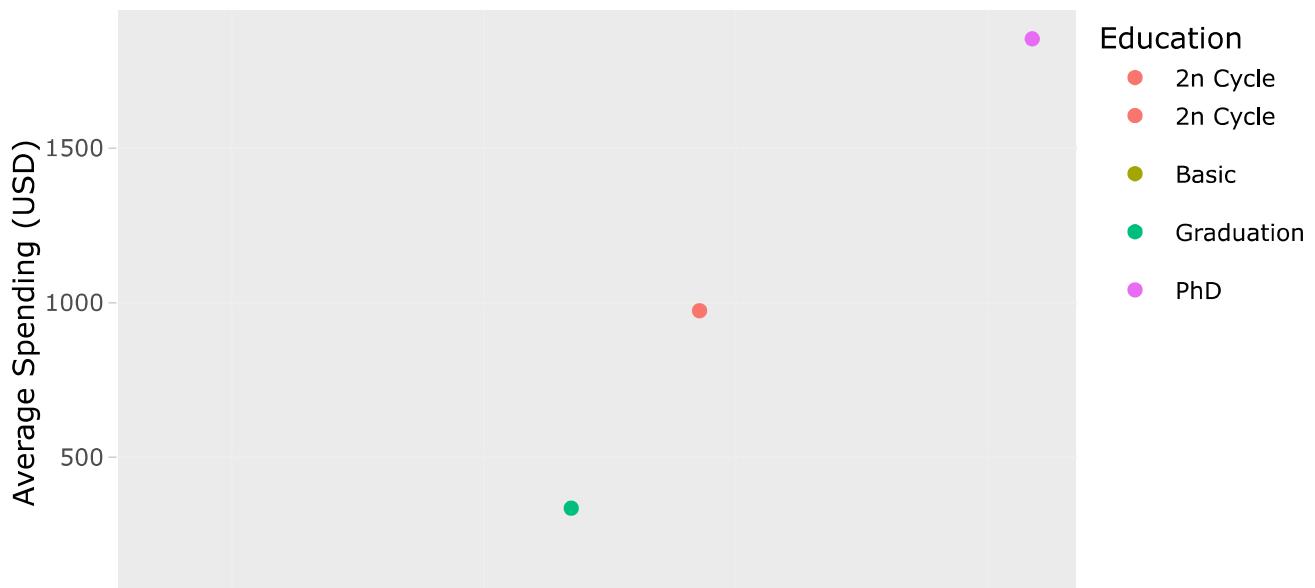
```

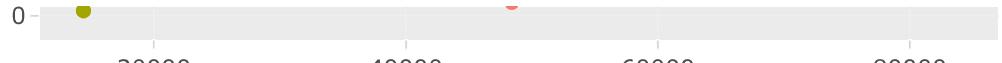
```

## Warning in p$x$data[firstFrame] <- p$x$frames[[1]]$data: number of items to
## replace is not a multiple of replacement length

```

Animated Average Spending by Education Level and Age Group





Average Income (USD) ~AgeGroup: 18-30

Play

