National Intelligence Quotient (IQ) Scores

Peyton Hall

08/06/2024

Load Necessary Libraries

```
library(readr)
library(ggplot2)
library(rworldmap)
```

```
## Loading required package: sp

## ### Welcome to rworldmap ###

## For a short introduction type :   vignette('rworldmap')
```

```
library(RColorBrewer)
library(dplyr)
```

```
##
## 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
```

Read the CSV file

```
avgIQpercountry <- read_csv("~/Desktop/NationalIQScores/avgIQpercountry.csv")
```

```
## Rows: 193 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Country, Continent, Population - 2023
## dbl (7): Rank, Average IQ, Literacy Rate, Nobel Prices, HDI (2021), Mean yea...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
avgIQpercountry
```

```
## # A tibble: 193 x 10
##     Rank Country       `Average IQ` Continent `Literacy Rate` `Nobel Prices`
##    <dbl> <chr>                <dbl> <chr>                <dbl>          <dbl>
##  1     1 Japan                 106. Asia                 0.99             29
##  2     2 Taiwan                106. Asia                 0.96              4
##  3     3 Singapore             106. Asia                 0.97              0
##  4     4 Hong Kong             105. Asia                 0.94              1
##  5     5 China                 104. Asia                 0.96              8
##  6     6 South Korea           102. Asia                 0.98              0
##  7     7 Belarus               102. Europe               1                 2
##  8     8 Finland               101. Europe               1                 5
##  9     9 Liechtenstein         101. Europe               1                 0
## 10    10 Germany               101. Europe               0.99            111
## # i 183 more rows
## # i 4 more variables: `HDI (2021)` <dbl>,
## #   `Mean years of schooling - 2021` <dbl>, `GNI - 2021` <dbl>,
## #   `Population - 2023` <chr>
```

Define Global Variables

```
# Save the value of the column header "Average IQ" to the variable Average_IQ
Average_IQ <- avgIQpercountry$`Average IQ`

# Pass Average_IQ into the iq_colors function
iq_colors <- function(Average_IQ) {
  if (is.na(Average_IQ)) {
    return("gray")
    }
  else if (Average_IQ < 50) {
    return("#8B0000")
    }
  else if (50 <= Average_IQ & Average_IQ < 60) {
    return("#FF4500")
    }
  else if (60 <= Average_IQ & Average_IQ < 70) {
    return("#FFA500")
    }
  else if (70 <= Average_IQ & Average_IQ < 80) {
    return("#F5DEB3")
    }
  else if (80 <= Average_IQ & Average_IQ < 90) {
    return("#ADD8E6")
    }
  else if (90 <= Average_IQ & Average_IQ < 100) {
    return("#7B68EE")
  }
  else {
    return("#0000CD")
    }
} # end iq_colors
```

```r
iq_labels <- function(Average_IQ) {
  if (is.na(Average_IQ)) {
    return("N/A")
    }
  else if (Average_IQ < 50) {
    return("<50")
    }
  else if (50 <= Average_IQ & Average_IQ < 60) {
    return("50-59")
    }
  else if (60 <= Average_IQ & Average_IQ < 70) {
    return("60-69")
    }
  else if (70 <= Average_IQ & Average_IQ < 80) {
    return("70-79")
    }
  else if (80 <= Average_IQ & Average_IQ < 90) {
    return("80-89")
    }
  else if (90 <= Average_IQ & Average_IQ < 100) {
    return("90-99")
  }
  else {
    return("100+")
    }
} # end iq_labels

iq_breaks <- c(-Inf, 50, 60, 70, 80, 90, 100, Inf)  # Define the breaks
```

Global Map of Average IQ per Country

```r
# Set graphical parameters to use Times New Roman for the title
par(family = "serif")

# Join the country data to the map
sPDF <- joinCountryData2Map(avgIQpercountry, joinCode = "NAME",
                            nameJoinColumn = "Country")
```

```
## 189 codes from your data successfully matched countries in the map
## 4 codes from your data failed to match with a country code in the map
## 54 codes from the map weren't represented in your data
```

```r
# Apply the color function to create a color palette for the map
sPDF$color <- sapply(sPDF$`Average IQ`, iq_colors)

# Define the legend text based on the IQ values
legend_labels <- unique(sapply(sort(unique(sPDF$`Average IQ`)), iq_labels))
legend_colors <- unique(sapply(sort(unique(sPDF$`Average IQ`)), iq_colors))

# Plot the map
mapParams <- mapCountryData(sPDF,
                            nameColumnToPlot = "Average IQ",
```

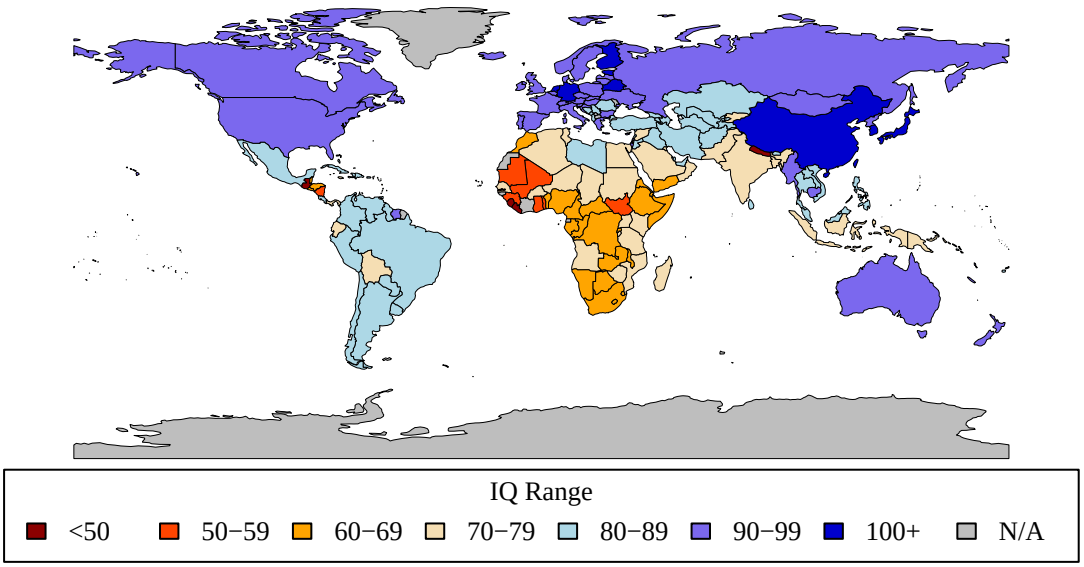
```
                            mapTitle = "Global Map of Average IQ per Country",
                            colourPalette = legend_colors,
                            catMethod = iq_breaks,
                            oceanCol = "#FFFFFF",
                            missingCountryCol = "gray",
                            addLegend = FALSE,
                            borderCol = "#000000",
                            xlim = c(-180, 180), ylim = c(-90, 90))

# Add "N/A" to the legend manually
legend_labels <- c(legend_labels, "N/A")
legend_colors <- c(legend_colors, "gray")

# Add the legend manually at the bottom
legend("bottom",
       legend = legend_labels,
       fill = legend_colors,
       title = "IQ Range",
       horiz = TRUE,
       cex = 0.8,
       inset = c(0, -0.16),
       xpd = TRUE)  # Allow legend to be drawn outside plot area

# Add caption
mtext("Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.",
      side = 1, line = 2.5, adj = 0.5, cex = 1, family = "serif")
```

Global Map of Average IQ per Country



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

The global map provides a broad overview of what IQ scores look like per country.

