Homework 9

GONG Kuiyuan

Student ID: 39-246182

Student name: GONG Kuiyuan

Preferred name: Eddie

Answers:

Task 1:

```
library(pacman)
p_load(griffen, griffendata, ggplot2, ggmap, tidyverse, ggthemes, sf, scales, geomtextpath, g

japan_prefecture <- read.csv("/Users/edisonkung/Desktop/R for empirical research/HW/japan_pre
print(japan_prefecture)
```

```
   Prefecture Visited
1    Hokkaido    TRUE
2      Aomori   FALSE
3       Iwate   FALSE
4      Miyagi    TRUE
5       Akita   FALSE
6    Yamagata   FALSE
7   Fukushima   FALSE
8     Ibaraki    TRUE
9     Tochigi    TRUE
10      Gunma    TRUE
11    Saitama    TRUE
12      Chiba    TRUE
13      Tokyo    TRUE
14   Kanagawa    TRUE
```

```
15    Niigata    TRUE
16     Toyama   FALSE
17   Ishikawa   FALSE
18      Fukui   FALSE
19  Yamanashi   FALSE
20     Nagano   FALSE
21       Gifu   FALSE
22   Shizuoka    TRUE
23      Aichi    TRUE
24        Mie   FALSE
25      Shiga    TRUE
26      Kyoto    TRUE
27      Osaka    TRUE
28      Hyogo    TRUE
29       Nara   FALSE
30   Wakayama   FALSE
31    Tottori   FALSE
32    Shimane   FALSE
33    Okayama   FALSE
34  Hiroshima    TRUE
35  Yamaguchi    TRUE
36  Tokushima   FALSE
37     Kagawa   FALSE
38      Ehime   FALSE
39      Kochi   FALSE
40    Fukuoka    TRUE
41       Saga   FALSE
42   Nagasaki   FALSE
43   Kumamoto   FALSE
44       Oita   FALSE
45   Miyazaki   FALSE
46  Kagoshima   FALSE
47    Okinawa    TRUE
```

Task 2:

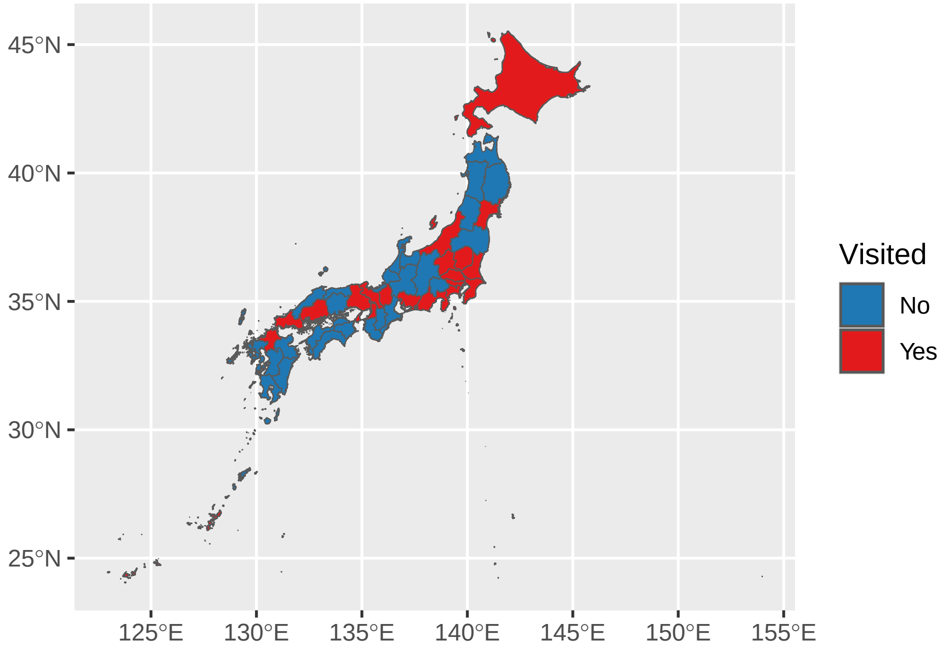
```
japan_prefecture <- japan_prefecture |>
  rename(prefecture = Prefecture) |>
  arrange(prefecture) |>
  mutate(Visited = if_else(Visited, "Yes", "No"))

japan_sf_merge <- japan_sf |>
```

```
  left_join(japan_prefecture, by = "prefecture")  |>
  print()
```

```
Simple feature collection with 47 features and 2 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: 122.9332 ymin: 24.04542 xmax: 153.9869 ymax: 45.52279
Geodetic CRS:  WGS 84
First 10 features:
   prefecture Visited                       geometry
1       Aichi     Yes MULTIPOLYGON (((137.0952 34...
2       Akita      No MULTIPOLYGON (((139.5573 39...
3      Aomori      No MULTIPOLYGON (((141.3986 40...
4       Chiba     Yes MULTIPOLYGON (((139.8249 34...
5       Ehime      No MULTIPOLYGON (((132.5586 32...
6       Fukui      No MULTIPOLYGON (((136.0419 35...
7     Fukuoka     Yes MULTIPOLYGON (((130.0697 33...
8   Fukushima      No MULTIPOLYGON (((140.7961 36...
9        Gifu      No MULTIPOLYGON (((137.2905 36...
10      Gunma     Yes MULTIPOLYGON (((139.1804 36...
```

