DATASCI 306, Fall 2024, Final Group Project

Shalini Asokkumar, Sophia Giuliani, Jonathan Sarasa, Alicia Zhou

Throughout this course, you've dedicated yourself to refining your analytical abilities using R programming language. These skills are highly coveted in today's job market!

Now, for the semester project, you'll apply your learning to craft a compelling `Data Story` that can enrich your portfolio and impress prospective employers. Collaborating with a team (up to 5 members of your choosing), you'll construct a Data Story akin to the example provided here: https://ourworldindata.org/un-population-2024-revision

Data is already in the `data` folder. This data is downloaded from: https://population.un.org/wpp/Download/Standard/MostUsed/

You'll conduct Exploratory Data Analysis (EDA) on the provided data. The provided article already includes 6 diagrams. Show either the line or the map option for these 6 charts. You may ignore the table view. I'm also interested in seeing how each team will expand upon the initial analysis and generate additional 12 insightful charts that includes US and any other region or country that the author did not show. For e.g., one question you may want to answer is; US population is expected to increase to 421 million by 2100. You may want to show how the fertility rate and migration may be contributing to this increase in population.

Deliverable

1. **Requirement-1 (2 pt)** Import the data given in the .xlxs file into two separate dataframes;
   - one dataframe to show data from the `Estimates` tab
   - one dataframe to show data from the `Medium variant` tab

Hint: Some of the steps you may take while importing include:

- skip the first several comment lines in the spread sheet
- Importing the data as text first and then converting the relevant columns to different datatypes in step 2 below.

```
estimates = read_excel("data.xlsx", skip = 15, sheet = "Estimates")
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
## * `` -> `...13`
## * `` -> `...14`
## * `` -> `...15`
## * `` -> `...16`
```

```
## * `` -> `...17`
## * `` -> `...18`
## * `` -> `...19`
## * `` -> `...20`
## * `` -> `...21`
## * `` -> `...22`
## * `` -> `...23`
## * `` -> `...25`
## * `` -> `...26`
## * `` -> `...27`
## * `` -> `...28`
## * `` -> `...29`
## * `` -> `...30`
## * `` -> `...32`
## * `` -> `...33`
## * `` -> `...34`
## * `` -> `...35`
## * `` -> `...36`
## * `` -> `...37`
## * `` -> `...38`
## * `` -> `...39`
## * `` -> `...40`
## * `` -> `...41`
## * `` -> `...42`
## * `` -> `...43`
## * `` -> `...44`
## * `` -> `...45`
## * `` -> `...46`
## * `` -> `...47`
## * `` -> `...48`
## * `` -> `...49`
## * `` -> `...50`
## * `` -> `...51`
## * `` -> `...52`
## * `` -> `...53`
## * `` -> `...54`
## * `` -> `...55`
## * `` -> `...56`
## * `` -> `...57`
## * `` -> `...58`
## * `` -> `...59`
## * `` -> `...60`
## * `` -> `...61`
## * `` -> `...62`
## * `` -> `...63`
## * `` -> `...65`
```

```
colnames(estimates) <- as.character(unlist(estimates[1, ]))
estimates = estimates[-1,]

mediums = read_excel("data.xlsx", skip = 15, sheet = "Medium variant")
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
```

```
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
## * `` -> `...13`
## * `` -> `...14`
## * `` -> `...15`
## * `` -> `...16`
## * `` -> `...17`
## * `` -> `...18`
## * `` -> `...19`
## * `` -> `...20`
## * `` -> `...21`
## * `` -> `...22`
## * `` -> `...23`
## * `` -> `...25`
## * `` -> `...26`
## * `` -> `...27`
## * `` -> `...28`
## * `` -> `...29`
## * `` -> `...30`
## * `` -> `...32`
## * `` -> `...33`
## * `` -> `...34`
## * `` -> `...35`
## * `` -> `...36`
## * `` -> `...37`
## * `` -> `...38`
## * `` -> `...39`
## * `` -> `...40`
## * `` -> `...41`
## * `` -> `...42`
## * `` -> `...43`
## * `` -> `...44`
## * `` -> `...45`
## * `` -> `...46`
## * `` -> `...47`
## * `` -> `...48`
## * `` -> `...49`
## * `` -> `...50`
## * `` -> `...51`
## * `` -> `...52`
## * `` -> `...53`
## * `` -> `...54`
## * `` -> `...55`
## * `` -> `...56`
## * `` -> `...57`
## * `` -> `...58`
## * `` -> `...59`
```

```
## * `` -> `...60`
## * `` -> `...61`
## * `` -> `...62`
## * `` -> `...63`
## * `` -> `...65`
```

```
colnames(mediums) <- as.character(unlist(mediums[1, ]))
mediums = mediums[-1,]
```

2. Requirement-2 (5 pt)

You should show at least 5 steps you adopt to clean and/or transform the data. Your cleaning should include:

- Renaming column names to make it more readable; removing space, making it lowercase or completely giving a different short name; all are acceptable.
- Removing rows that are irrelevant; look at rows that have Type value as 'Label/Separator'; are those rows required?
- Removing columns that are redundant; For e.g., variant column
- Converting text values to numeric on the columns that need this transformation

You could also remove the countries/regions that you are not interested in exploring in this step and re-save a smaller file in the same `data` folder, with a different name so that working with it becomes easier going forward.

Explain your reasoning for each clean up step.

```
est_values <- estimates |> filter(Type != "Label/Separator") |>
  select(Index, Year:last_col()) |>
  mutate(across(where(is.character), as.double, .names = '{col}'))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `across(where(is.character), as.double, .names = "{col}")`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
  # converts all numeric columns from characters into doubles
  # removes Label/Separator type
estimates$Index <- estimates$Index |> as.double()
  # creates key column of Index and converts to double
estimates <- estimates |> select(Index, `Region, subregion, country or area *`,
                                 Type)
  # selecting relevant columns from original database
estimates <- estimates |> full_join(est_values, join_by(Index)) |> filter(Type != "Label/Separator")
  # only includes relevant data in Estimates

med_values <- mediums |> filter(Type != "Label/Separator") |>
  select(Index, Year:last_col()) |>
  mutate(across(where(is.character), as.double, .names = '{col}'))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `across(where(is.character), as.double, .names = "{col}")`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
mediums$Index <- mediums$Index |> as.double()
mediums <- mediums |> select(Index, `Region, subregion, country or area *`,
                                 Type)
mediums <- mediums |> full_join(med_values, join_by(Index)) |> filter(Type != "Label/Separator")
  # replicated the above for the mediums dataset
```

```r
#renaming columns
corrected_colnames <- c(
  "index",
  "region_subregion_country_area",
  "type",
  "year",
  "total_pop_january_thousands",
  "total_pop_july_thousands",
  "male_pop_july_thousands",
  "female_pop_july_thousands",
  "pop_density_july_person_per_sq_km",
  "pop_sex_ratio_july_males_per_100_females",
  "med_age_july_years",
  "natural_change_births_minus_deaths_thousands",
  "rate_of_natural_change_per_1000",
  "population_change_thousands",
  "population_growth_rate_percentage",
  "population_annual_doubling_time_years",
  "births_thousands",
  "births_by_woman_aged_15_to_19_thousands",
  "crude_birth_rate_per_1000_pop",
  "total_fertility_rate_live_births_per_woman",
  "net_reproduction_rate_surviving_daughters_per_woman",
  "mean_age_childbearing_years",
  "sex_ratio_at_birth_males_per_100_female_births",
  "total_deaths_thousands",
  "male_deaths_thousands",
  "female_deaths_thousands",
  "crude_death_rate_deaths_per_1000_population",
  "total_life_expectancy_at_birth_years",
  "male_life_expectancy_at_birth_years",
  "female_life_expectancy_at_birth_years",
  "total_life_expectancy_at_age_15_years",
  "male_life_expectancy_at_age_15_years",
  "female_life_expectancy_at_age_15_years",
  "total_life_expectancy_at_age_65_years",
  "male_life_expectancy_at_age_65_years",
  "female_life_expectancy_at_age_65_years",
  "total_life_expectancy_at_age_80_years",
  "male_life_expectancy_at_age_80_years",
  "female_life_expectancy_at_age_80_years",
  "infant_deaths_under_age_1_thousands",
  "infant_morality_rate_infant_deaths_per_1000_births",
  "live_births_surviving_to_age_1_thousands",
  "under_five_deaths_thousands",
  "deaths_under_age_5_per_1,000_live_births",
  "total_male_mortality_before_age_40_per_1000_births",
  "male_mortality_before_age_40_per_1000_births",
  "female_mortality_before_age_40_per_1000_births",
  "total_mortality_before_age_60_per_1000_births",
  "male_mortality_before_age_60_per_1000_births",
  "female_mortality_before_age_60_per_1000_births",
  "deaths_under_age_50_per_1000_total_alive_at_15",
```

```
  "deaths_under_age_50_per_1000_males_alive_at_15",
  "deaths_under_age_50_per_1000_females_alive_at_15",
  "deaths_under_age_60_per_1000_total_alive_at_15",
  "deaths_under_age_60_per_1000_males_alive_at_15",
  "deaths_under_age_60_per_1000_females_alive_at_15",
  "net_num_migrants_thousands",
  "net_migration_rate_per_1000"
)

colnames(estimates) <- corrected_colnames
colnames(mediums) <- corrected_colnames
```

3. Requirement-3 (3 pt) Replicate the 6 diagrams shown in the article. Show only the '2024' projection values where ever you have both '2022' and '2024' displayed. Show only the diagrams that are shown on the webpage with default options.