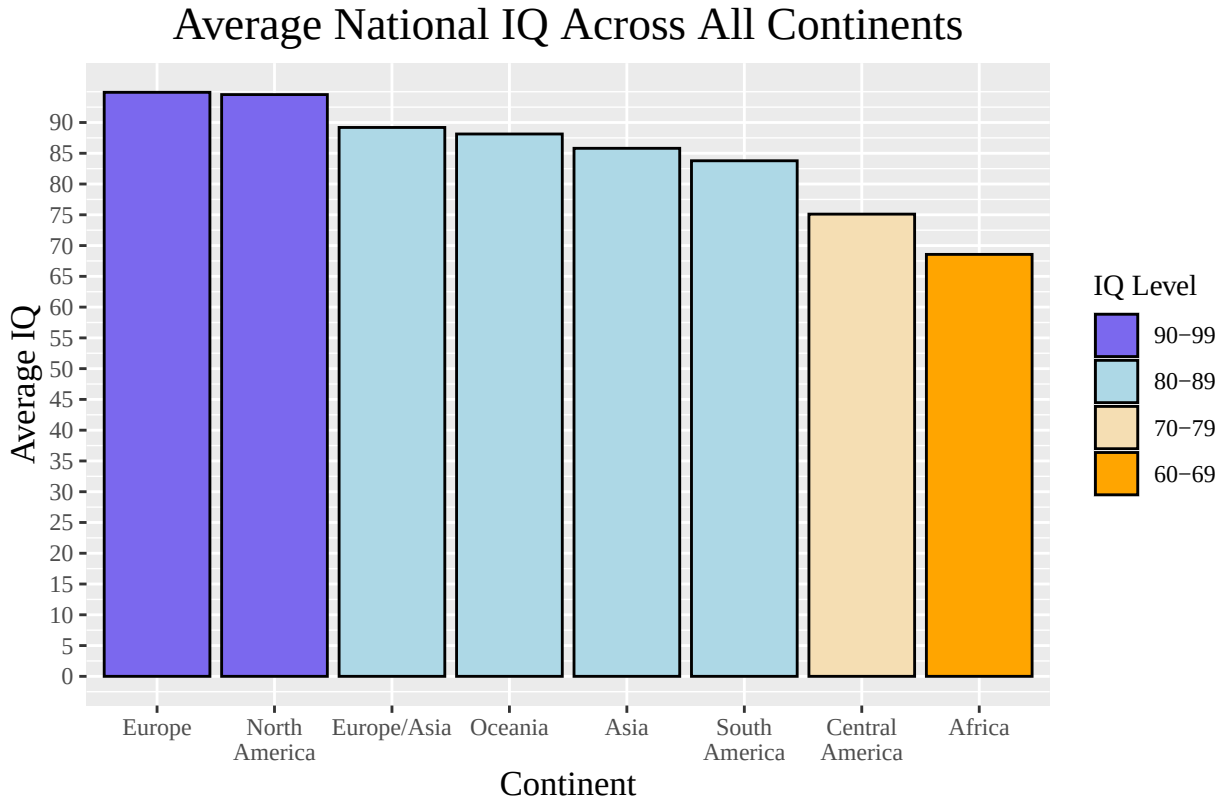
Compare Average National IQ Across All Continents

```
# Calculate average IQ for each continent
avg_continent_iq <- avgIQpercountry %>%
  group_by(Continent) %>%
  summarize(Average_IQ = mean(`Average IQ`, na.rm = TRUE)) %>%
  mutate(Color = sapply(Average_IQ, iq_colors),  # Call global function iq_colors
         Label = sapply(Average_IQ, iq_labels)) %>%
  arrange(desc(Average_IQ))  # Sort continents in descending order of Average IQ
avg_continent_iq # print the calculated results
```

```
## # A tibble: 8 x 4
##   Continent       Average_IQ Color   Label
##   <chr>                <dbl> <chr>   <chr>
## 1 Europe                94.9 #7B68EE 90-99
## 2 North America         94.5 #7B68EE 90-99
## 3 Europe/Asia           89.2 #ADD8E6 80-89
## 4 Oceania               88.1 #ADD8E6 80-89
## 5 Asia                  85.8 #ADD8E6 80-89
## 6 South America         83.8 #ADD8E6 80-89
## 7 Central America       75.1 #F5DEB3 70-79
## 8 Africa                68.6 #FFA500 60-69
```

```r
# Adjusting the labels for better display on the graph
avg_continent_iq$Continent <- gsub("North America", "North\nAmerica",
                                   avg_continent_iq$Continent)
avg_continent_iq$Continent <- gsub("South America", "South\nAmerica",
                                   avg_continent_iq$Continent)
avg_continent_iq$Continent <- gsub("Central America", "Central\nAmerica",
                                   avg_continent_iq$Continent)

ggplot(avg_continent_iq, aes(x = reorder(Continent, -Average_IQ), y = Average_IQ,
                             fill = Label)) +
  geom_bar(stat = "identity", color = "black") +
  scale_fill_manual(values = c("60-69" = "#FFA500", "70-79" = "#F5DEB3",
                               "80-89" = "#ADD8E6", "90-99" = "#7B68EE"),
                    name = "IQ Level") +  # Change the legend title here
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  scale_y_continuous(breaks = seq(0, 90, by = 5)) +
  labs(title = "Average National IQ Across All Continents",
       x = "Continent",
       y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(1)), # x-axis labels
        axis.text.y = element_text(size = rel(1)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1))) # Center caption
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

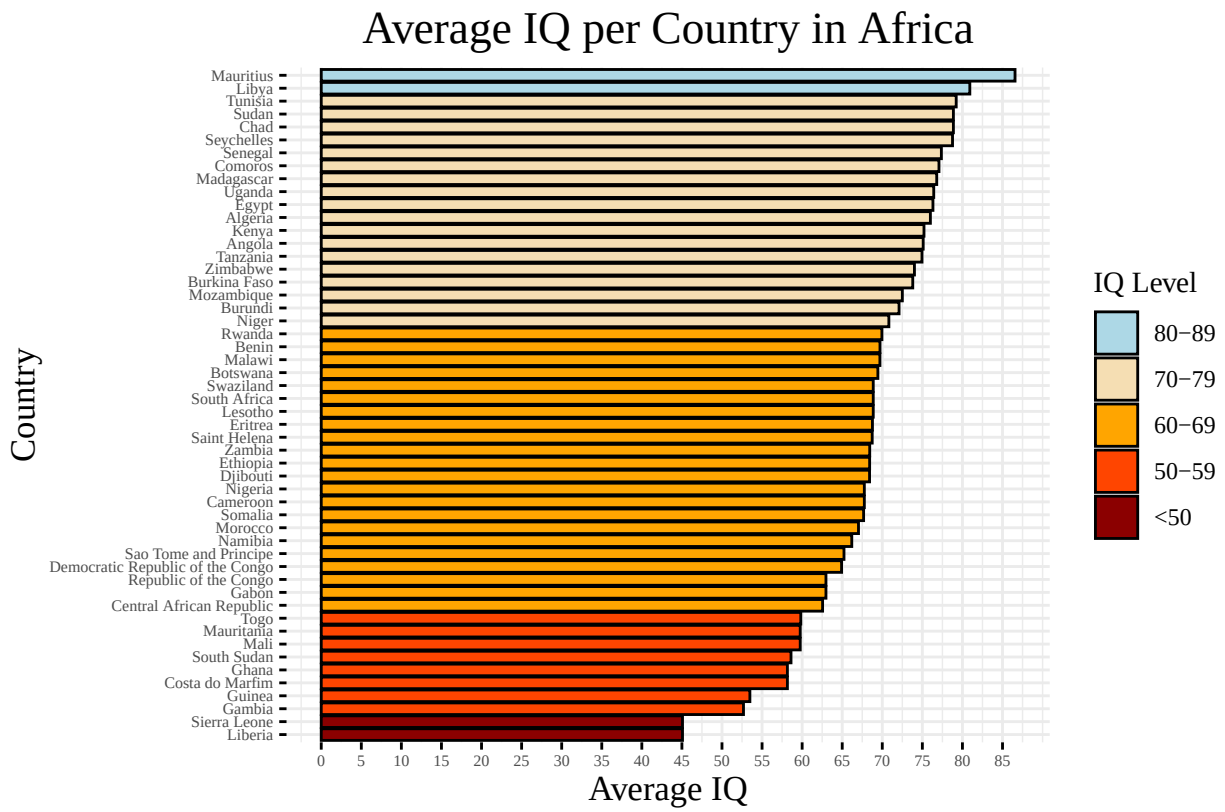
Bar Graph of Average IQ per Country In Africa