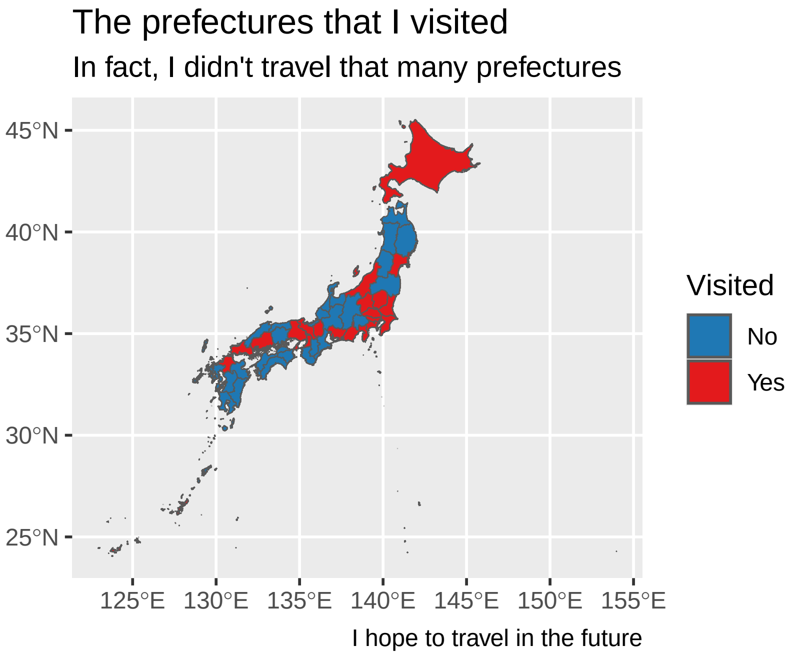
Task 3:

```
visited_prefecture <- japan_sf_merge |>
  ggplot(
    aes(fill = Visited)
  ) +
  geom_sf() +
  scale_fill_manual(values = c("Yes" = "#e31a1c", "No" = "#1f78b4"))

visited_prefecture
```



Task 4:

```
visited_prefecture <- visited_prefecture +
  labs(
    title = "The prefectures that I visited",
    subtitle = "In fact, I didn't travel that many prefectures",
    caption = "I hope to travel in the future"
  )

visited_prefecture
```