- population projections from 2024
- projections broken down by world and continent
- fertility rate in children/woman from 1950 - 2100
- population 1950 to 2100
- life expectancy from 1950 to 2023
- annual net migration 1950 to 2023

```
# Population Projections from 2024 (Sophia)

mediums %>% select(year) %>% filter(!is.na(year)) %>% range() # 2024 - 2100
```

```
## [1] 2024 2100
```

```
estimates %>% select(year) %>% filter(!is.na(year)) %>% range() # 1950 - 2023
```

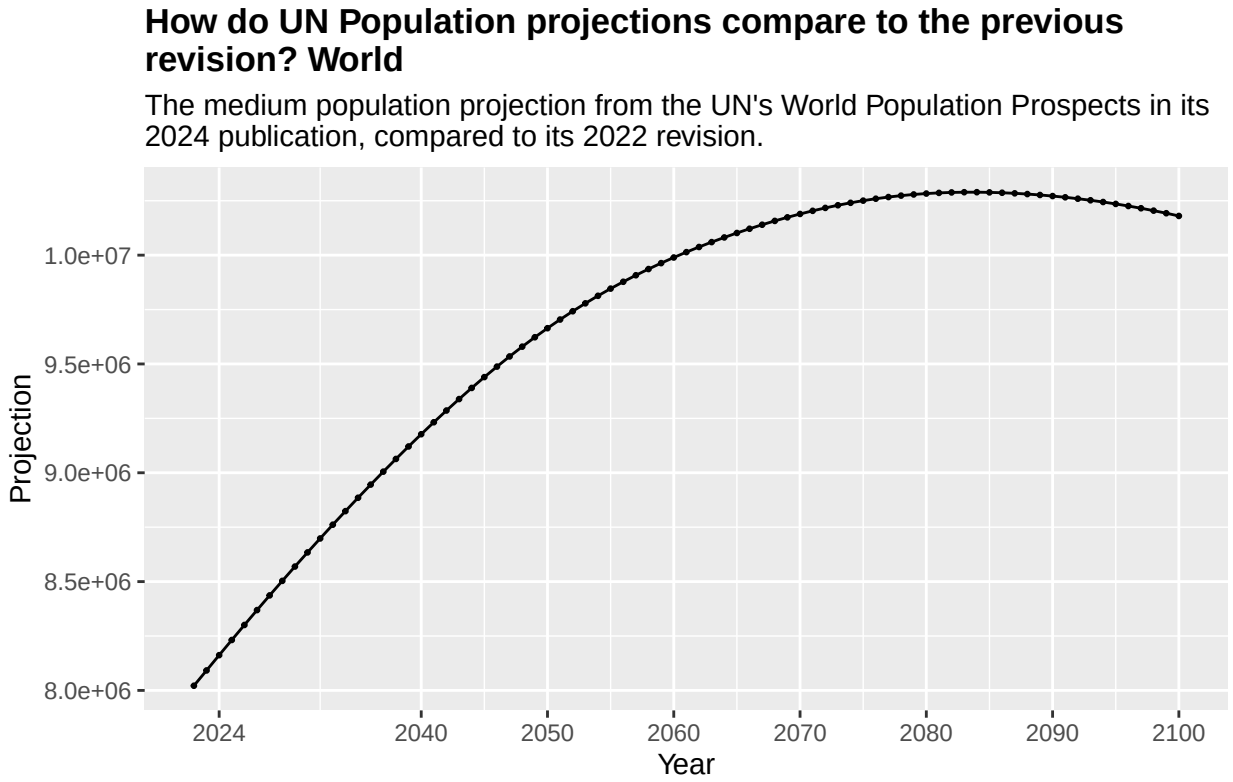
```
## [1] 1950 2023
```

```
# Need to bind the rows from the two dataframes
combined <- rbind(estimates, mediums)
combined %>% select(year) %>% filter(!is.na(year)) %>% range() # 1950 - 2100
```

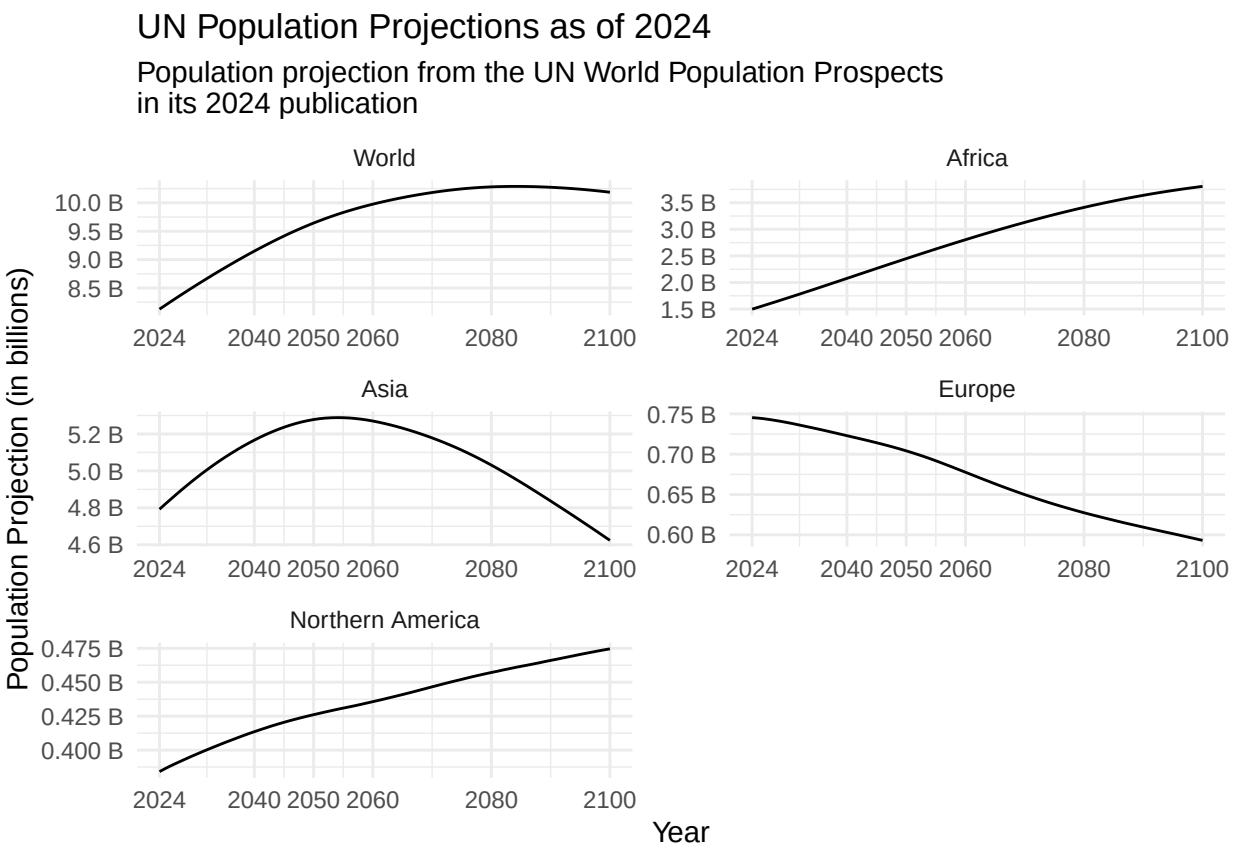
```
## [1] 1950 2100
```

```
combined %>%
  select(year, region_subregion_country_area, total_pop_july_thousands) %>%
  filter(!is.na(year)) %>%
  filter(year >= "2022") %>%
  filter(region_subregion_country_area == "World") %>%
  ggplot(mapping = aes(x = year, y = total_pop_july_thousands)) +
  geom_line() +
  geom_point(size = 0.5) +
  scale_x_continuous(breaks = c(2024, 2040, 2050, 2060, 2070, 2080, 2090, 2100)) +

  # Use an escape sequence to get a new line (to match the graph from article)
  labs(title = "How do UN Population projections compare to the previous\nrevision? World",
       subtitle = str_wrap("The medium population projection from the UN's World Population Prospects i
       x = "Year",
       y = "Projection",
       caption = "Data Source: UN, World Population Prospects (2024)\nOurWorldinData.org/population-grow

  # Modify the caption position using hjust (see citation below)
```