```
# Filter the dataset for African countries
africa_data <- avgIQpercountry %>%
  filter(Continent == "Africa") %>%
  mutate(IQ_level = sapply(`Average IQ`, iq_labels))

# Create the color palette based on unique IQ levels in africa_data
iq_color_values <- sapply(unique(africa_data$IQ_level), iq_colors)
names(iq_color_values) <- unique(africa_data$IQ_level)

# Create bar graph
ggplot(africa_data, aes(x = reorder(Country, `Average IQ`), y = `Average IQ`,
                        fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000") +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in Africa", x = "Country",
       y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(africa_data$`Average IQ`, na.rm = TRUE),
                                  by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(0.7)), # x-axis labels
```

```
        axis.text.y = element_text(size = rel(0.7)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

Bar Graph of Average IQ per Country In Asia

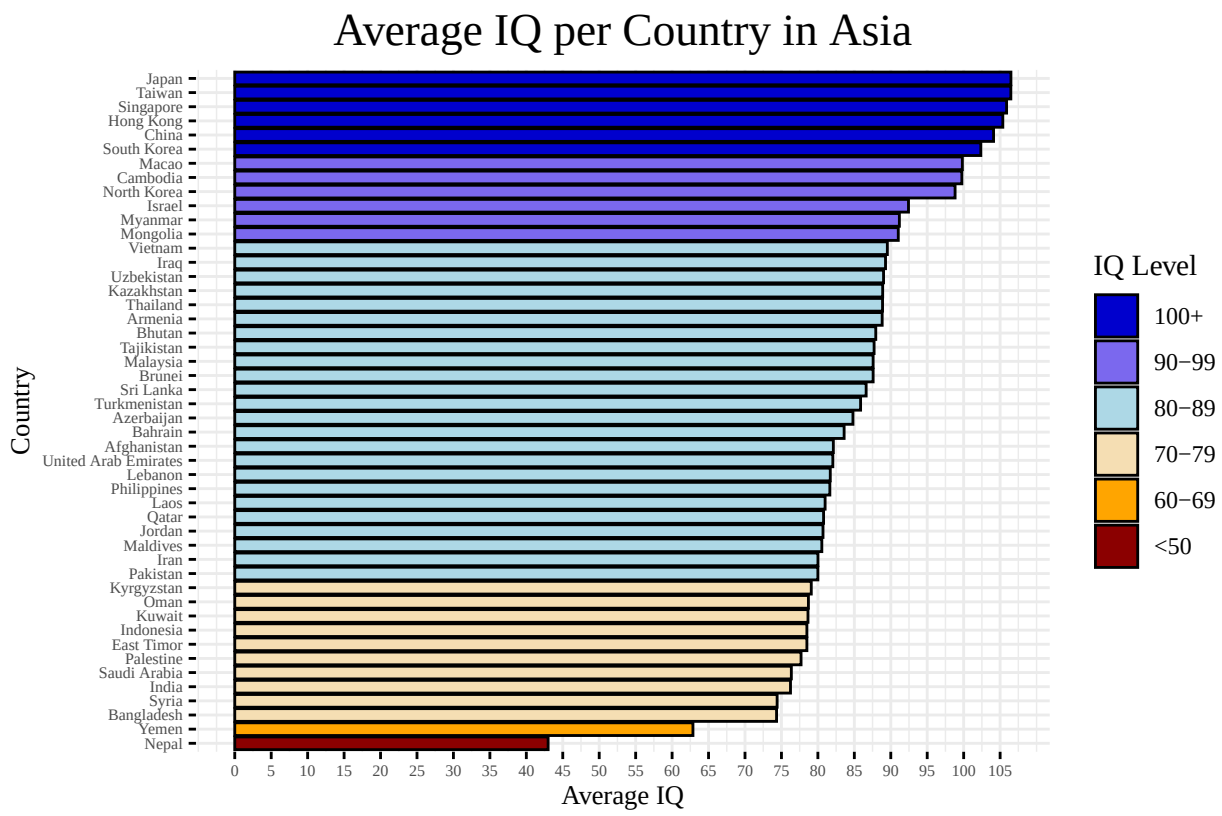
```
# Define the order of levels for IQ_level factor
iq_levels_ordered <- c("<50", "60-69", "70-79", "80-89", "90-99", "100+")

# Filter the dataset for Asian countries
asia_data <- avgIQpercountry %>%
  filter(Continent == "Asia") %>%
  mutate(IQ_level = factor(sapply(`Average IQ`, iq_labels),
                           levels = iq_levels_ordered))

# Define the color palette
iq_color_values <- c("<50" = "#8B0000", "60-69" = "#FFA500", "70-79" = "#F5DEB3",
                     "80-89" = "#ADD8E6", "90-99" = "#7B68EE", "100+" = "#0000CD")

# Create bar graph
ggplot(asia_data, aes(x = reorder(Country, `Average IQ`), y = `Average IQ`,
```

```
                      fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000") +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in Asia", x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(asia_data$`Average IQ`, na.rm = TRUE),
                                  by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(0.7)), # x-axis labels
        axis.text.y = element_text(size = rel(0.7)), # y-axis labels
        axis.title.x = element_text(size = rel(0.9)), # x-axis title
        axis.title.y = element_text(size = rel(0.9)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

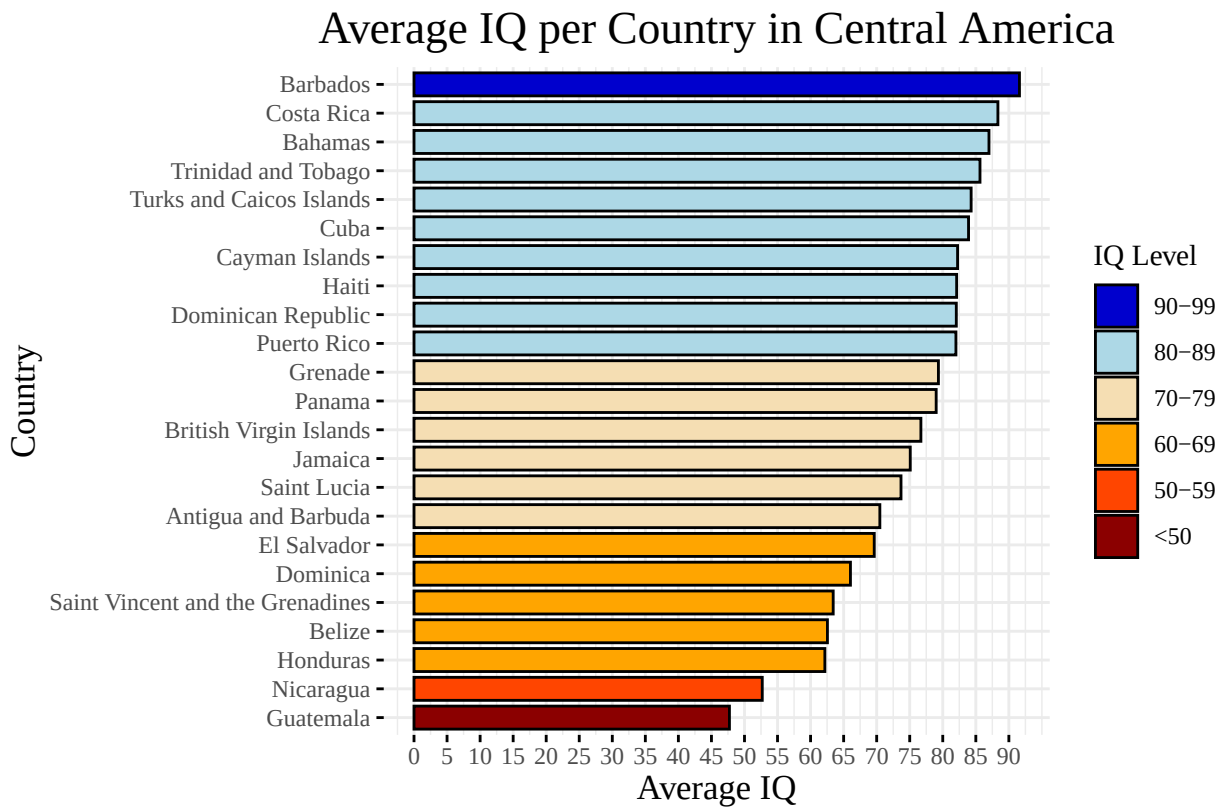
Bar Graph of Average IQ per Country In Central America