The prefectures that I visited
In fact, I didn't travel that many prefectures
45°N
40°N
35°N
30°N
25°N
125°E
130°E
135°E
140°E
145°E
150°E
155°E
I hope to travel in the future
Visited
🟦 No
🟥 Yes

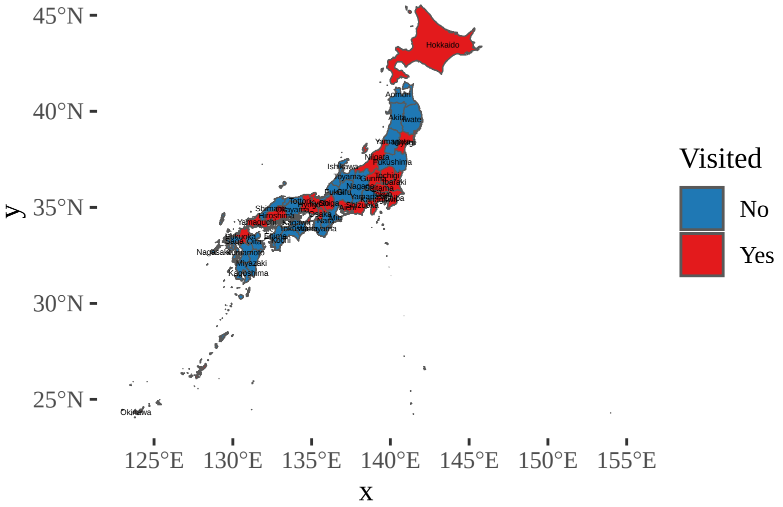
Task 5:

```
visited_prefecture <- visited_prefecture +
  geom_sf_text(
    aes(label = prefecture),
    size = 1
  ) +
  theme_tufte()

visited_prefecture
```

```
Warning in st_point_on_surface.sfc(sf::st_zm(x)): st_point_on_surface may not
give correct results for longitude/latitude data
Warning in st_point_on_surface.sfc(sf::st_zm(x)): st_point_on_surface may not
give correct results for longitude/latitude data
```