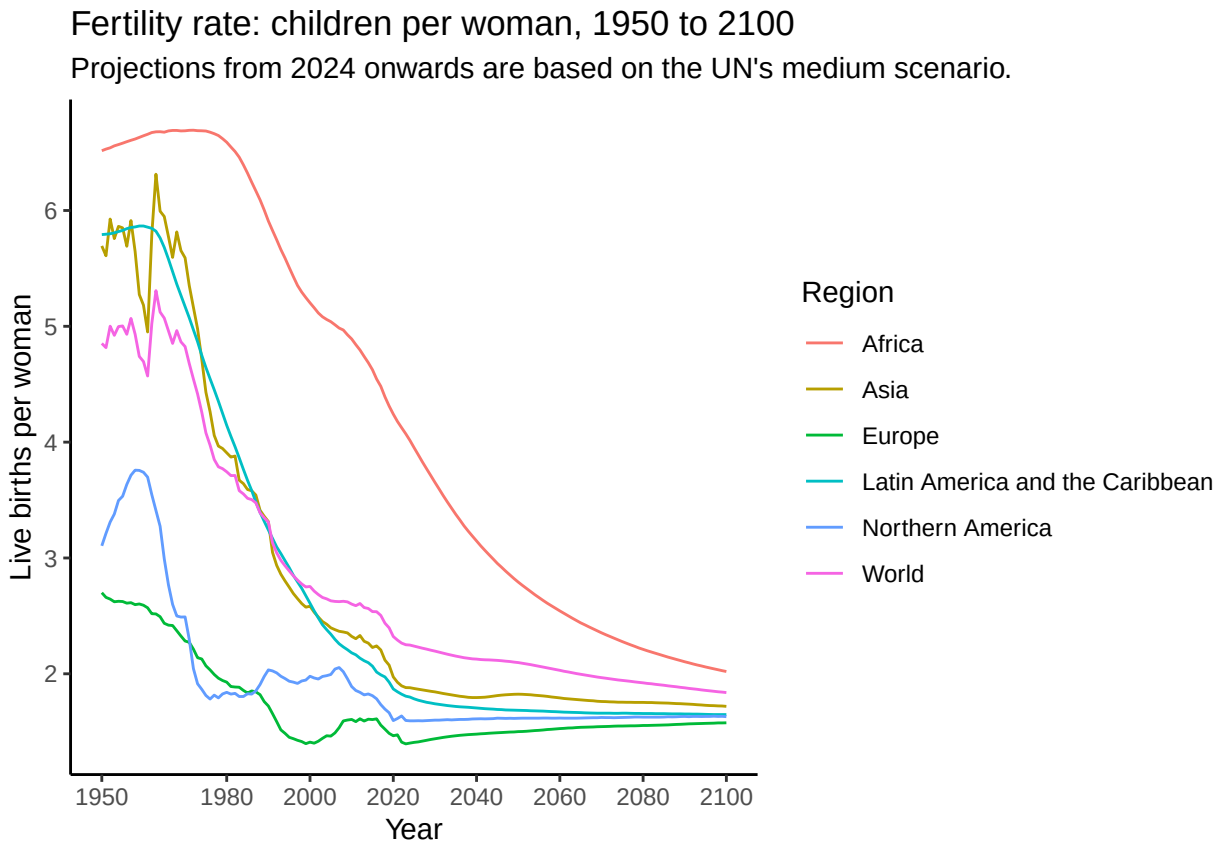
```
  # Citation: https://www.datanovia.com/en/blog/ggplot-title-subtitle-and-caption/#change-caption-posit
  theme(plot.caption = element_text(hjust = 0),
        plot.title = element_text(face = "bold"))
```




```
# projections broken down by world and continent (Shalini)
population_projections <- mediums |>
  select(year, region_subregion_country_area, total_pop_january_thousands) |>
  filter(region_subregion_country_area %in%
           c("World", "Africa", "Asia", "Europe", "Northern America", "Latin America and the Carribean"

population_projections$facet = factor(population_projections$region_subregion_country_area,
                                      levels = c("World", "Africa", "Asia", "Europe", "Northern America
                                                 "Latin America and the Carribean"))

population_projections |>
  ggplot(aes(x = year,
             y = total_pop_january_thousands)) +
  geom_line() +
  facet_wrap(~facet, scales = "free",
             ncol = 2) +
  theme_minimal() +
  scale_y_continuous(labels = unit_format(unit = "B", scale = 1e-6)) +
  scale_x_continuous(breaks = c(2024, 2040, 2050, 2060, 2080, 2100)) +
  labs(title = "UN Population Projections as of 2024",
       subtitle = "Population projection from the UN World Population Prospects \nin its 2024 publicati
       x = "Year",
       y = "Population Projection (in billions)")
```

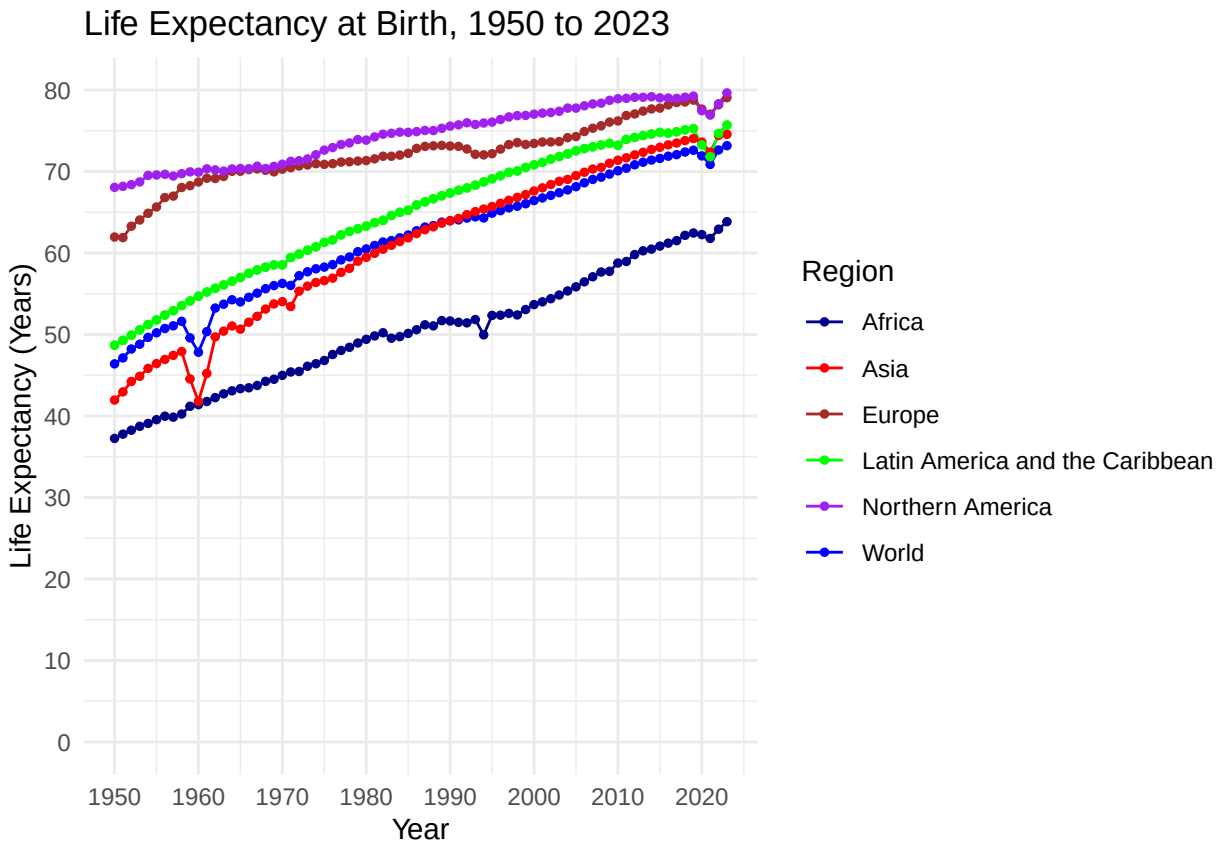



```
# fertility rate in children/woman from 1950 - 2100 (Jonathan)
estimates %>%
  rbind(mediums) %>%
  filter(region_subregion_country_area %in% c('World', 'Africa', 'Asia',
                                              'Northern America','Latin America and the Caribbean',
                                              'Europe')) %>%
  ggplot(aes(x = year, y = total_fertility_rate_live_births_per_woman,
             color = region_subregion_country_area)) +
  geom_line() +
  scale_x_continuous(breaks = c(1950, 1980, 2000, 2020, 2040, 2060, 2080, 2100)) +
  labs(
    title = "Fertility rate: children per woman, 1950 to 2100",
    subtitle = "Projections from 2024 onwards are based on the UN's medium scenario.",
    x = "Year",
    y = "Live births per woman",
    color = "Region"
  ) +
  theme_classic()
```