```
# Filter the dataset for Central American countries
centralamerica_data <- avgIQpercountry %>%
  filter(Continent == "Central America") %>%
```

```r
  mutate(IQ_level = sapply(`Average IQ`, iq_labels))

# Create the color palette based on unique IQ levels in centralamerica_data
iq_color_values <- sapply(unique(centralamerica_data$IQ_level), iq_colors)
names(iq_color_values) <- unique(centralamerica_data$IQ_level)

# Create bar graph
ggplot(centralamerica_data, aes(x = reorder(Country, `Average IQ`),
                                y = `Average IQ`, fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000", width = 0.8) +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in Central America",
       x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(centralamerica_data$`Average IQ`,
                                         na.rm = TRUE), by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(1)), # x-axis labels
        axis.text.y = element_text(size = rel(1)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

Bar Graph of Average IQ per Country In Europe

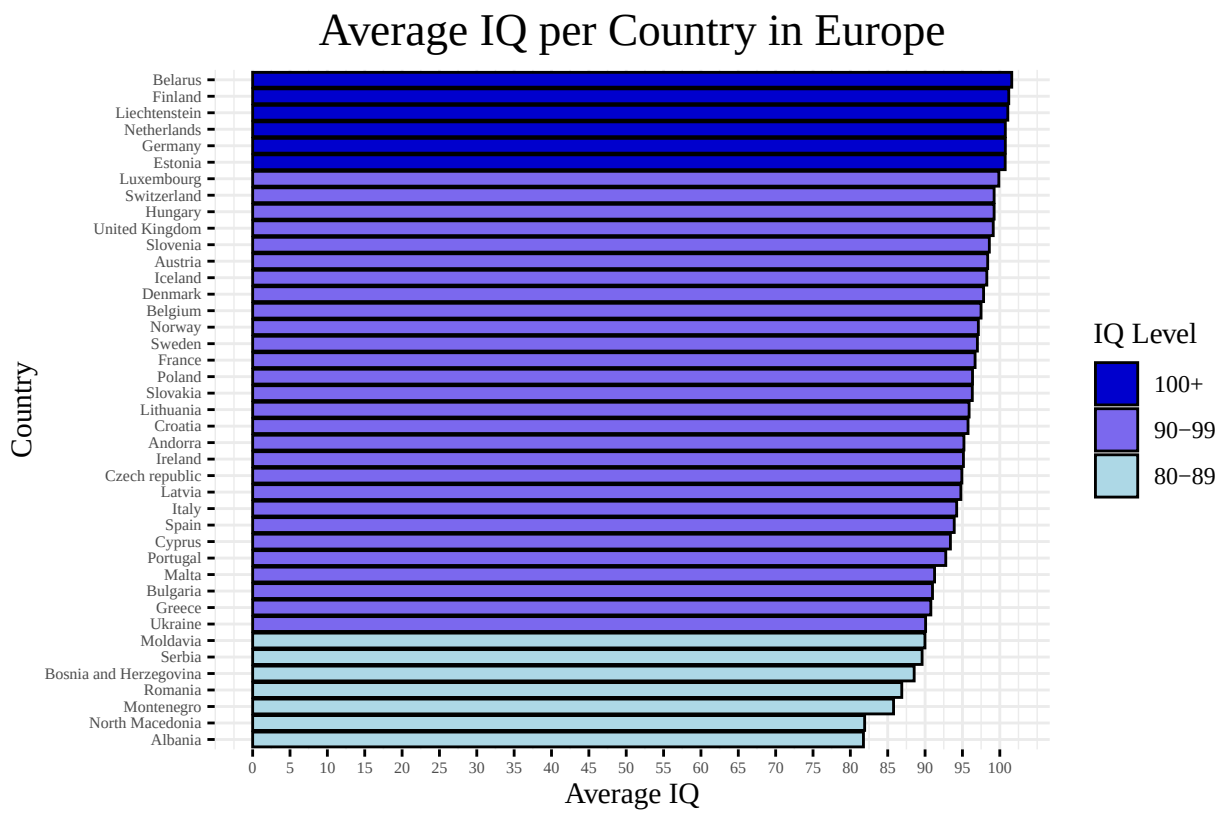
```
# Define the order of levels for IQ_level factor
iq_levels_ordered <- c("80-89", "90-99", "100+")

# Filter the dataset for European countries
europe_data <- avgIQpercountry %>%
  filter(Continent == "Europe") %>%
  mutate(IQ_level = factor(sapply(`Average IQ`, iq_labels),
                           levels = iq_levels_ordered))

# Define the color palette
iq_color_values <- c("80-89" = "#ADD8E6",
                     "90-99" = "#7B68EE",
                     "100+" = "#0000CD")

# Create bar graph
ggplot(europe_data, aes(x = reorder(Country, `Average IQ`),
                        y = `Average IQ`, fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000") +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in Europe",
       x = "Country", y = "Average IQ",
```

```
      caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(europe_data$`Average IQ`, na.rm = TRUE),
                                  by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(0.7)), # x-axis labels
        axis.text.y = element_text(size = rel(0.7)), # y-axis labels
        axis.title.x = element_text(size = rel(1)), # x-axis title
        axis.title.y = element_text(size = rel(1)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



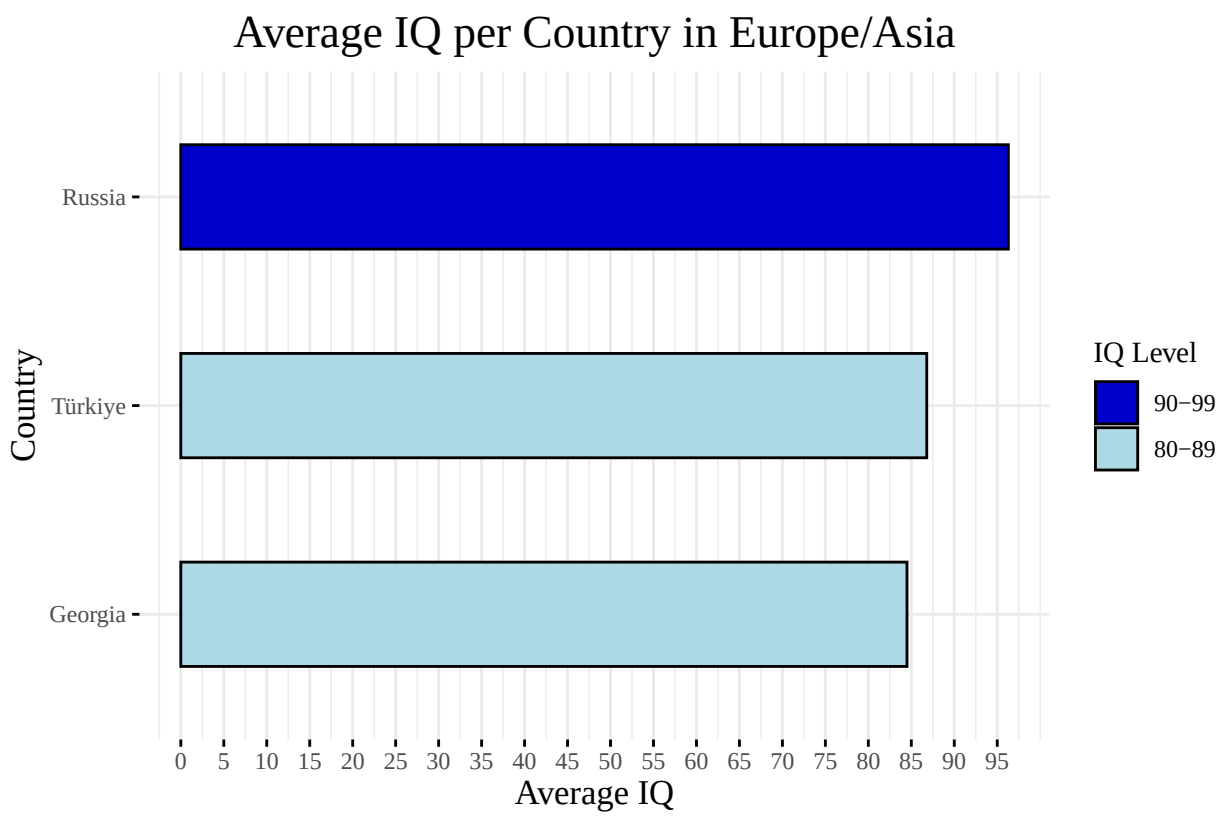
Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

Bar Graph of Average IQ per Country In Europe/Asia