The prefectures that I visited

In fact, I didn't travel that many prefectures



I hope to travel in the future

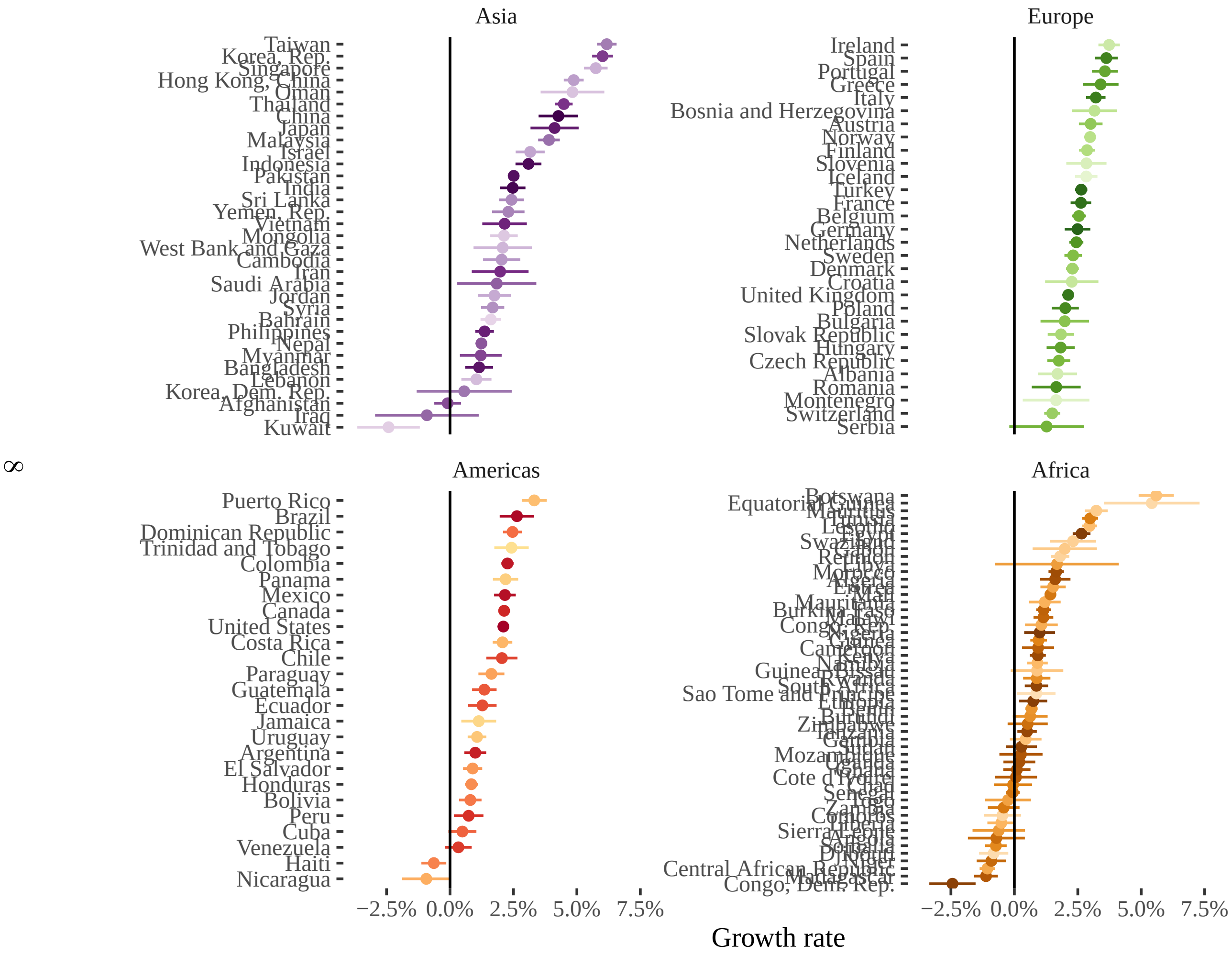
Task 6:

```
growth <- function(df) {
  model <- lm(log(gdpPercap) ~ year, df)
  return(model)
}

gapminder_nested <- gapminder |>
  group_by(country) |>
  nest() |>
  mutate(model = map(data, growth)) |>
  mutate(tidy_model = map(model, tidy, conf.int = TRUE)) |>
  select(-data, -model) |>
  unnest(cols = c("tidy_model")) |>
  filter(term == "year") |>
  left_join(gapminder |> select(country, continent) |> distinct(), by = "country") |>
  filter(continent != "Oceania") |>
  group_by(continent) |>
  arrange(estimate, .by_group = TRUE) |>
  mutate(country = factor(country, levels = country)) |>
  mutate(continent = factor(continent, levels = c("Asia", "Europe", "Americas", "Africa")))
```

```
gapminder_plot <- gapminder_nested |>
  ggplot(
    aes(x = estimate, y = country, color = country)
  ) +
  geom_point(show.legend = FALSE) +
  facet_wrap(~continent, scales = "free_y") +
  scale_x_continuous(labels = percent_format()) +
  theme_tufte() +
  scale_color_manual(values = country_colors) +
  geom_segment(aes(x = conf.low, xend = conf.high, y = country, yend = country), show.legend = FALSE) +
  geom_vline(xintercept = 0) +
  labs(
    x = "Growth rate",
    y = NULL,
    title = "Annual growth in per capita gdp",
    subtitle = "1952 - 2007",
    caption = "Source: Gapminder data, bars represent 95% confidence intervals, zero growth highlighted with black
  )

print(gapminder_plot)
```

Annual growth in per capita gdp

1952 − 2007



Source: Gapminder data, bars represent 95% confidence intervals, zero growth highlighted with black line.