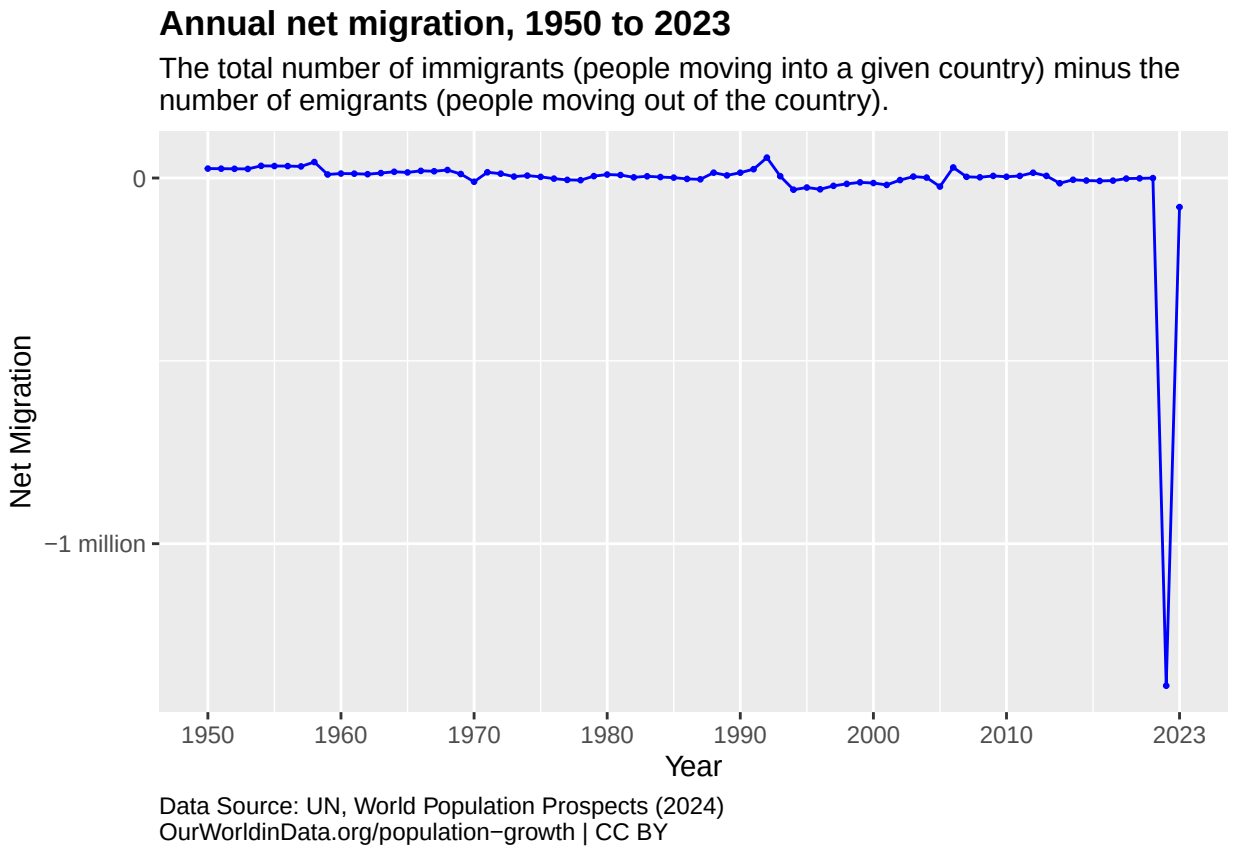



```
# population 1950 to 2100
```

```
# life expectancy from 1950 to 2023 (Anusha)
life_expectancy_data <- estimates %>%
  select(year, region_subregion_country_area, total_life_expectancy_at_birth_years) %>%
  filter(region_subregion_country_area %in% c("World", "Northern America", "Europe", "Asia", "Africa",

ggplot(life_expectancy_data, aes(x = year, y = total_life_expectancy_at_birth_years, color = region_subi
  geom_line() +
  geom_point(size = 1) +
  labs(title = "Life Expectancy at Birth, 1950 to 2023",
       x = "Year",
       y = "Life Expectancy (Years)",
       color = "Region") +
  scale_y_continuous(limits = c(0, 80), breaks = seq(0, 80, by = 10)) +
  scale_x_continuous(limits = c(1950, 2023), breaks = seq(1950, 2023, by = 10)) +
  theme_minimal() +
  scale_color_manual(values = c("World" = "blue",
                                "Northern America" = "purple",
                                "Europe" = "brown",
                                "Asia" = "red",
                                "Africa" = "darkblue",
                                "Latin America and the Caribbean" = "green"))
```




```
# Annual Net Migration 1950 to 2023 (Sophia)

estimates %>%
  select(net_migration_rate_per_1000, year, region_subregion_country_area) %>%
  filter(!is.na(year)) %>%
  filter(region_subregion_country_area == "Ukraine") %>%
  filter(year <= "2023" & year >= "1950") %>%
  ggplot(mapping = aes(x = year, y = net_migration_rate_per_1000)) +
  geom_line(color = "blue") +
  geom_point(size = 0.5, color = "blue") + # Need to make the points/dots smaller
  scale_x_continuous(breaks = c(1950, 1960, 1970, 1980, 1990, 2000, 2010, 2023)) +
  scale_y_continuous(breaks = c(-500, -400, -300, -200, -100, 0),
                     labels = c("-5 million", "-4 million", "-3 million", "-2 million", "-1 million", "
  labs(title = "Annual net migration, 1950 to 2023",
       subtitle = str_wrap("The total number of immigrants (people moving into a given country) minus th
       x = "Year",
       y = "Net Migration",

       # Use an escape sequence in the caption to get a new line
       caption = "Data Source: UN, World Population Prospects (2024)\nOurWorldinData.org/population-grou
  theme(plot.caption = element_text(hjust = 0),
        plot.title = element_text(face = "bold"))
```

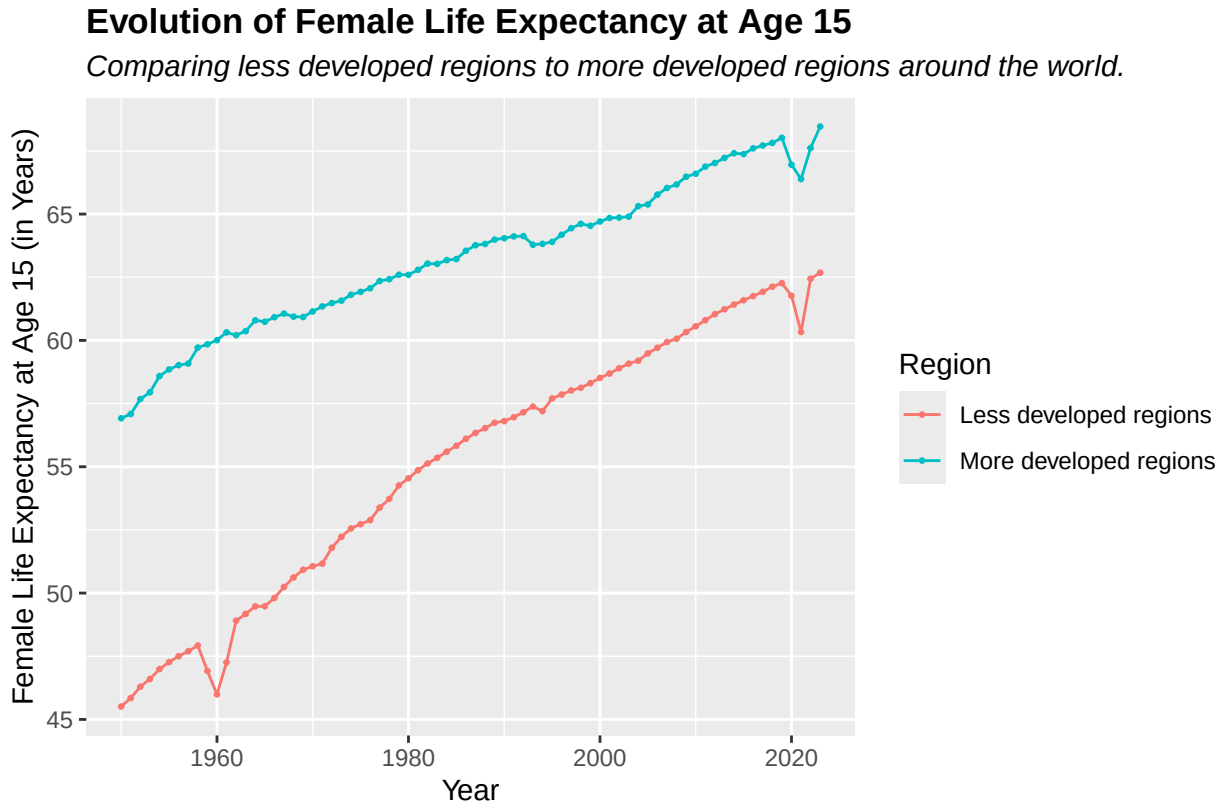


4. Requirement-4 (12 pt)

Select United States related data, and any other country or region(s) of your choosing to perform EDA. Chart at least 12 additional diagrams that may show relationships like correlations, frequencies, trend charts, between various variables with plots of at least 3 different types (line, heatmap, pie, etc.). Every plot should have a title and the x/y axis should have legible labels without any label overlaps for full credit.