```
# Filter the dataset for Eurasian countries
eurasian_data <- avgIQpercountry %>%
  filter(Continent == "Europe/Asia") %>%
  mutate(Country = ifelse(Country == "Turkey", "Türkiye", Country),
         IQ_level = sapply(`Average IQ`, iq_labels))

# Create the color palette based on unique IQ levels in eurasian_data
iq_color_values <- sapply(unique(eurasian_data$IQ_level), iq_colors)
names(iq_color_values) <- unique(eurasian_data$IQ_level)
```

```r
# Create bar graph
ggplot(eurasian_data, aes(x = reorder(Country, `Average IQ`),
                          y = `Average IQ`, fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000", width = 0.5) +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in Europe/Asia",
       x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(eurasian_data$`Average IQ`, na.rm = TRUE),
                                  by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(1)), # x-axis labels
        axis.text.y = element_text(size = rel(1)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



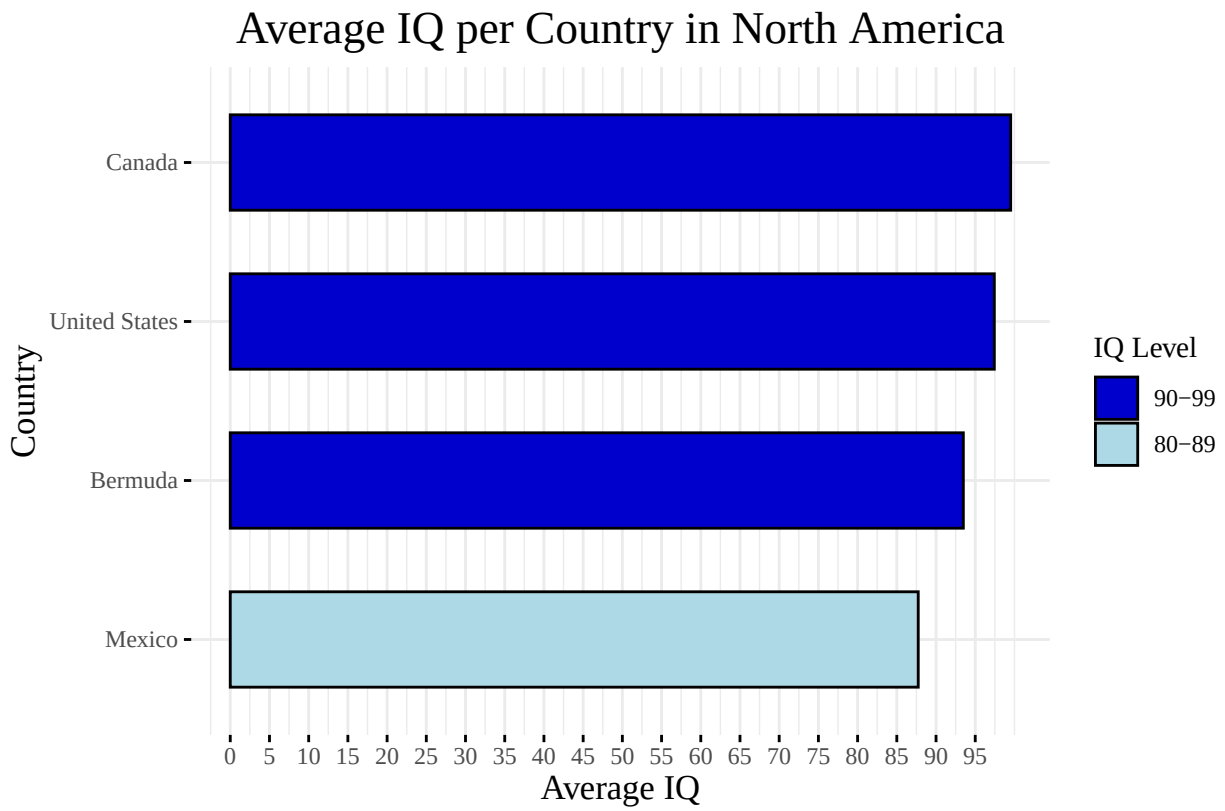
Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