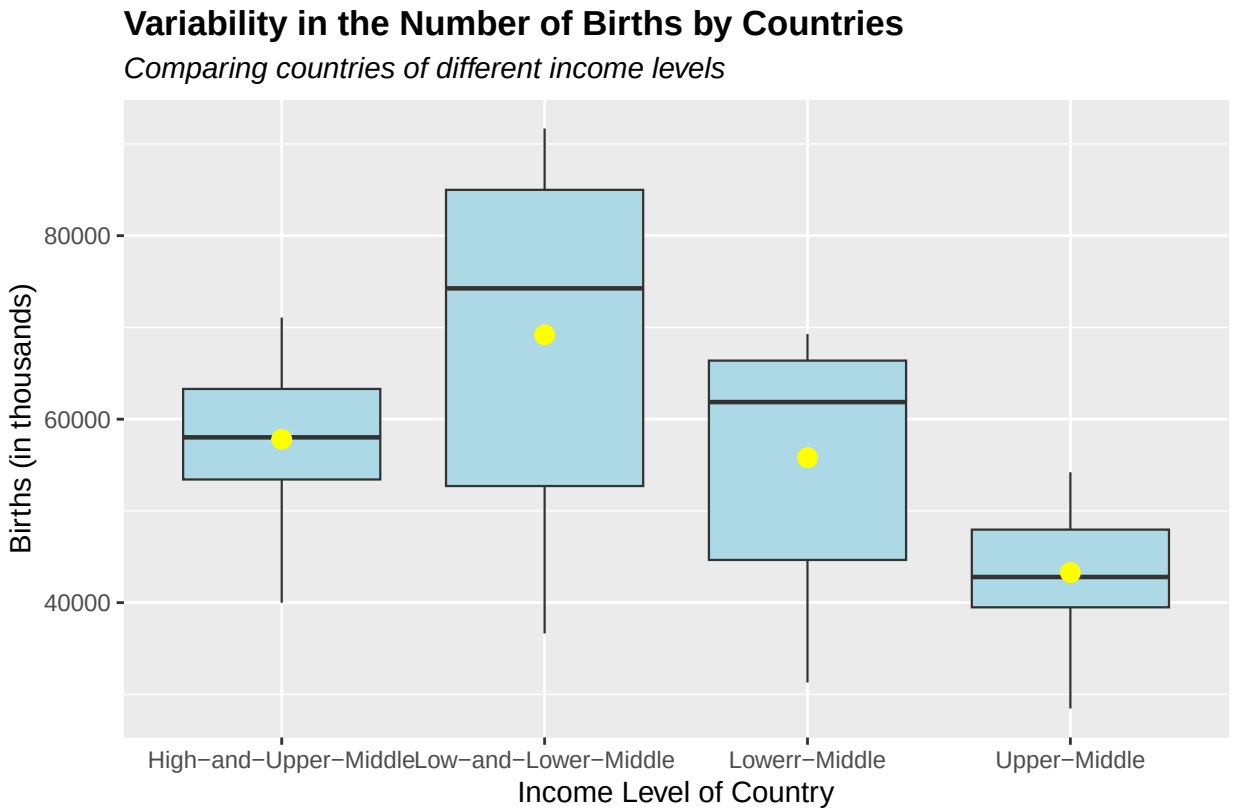
Summarize your interpretations after each chart.

```
# 1 Sophia
# Question: How has the female life expectancy at age 15 evovled over the last 50 years in less develop

estimates %>%
  select(region_subregion_country_area, year, female_life_expectancy_at_age_15_years) %>%
  filter(region_subregion_country_area %in% c("More developed regions", "Less developed regions")) %>%
  ggplot(mapping = aes(x = year, y = female_life_expectancy_at_age_15_years, color = region_subregion_c
  geom_line() +
  geom_point(size = 0.5) +
  scale_x_continuous(limits = c(1950, 2023)) %>%
  scale_y_continuous(breaks = c(45, 50, 55, 60, 65, 70, 75)) %>%
  labs(title = "Evolution of Female Life Expectancy at Age 15",
       subtitle = str_wrap("Comparing less developed regions to more developed regions around the world
       x = "Year",
       y = "Female Life Expectancy at Age 15 (in Years)",
       color = "Region",
       caption = "Data Source: UN, World Population Prospects (2022) - processed by Our World in Data")
  theme(plot.title = element_text(face = "bold"),
        plot.subtitle = element_text(face = "italic"))
```

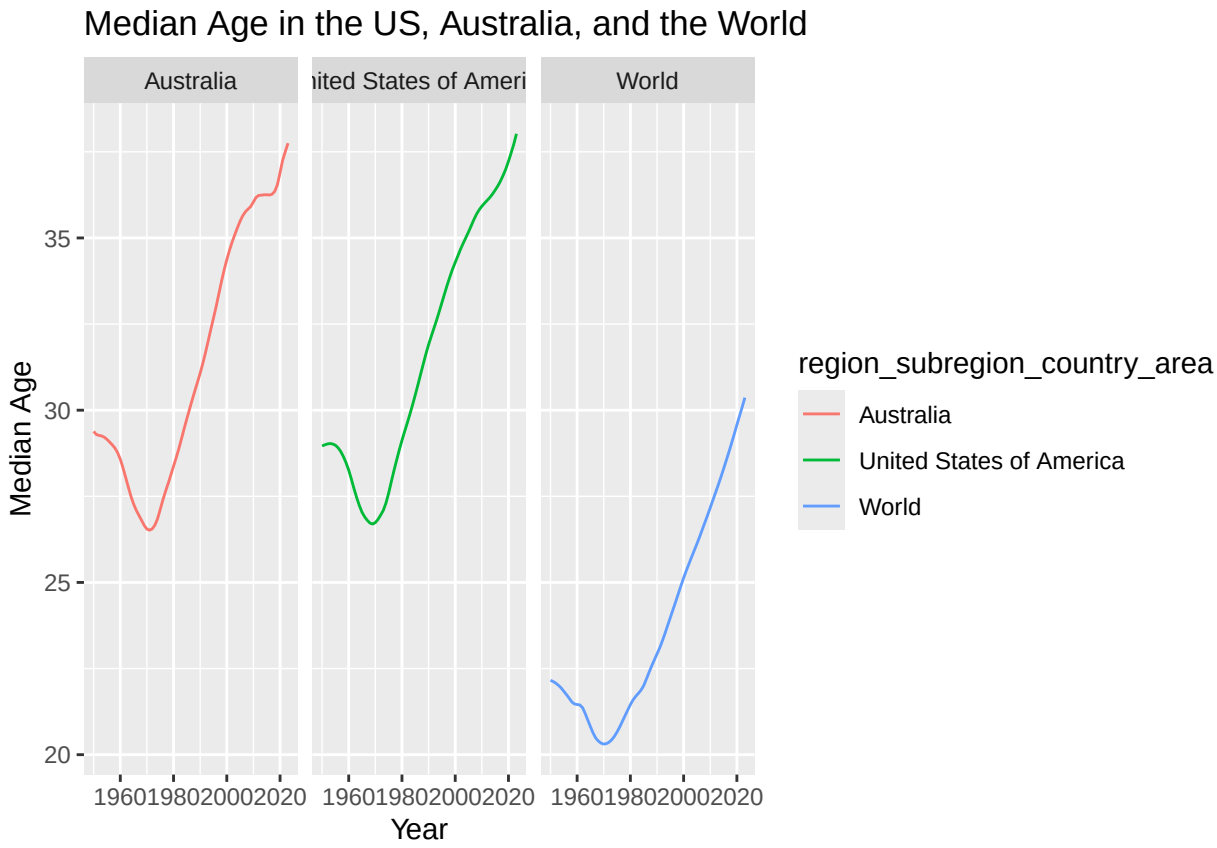


ource: UN, World Population Prospects (2022) − processed by Our World in Data

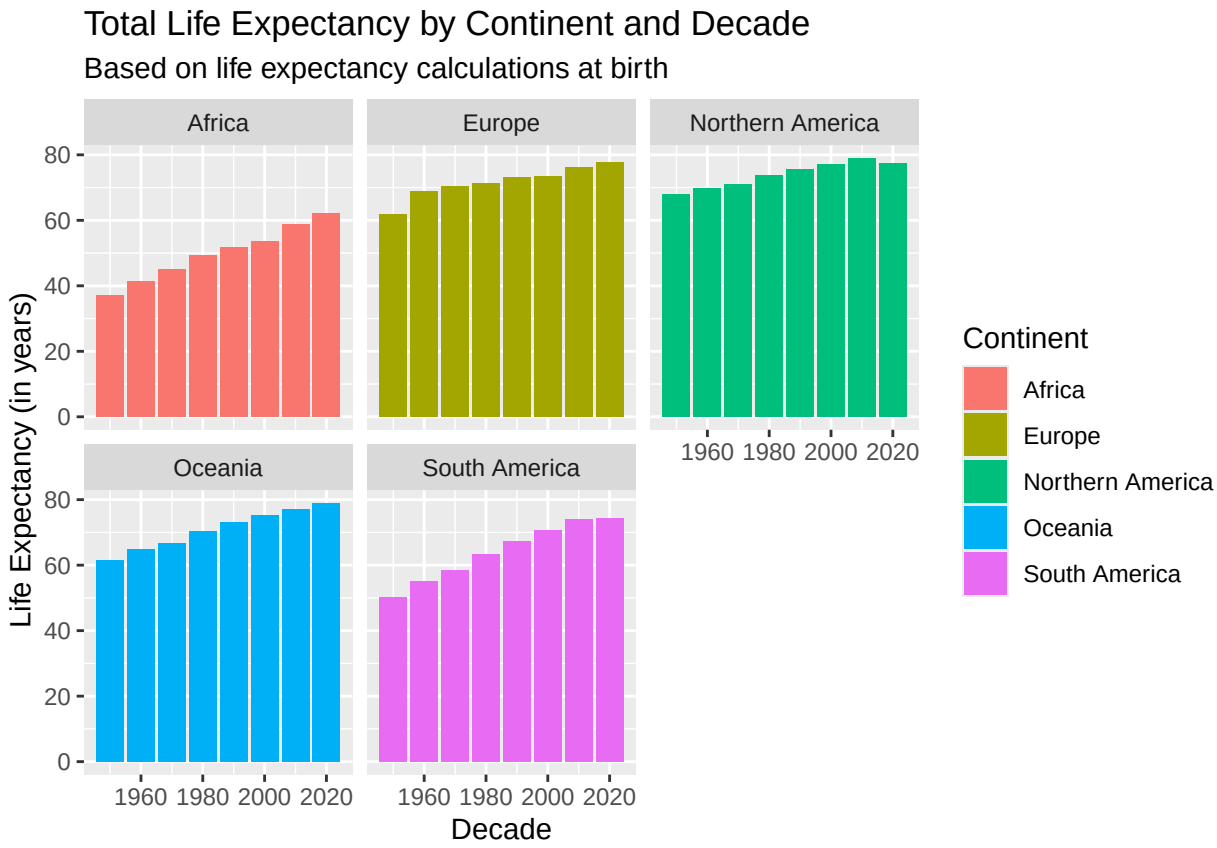
```
# 2 Sophia
# Question: Which income range has the largest variability in the number of births? Are counries of dif

estimates %>%
  select(region_subregion_country_area, births_thousands) %>%
  filter(region_subregion_country_area %in% c("Lower-middle-income countries","Upper-middle-income coun
  ggplot(mapping = aes(x = region_subregion_country_area, y = births_thousands)) +
  geom_boxplot(linewidth = 0.4, fill = "lightblue") +
  stat_summary(fun = mean, size = 3, color = "yellow", geom = "point") +
  scale_x_discrete(labels = c("High-and-Upper-Middle", "Low-and-Lower-Middle", "Lowerr-Middle", "Upper-
  labs(title = "Variability in the Number of Births by Countries",
       subtitle = "Comparing countries of different income levels",
       x = "Income Level of Country",
       y = "Births (in thousands)",
       caption = "Data Source: UN, World Population Prospects (2022) - processed by Our World in Data")
   theme(plot.title = element_text(face = "bold"),
         plot.subtitle = element_text(face = "italic"),
         plot.caption = element_text(hjust = 0))
```



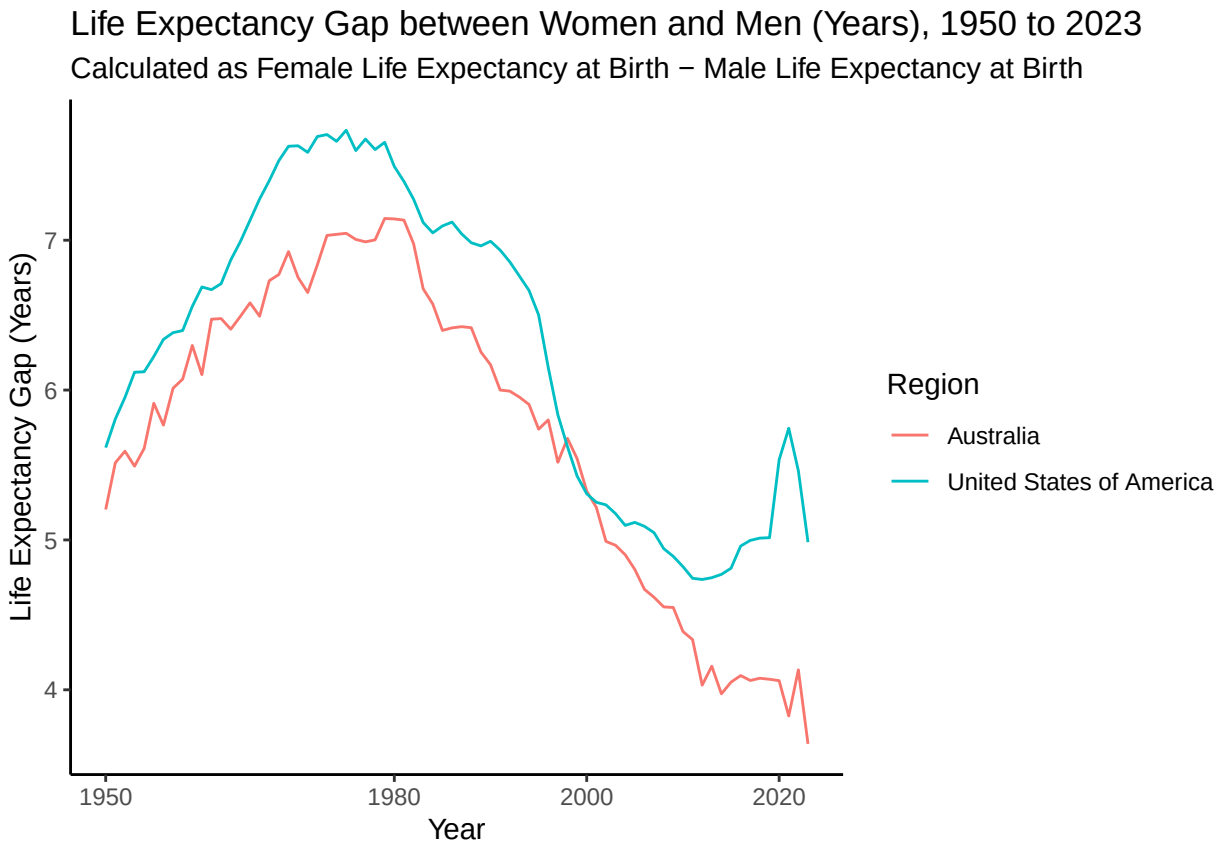

```
# 3 Shalini
# Question: How is the median age changing in the United States, Australia, and the World?

estimates |> select(region_subregion_country_area, med_age_july_years, year) |>
  filter(region_subregion_country_area %in% c("United States of America", "Australia", "World")) |>
  ggplot(aes(x = year, y = med_age_july_years,
             color = region_subregion_country_area)) +
  geom_line() +
  facet_wrap(~region_subregion_country_area) +
  labs(title = "Median Age in the US, Australia, and the World",
       x = "Year",
       y = "Median Age")
```




```
# 4 Shalini
# Question: How has total life expectancy at birth changed by continent at the beginning of each decade

estimates |> filter(region_subregion_country_area %in%
                      c("Northern America", "South America",
                        "Oceania", "Europe", "Africa")) |>
  filter(year == 1950 | year == 1960 |
         year == 1970 | year == 1980 |
         year == 1990 | year == 2000 |
         year == 2010 | year == 2020) |>
  group_by(region_subregion_country_area, year) |>
  mutate(mean_life_exp = mean(total_life_expectancy_at_birth_years)) |>
  select(region_subregion_country_area, year, mean_life_exp) |>
  ggplot(aes(x = year, y = mean_life_exp, fill = region_subregion_country_area)) +
  geom_col() +
  facet_wrap(~region_subregion_country_area) +
  labs(title = "Total Life Expectancy by Continent and Decade",
       subtitle = "Based on life expectancy calculations at birth",
       x = "Decade",
       y = "Life Expectancy (in years)",
       fill = "Continent")
```




```
# 5 Jonathan
# Question: How has the Female/Male Life Expectancy Gap changed between the United States and Australia
estimates %>%
  filter(region_subregion_country_area %in% c('United States of America', 'Australia')) %>%
  ggplot(aes(x = year, y = female_life_expectancy_at_birth_years - male_life_expectancy_at_birth_years,
             color = region_subregion_country_area)) +
  geom_line() +
  scale_x_continuous(breaks = c(1950, 1980, 2000, 2020)) +
  labs(
    title = "Life Expectancy Gap between Women and Men (Years), 1950 to 2023",
    subtitle = "Calculated as Female Life Expectancy at Birth - Male Life Expectancy at Birth",
    x = "Year",
    y = "Life Expectancy Gap (Years)",
    color = "Region"
  ) +
  theme_classic()
```