Bar Graph of Average IQ per Country In North America

```r
# Filter the dataset for North American countries
northamerican_data <- avgIQpercountry %>%
  filter(Continent == "North America") %>%
  mutate(IQ_level = sapply(`Average IQ`, iq_labels))

# Create the color palette based on unique IQ levels in northamerican_data
iq_color_values <- sapply(unique(northamerican_data$IQ_level), iq_colors)
names(iq_color_values) <- unique(northamerican_data$IQ_level)

# Create bar graph
ggplot(northamerican_data, aes(x = reorder(Country, `Average IQ`),
                               y = `Average IQ`, fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000", width = 0.6) +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in North America",
       x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(northamerican_data$`Average IQ`,
                                         na.rm = TRUE), by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(1)), # x-axis labels
        axis.text.y = element_text(size = rel(1)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

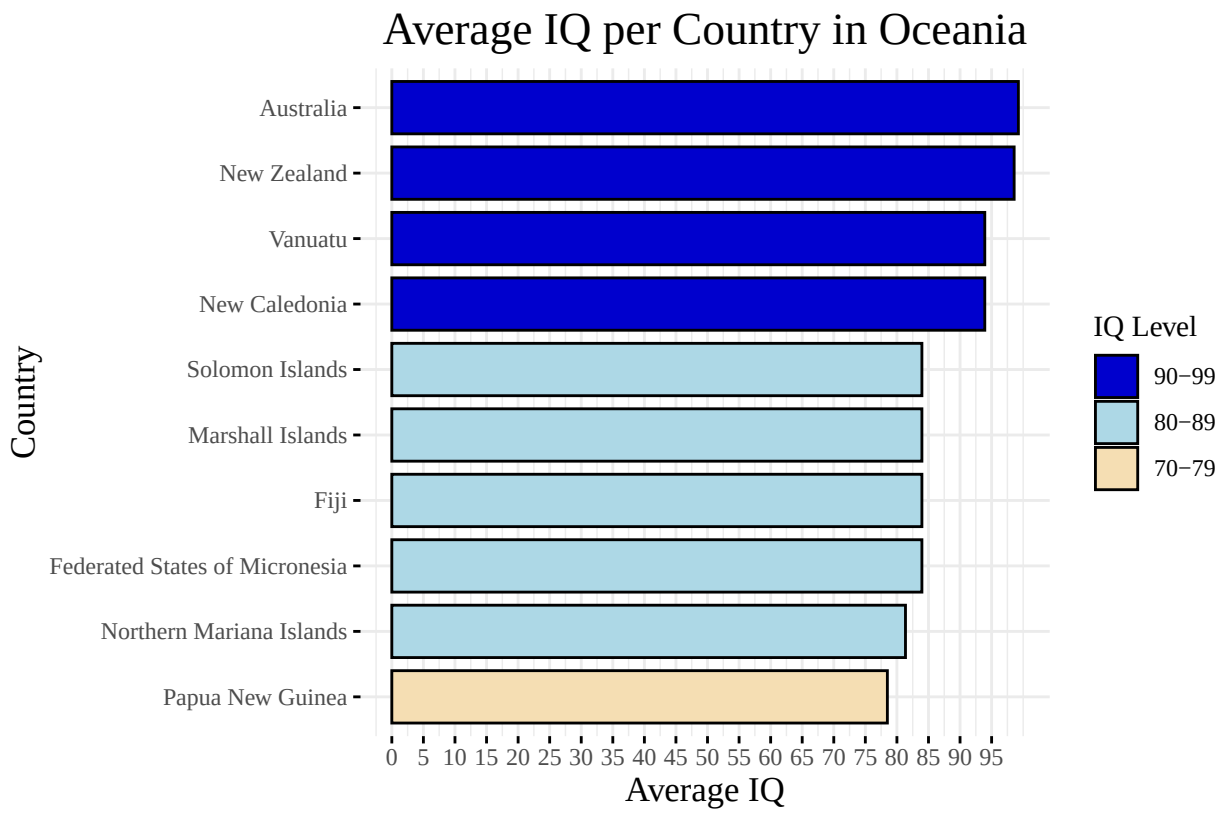
Bar Graph of Average IQ per Country In Oceania

```
# Filter the dataset for Oceanian countries
oceanian_data <- avgIQpercountry %>%
  filter(Continent == "Oceania") %>%
  mutate(IQ_level = sapply(`Average IQ`, iq_labels))

# Create the color palette based on unique IQ levels in oceanian_data
iq_color_values <- sapply(unique(oceanian_data$IQ_level), iq_colors)
names(iq_color_values) <- unique(oceanian_data$IQ_level)

# Create bar graph
ggplot(oceanian_data, aes(x = reorder(Country, `Average IQ`),
                          y = `Average IQ`, fill = IQ_level)) +
  geom_bar(stat = "identity", color = "black", width = 0.8) +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in Oceania",
       x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(oceanian_data$`Average IQ`, na.rm = TRUE),
                                  by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(1)), # x-axis labels
```

```
        axis.text.y = element_text(size = rel(1)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