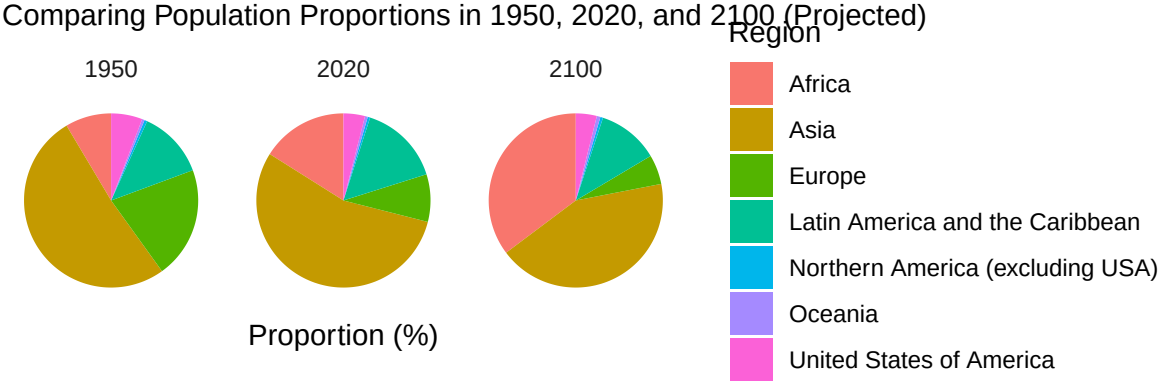



```
# 6 Jonathan
# Question: Comparing 1950 vs 2020 vs 2100, what proportion of global population growth did each region

#Regions: USA, North America - USA, Latin America & Caribbean, Europe, Asia, Africa, Oceania
graph6data <- estimates %>%
  rbind(mediums) %>%
  filter(region_subregion_country_area %in% c("United States of America", "Northern America", "Latin Ame
                                              "Europe", "Asia", "Africa", "Oceania")) %>%
  filter(year %in% c(1950, 2020, 2100)) %>%
  group_by(year, region_subregion_country_area) %>%
  summarise(total_population = sum(total_pop_january_thousands, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
# Remove USA from North America population
graph6data <- graph6data %>%
  mutate(total_population = ifelse(region_subregion_country_area == "Northern America", total_populatio
  group_by(year) %>%
  mutate(global_population = sum(total_population, na.rm = TRUE)) %>%
  mutate(proportion = total_population / global_population * 100) %>%
  mutate(region_subregion_country_area = ifelse(region_subregion_country_area == "Northern America", "N



# pie chart
graph6data %>%
  ggplot(aes(x = "", y = proportion, fill = region_subregion_country_area)) +
```

```
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  facet_wrap(~year) +
  labs(
    title = "Proportion of Global Population by Region",
    subtitle = "Comparing Population Proportions in 1950, 2020, and 2100 (Projected)",
    fill = "Region",
    x = "",
    y = "Proportion (%)"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
  axis.ticks = element_blank(),
  panel.grid = element_blank())
```



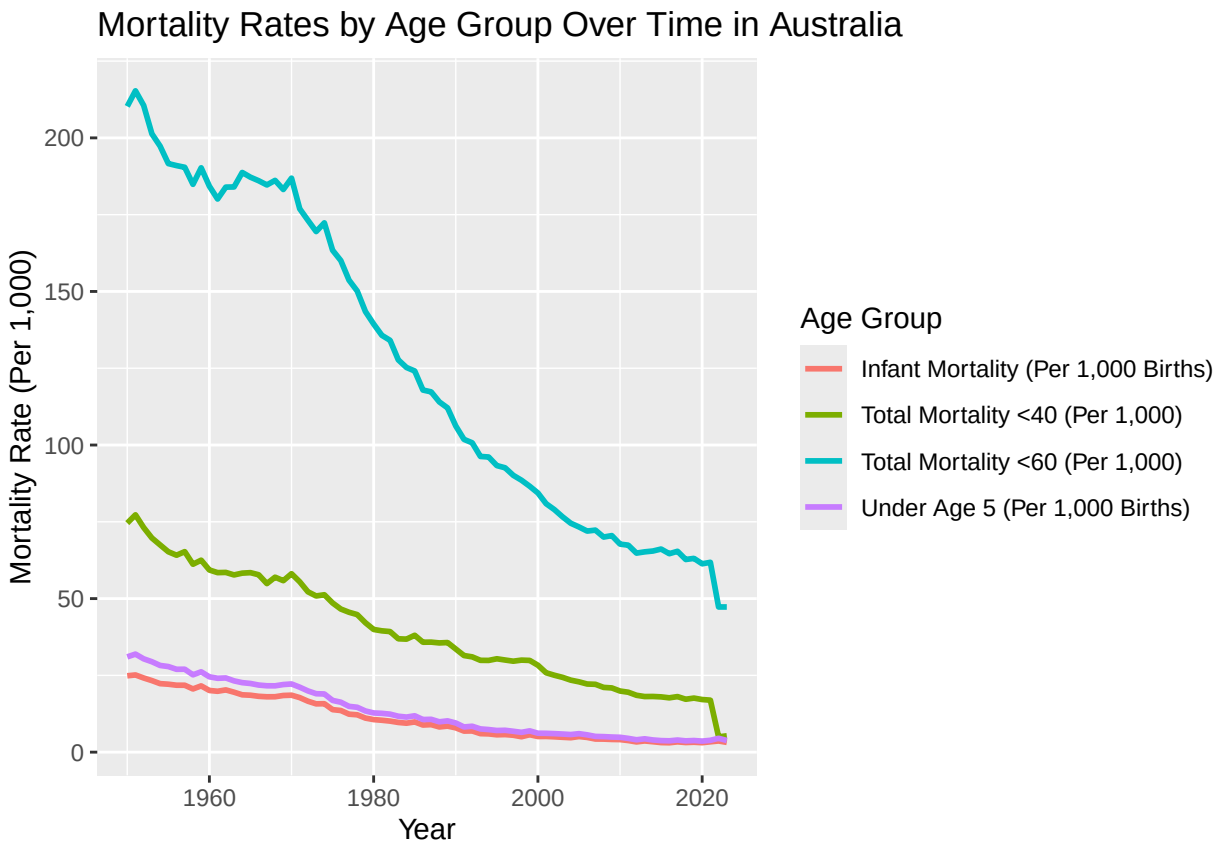

```
# 7 Alicia
# Question:
```

```
# 8 Alicia
# Question:
```

```
# 9 Anusha
# Question: How have mortality rates across different age groups changed over time in Australia?
australia_data <- estimates %>%
  filter(region_subregion_country_area == "Australia") %>%
  select(year,
         infant_morality_rate_infant_deaths_per_1000_births,
         "deaths_under_age_5_per_1,000_live_births",
         total_male_mortality_before_age_40_per_1000_births,
         total_mortality_before_age_60_per_1000_births) %>%
  pivot_longer(cols = starts_with("infant_morality_rate"):starts_with("total_mortality"),
               names_to = "age_group",
               values_to = "mortality_rate")

australia_data$age_group <- recode(australia_data$age_group,
                                   "infant_morality_rate_infant_deaths_per_1000_births" = "Infant Mortal
                                   "deaths_under_age_5_per_1,000_live_births" = "Under Age 5 (Per 1,000
                                   "total_male_mortality_before_age_40_per_1000_births" = "Total Mortali
                                   "total_mortality_before_age_60_per_1000_births" = "Total Mortality <
```

```
ggplot(australia_data, aes(x = year,
                           y = mortality_rate,
                           color = age_group,
                           group = age_group)) +
  geom_line(size = 1) +
  labs(title = "Mortality Rates by Age Group Over Time in Australia",
       x = "Year",
       y = "Mortality Rate (Per 1,000)",
       color = "Age Group")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

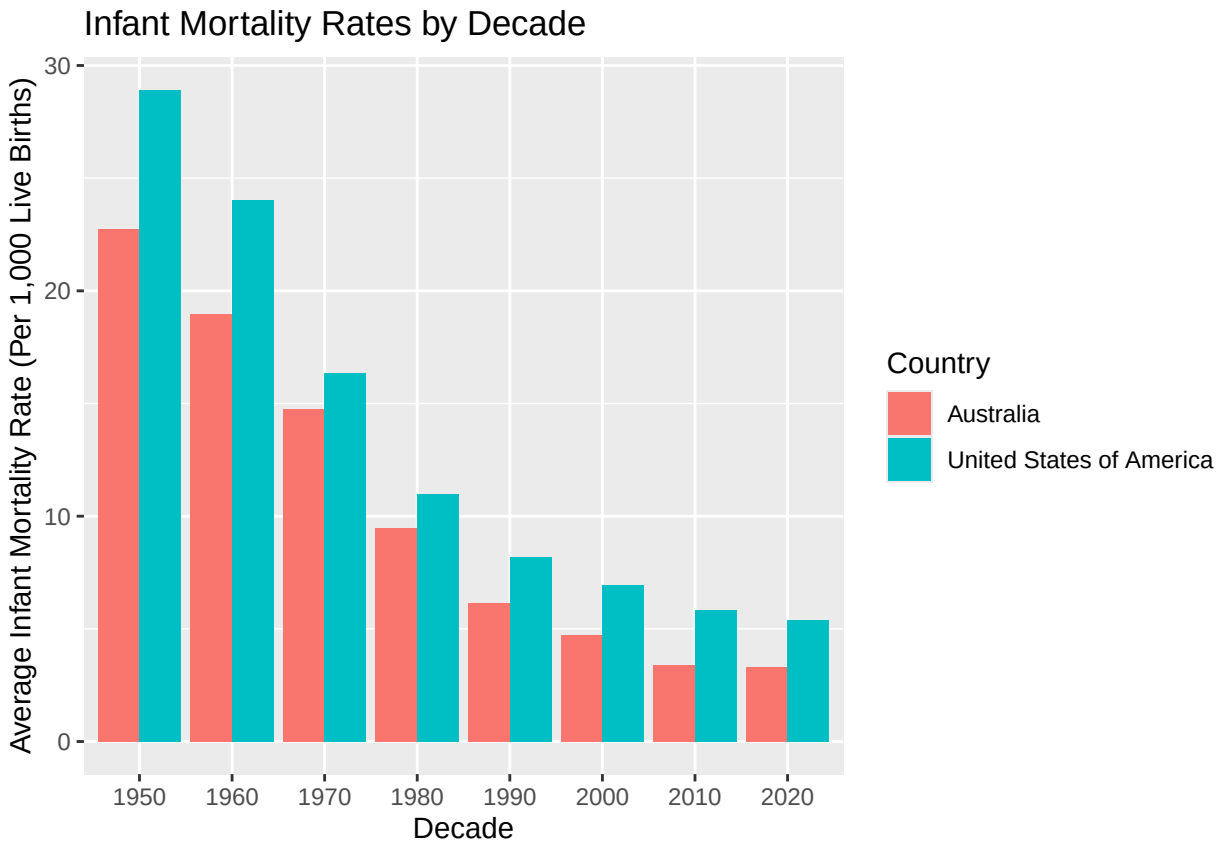



```
# 10 Anusha
# Question: How do the infant mortality rates in Australia and the United States compare across differe

estimates <- estimates %>%
  mutate(decade = floor(year / 10) * 10)

filtered_data <- estimates %>%
  filter(region_subregion_country_area %in% c("Australia", "United States of America")) %>%
  group_by(decade, region_subregion_country_area) %>%
  summarise(avg_infant_mortality_rate = mean(infant_morality_rate_infant_deaths_per_1000_births, na.rm =
```

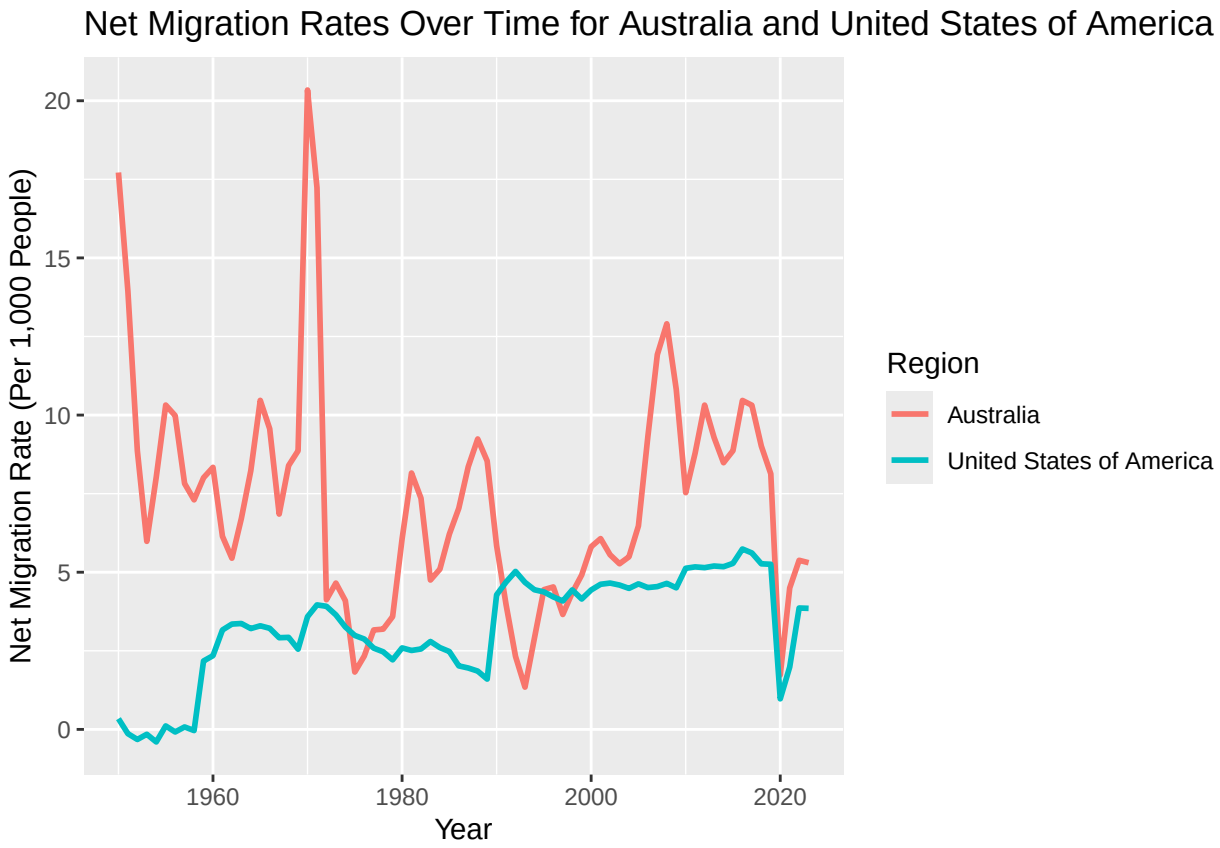
```
## `summarise()` has grouped output by 'decade'. You can override using the
```

```
## `.groups` argument.
```

```
ggplot(filtered_data, aes(x = factor(decade),
                          y = avg_infant_mortality_rate,
                          fill = region_subregion_country_area)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Infant Mortality Rates by Decade",
       x = "Decade",
       y = "Average Infant Mortality Rate (Per 1,000 Live Births)",
       fill = "Country")
```




```
# 11 Anusha
# Question: How have net migration rates changed over time in Australia vs United States?

filtered_data <- estimates %>%
  filter(region_subregion_country_area %in% c("Australia", "United States of America")) %>%
  select(year, region_subregion_country_area, net_migration_rate_per_1000)

ggplot(filtered_data, aes(x = year,
                          y = net_migration_rate_per_1000,
                          color = region_subregion_country_area,
                          group = region_subregion_country_area)) +
  geom_line(size = 1) +
  labs(title = "Net Migration Rates Over Time for Australia and United States of America",
       x = "Year",
       y = "Net Migration Rate (Per 1,000 People)",
       color = "Region")
```




```
# 12 Alicia
# Question:
```

5. Requirement-5 (2 pt) Having developed a strong understanding of your data, you'll now create a machine learning (ML) model to predict a specific metric. This involves selecting the most relevant variables from your dataset.

The UN's World Population Prospects provides a range of projected scenarios of population change. These rely on different assumptions in fertility, mortality and/or migration patterns to explore different demographic futures. Check this link for more info: https://population.un.org/wpp/DefinitionOfProjectionScenarios

You can choose to predict the same metric the UN provides (e.g., future population using fertility, mortality, and migration data). Compare your model's predictions to the UN's.

How significantly do your population projections diverge from those of the United Nations? Provide a comparison of the two. If you choose a different projection for which there is no UN data to compare with, then this comparison is not required.

6. Requirement-5 (1 pt)

Conclusion

Your analysis should conclude with a summary of key findings. I'm especially interested in any novel insights you uncover that go beyond the article's original conclusions.

7. Extra Credit (1 pt) Develop an interactive Shiny app to visualize your machine learning model's projections. The app must include at least one interactive widget (e.g., dropdown, radio buttons, text input) allowing users to select a variable value (such as country/region) and view the corresponding projections.

Submission

- You will upload the zip file containing finals.Rmd file and its PDF as a deliverable to Canvas. If you created a shiny app for predictions, you will add those files also to your zip file.
- You will present your findings by creating a video of a maximum 15 minutes duration, explaining the code and the workings of your project; all team members should explain their part in the project to receive credit. You will share the URL of the video on Canvas for us to evaluate. An ideal way to create this video would be to start a Zoom meeting, start recording, and then every member share their screen and explain their contribution.

It is not necessary to prepare slides (if you do it doesn't hurt) for the presentation. You may speak by showing the diagrams and/or code from your Posit project. Every team member should explain their part in the project along with the insights they derived by explaining the charts and summaries for full credit to each member.

Your project will be evaluated for clean code, meaningful/insightful EDA and predictions.

Note:

- Each plot must be accompanied by a summary that clarifies the rationale behind its creation and what insights the plot unveils. Every diagram should possess standalone significance, revealing something more compelling than the other charts
- After the deadline, instructors will select the top three outstanding analytics projects. The teams responsible for these exceptional analyses will have their video shared with the class

We will not accept submissions after the deadline; December 10th 4 pm