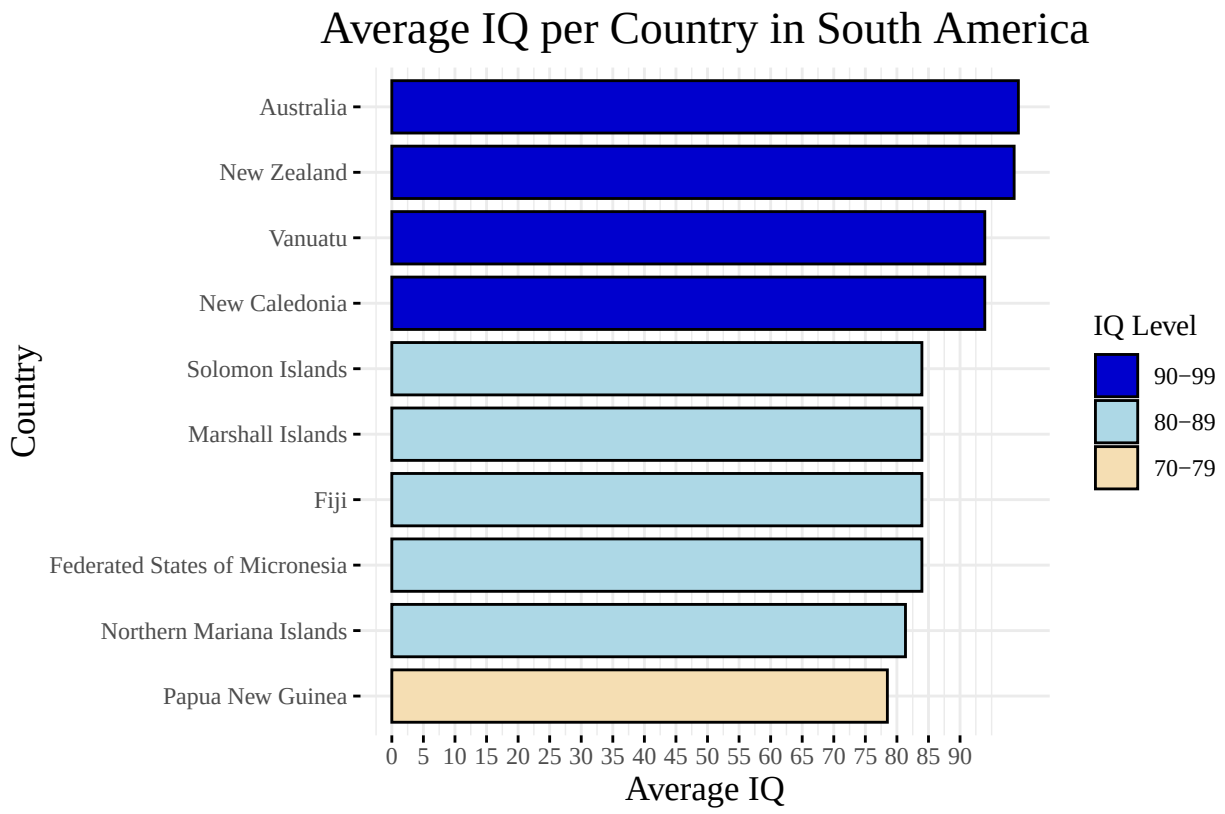
Bar Graph of Average IQ per Country In South America

```
# Filter the dataset for South American countries
southamerican_data <- avgIQpercountry %>%
  filter(Continent == "South America") %>%
  mutate(IQ_level = sapply(`Average IQ`, iq_labels))

# Create the color palette based on unique IQ levels in southamerican_data
iq_color_values <- sapply(unique(southamerican_data$IQ_level), iq_colors)
names(iq_color_values) <- unique(southamerican_data$IQ_level)

# Create bar graph
ggplot(oceanian_data, aes(x = reorder(Country, `Average IQ`), y = `Average IQ`,
                          fill = IQ_level)) +
  geom_bar(stat = "identity", color = "#000000", width = 0.8) +
  coord_flip() +
  theme_minimal() +
```

```
  scale_fill_manual(values = iq_color_values, name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "Average IQ per Country in South America",
       x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  scale_y_continuous(breaks = seq(0, max(southamerican_data$`Average IQ`, na.rm = TRUE), by = 5)) +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(size = rel(1)), # x-axis labels
        axis.text.y = element_text(size = rel(1)), # y-axis labels
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 1, size = rel(1)), # Center caption
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black"), # Add black ticks on y-axis
        legend.position = "right")
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

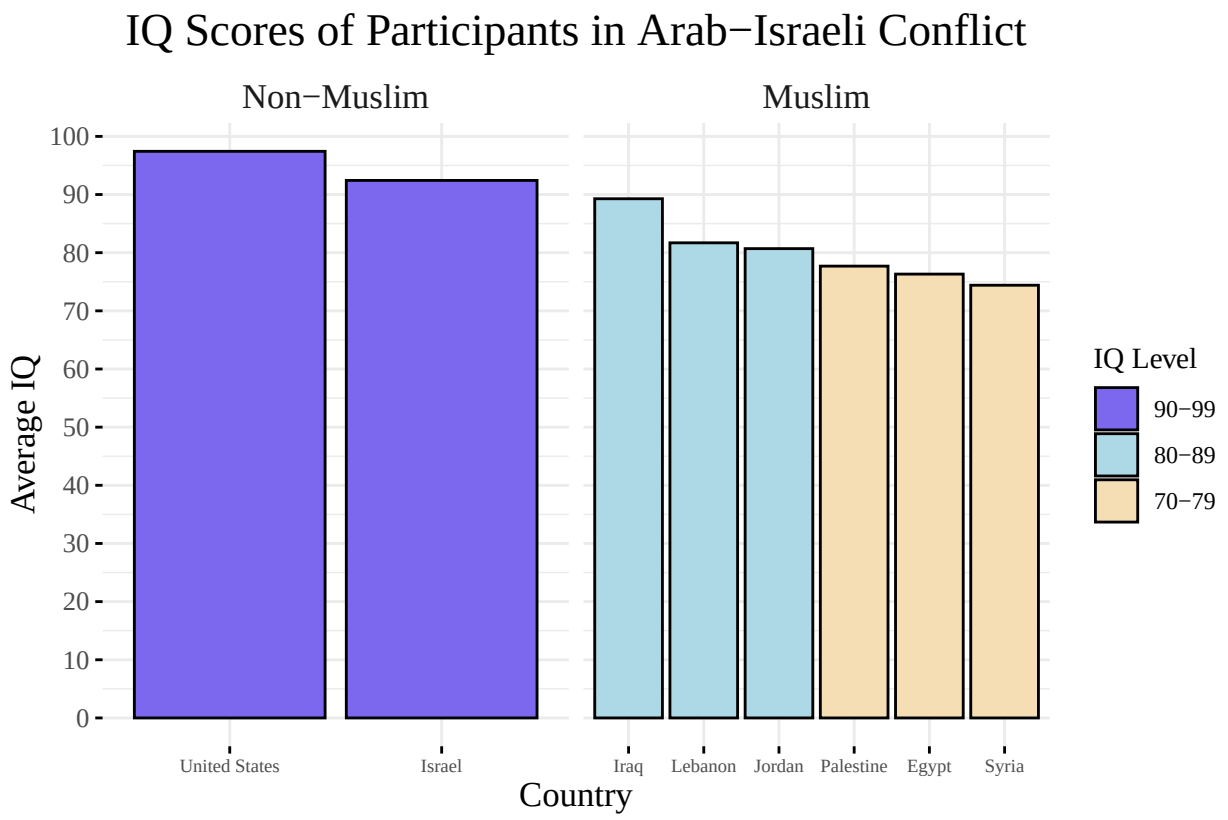
Compare IQ in Countries Involved in Arab-Israeli Conflict

```
# Define the grouping of countries and prepare the data
selected_countries_data <- avgIQpercountry %>%
  filter(Country %in% c("United States", "Israel",
                        "Palestine", "Syria",
                        "Jordan", "Egypt",
                        "Lebanon", "Iraq")) %>%
  select(Country, `Average IQ`) %>%
```

```r
  arrange(desc(`Average IQ`)) %>% # Sorting by descending IQ
  mutate(Color = sapply(`Average IQ`, iq_colors), # Apply the iq_colors function
         IQ_Range = sapply(`Average IQ`, iq_labels), # Apply the iq_labels function
         Category = factor(ifelse(Country %in% c("United States", "Israel"),
                                  "Non-Muslim", "Muslim"),
                           levels = c("Non-Muslim", "Muslim")))

# Ensure the order of factors in Country matches the descending IQ order
selected_countries_data$Country <- factor(selected_countries_data$Country,
                                          levels = selected_countries_data$Country)

# Graph their IQs with custom colors and relative text sizing
ggplot(selected_countries_data, aes(x = Country, y = `Average IQ`,
                                    fill = IQ_Range)) +
  geom_bar(stat = "identity", color = "#000000") +
  scale_fill_manual(values = setNames(as.character(selected_countries_data$Color),
                                      selected_countries_data$IQ_Range),
                    name = "IQ Level") +
  guides(fill = guide_legend(reverse = TRUE)) + # Reverse legend order
  labs(title = "IQ Scores of Participants in Arab-Israeli Conflict",
       x = "Country", y = "Average IQ",
       caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
  theme_minimal() +
  theme(text = element_text(family = "serif"),
        axis.text.x = element_text(hjust = 0.5, size = rel(0.8)),  # x-axis label
        axis.text.y = element_text(size = rel(1.1)), # y-axis label
        axis.title.x = element_text(size = rel(1.2)), # x-axis title
        axis.title.y = element_text(size = rel(1.2)), # y-axis title
        plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
        plot.caption = element_text(hjust = 0.5, size = rel(0.8)),# Center caption
        strip.text = element_text(size = rel(1.2)), # facet titles
        axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
        axis.ticks.y = element_line(color = "black")) + # Add black ticks on y-axis
  scale_y_continuous(breaks = seq(0, 100, by = 10)) + # y-axis ticks
  facet_wrap(~ Category, scales = "free_x") # Split graph into two categories
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.

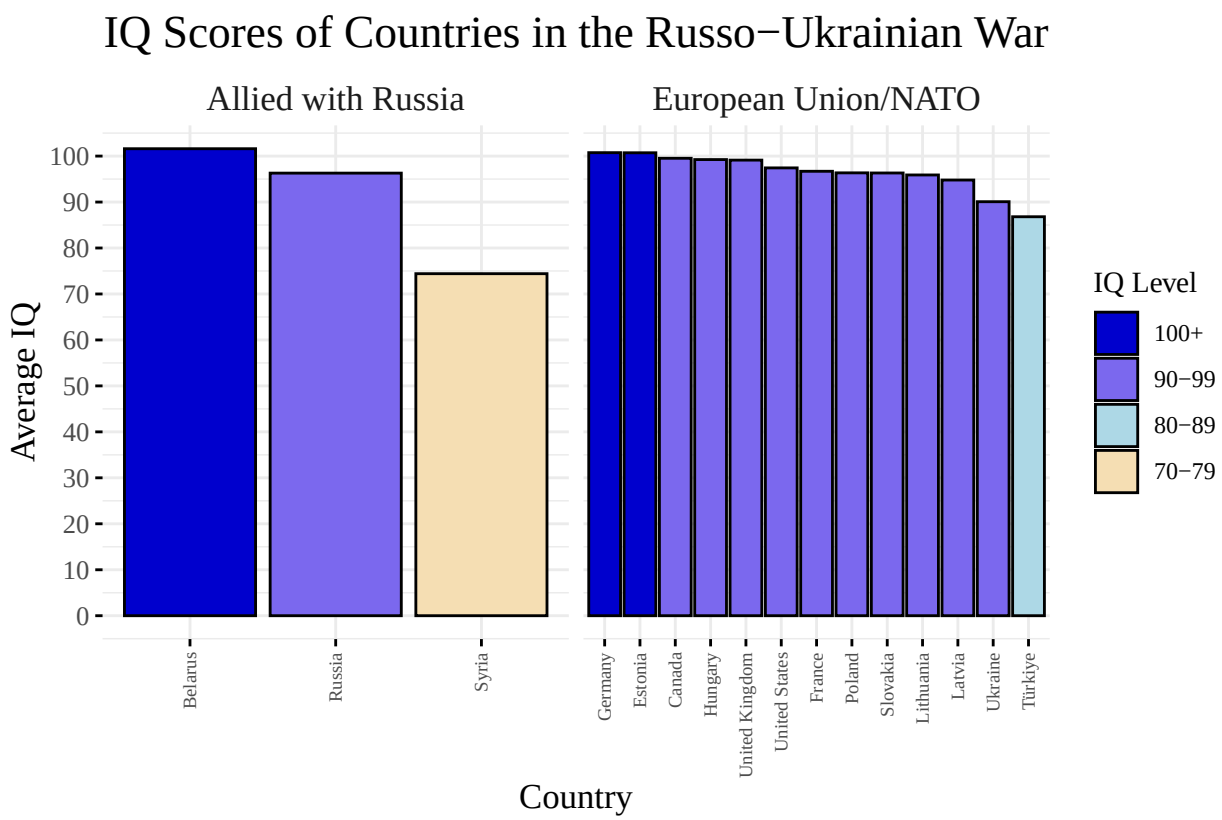
Compare IQ in Countries Involved in Russo-Ukrainian War

```
# Define the grouping of countries and prepare the data
selected_countries_data <- avgIQpercountry %>%
  filter(Country %in% c("Russia", "Syria", "Belarus", "Ukraine", "United States",
                        "United Kingdom", "Germany", "France", "Poland",
                        "Canada", "Turkey", "Estonia", "Latvia", "Lithuania",
                        "Moldova", "Hungary", "Slovakia")) %>%
  mutate(Country = gsub("Turkey", "Türkiye", Country)) %>% # Replace Turkey with Türkiye
  select(Country, `Average IQ`) %>%
  arrange(desc(`Average IQ`)) %>% # Sorting by descending IQ
  mutate(Color = sapply(`Average IQ`, iq_colors), # Apply the iq_colors function
         IQ_Range = sapply(`Average IQ`, iq_labels), # Apply the iq_labels function
         Category = factor(ifelse(Country %in% c("Russia", "Belarus", "Syria"),
                                  "Allied with Russia", "European Union/NATO"),
                           levels = c("Allied with Russia", "European Union/NATO")))

# Ensure the order of factors in Country matches the descending IQ order
selected_countries_data$Country <- factor(selected_countries_data$Country,
                                          levels = selected_countries_data$Country)

# Graph their IQs with custom colors and relative text sizing
ggplot(selected_countries_data, aes(x = Country, y = `Average IQ`,
                                    fill = IQ_Range)) +
  geom_bar(stat = "identity", color = "#000000") +
  scale_fill_manual(values = setNames(as.character(selected_countries_data$Color),
```

```
                                    selected_countries_data$IQ_Range),
                  name = "IQ Level",
                  breaks = c("100+", "90-99", "80-89", "70-79")) + # Order legend
guides(fill = guide_legend(reverse = FALSE)) + # Maintain specified order
labs(title = "IQ Scores of Countries in the Russo-Ukrainian War",
     x = "Country", y = "Average IQ",
     caption = "Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.") +
theme_minimal() +
theme(text = element_text(family = "serif"),
      axis.text.x = element_text(angle = 90, hjust = 1,
                                 vjust = .5, size = rel(0.8)), # x-axis label
      axis.text.y = element_text(size = rel(1.1)), # y-axis label
      axis.title.x = element_text(size = rel(1.2)), # x-axis title
      axis.title.y = element_text(size = rel(1.2)), # y-axis title
      plot.title = element_text(hjust = 0.5, size = rel(1.5)), # Center title
      plot.caption = element_text(hjust = 0.5, size = rel(0.8)), # Center caption
      strip.text = element_text(size = rel(1.2)), # facet titles
      axis.ticks.x = element_line(color = "black"), # Add black ticks on x-axis
      axis.ticks.y = element_line(color = "black")) + # Add black ticks on y-axis
scale_y_continuous(breaks = seq(0, 100, by = 10)) + # y-axis ticks
facet_wrap(~ Category, scales = "free_x") # Split graph into two categories
```



Source: Kaggle/ Google LLC (2023) Average Global IQ per Country with Other Stats.