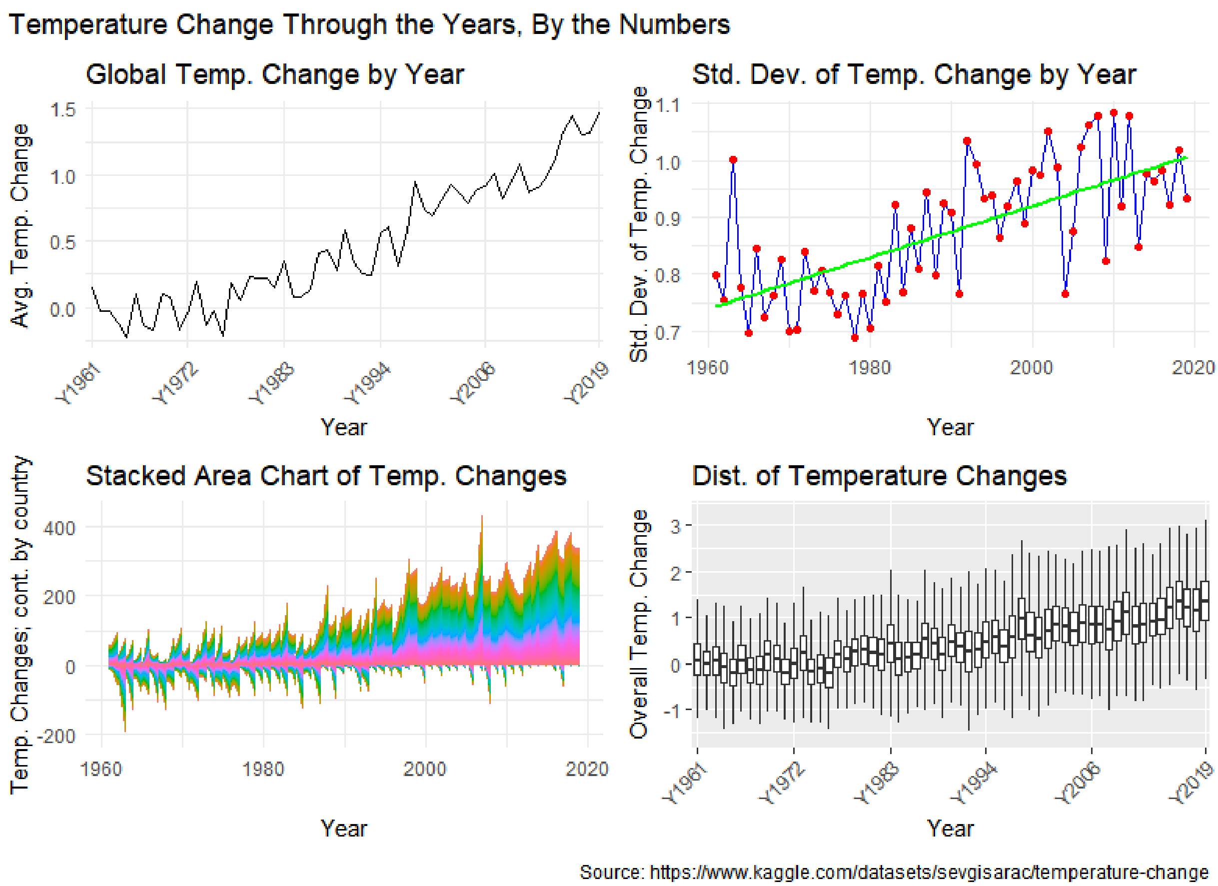
Foster Mosden
R Visualization Assignment



With the presidential election coming up, climate change has been a major topic in early primary debates, so I was interested in visualizing some data to explore whether a visual would make the reality of climate change more apparent to those still in denial of it. Many people believe that climate change is a constant that is not impacted by the actions of humans. By using data from 1960 to 2019, we have data for a significant period in technological advancement, and increased human intervention in the environment. I was hoping to prove not just that the climate is changing, but that the rate at which it has changed indicates that humans are a major cause. In this analysis, I explored a climate change dataset using R, including packages such as ggplot2, dplyr, tidyverse, and more, aiming to understand global temperature changes over the years. The dataset I used from Kaggle includes information on various countries' temperature changes for each year, as well as the standard deviation. I first loaded and examined the data to gain an understanding of its structure and completeness. I then went on to craft visualizations to find insights into different aspects of temperature changes.

The first visualization showcased the average temperature change across all areas for each year. Data was reshaped for clarity, and a line plot highlighted specific years to illustrate temperature change trends effectively. The second visualization focused on the standard deviation of temperature changes over time, revealing patterns and fluctuations through a line chart with a linear regression overlay as a point of reference for trends. The third visualization employed a stacked area chart, allowing for the comparison of temperature change patterns among different countries, to get a better understanding of how different countries might be contributing to the data. Lastly, the distribution of temperature changes was explored using box plots, omitting outliers to provide a clear overview. The visualizations were combined into a patchwork plot, to provide myself and any other viewers of my plots with a holistic view of temperature changes over time.

Again, I tried to indicate the impact of humans in my visualizations, but ultimately I don't think that my efforts did as good of a job as I would have hoped. For further analysis, it would be valuable to explore the factors contributing to extreme temperature changes in specific years or regions. Additionally, investigating the relationship between temperature changes and other environmental variables, such as greenhouse gas emissions or deforestation rates, could provide valuable insights into the underlying causes of temperature fluctuations. Time series modeling techniques could also be applied to forecast future temperature changes based on historical data patterns, contributing to climate change prediction efforts. Ultimately, I wish I had chosen a dataset with more cohesive information regarding climate change as opposed to the basic stats provided by the one I went with. I believe that putting more work into finding the right dataset, and increasing my EDA, I could have found better data, with which I could do better analysis, so that I could tell a better story. This was certainly a learning experience, and I gained a lot of experience that I will be applying to my next visualizations -- specifically in the storytelling department.

You will find the R code used in this assignment below

```r
# Setup ---------------------------------------------------------------------
# Read the CSV file with specified encoding
data <- read.csv("Environment_Temperature_change_E_All_Data_NOFLAG.csv",
fileEncoding = "Windows-1252")

library(ggplot2)
library(reshape2)
library(tidyverse)
library(dplyr)
library(summarytools)
library(patchwork)


# EDA -----------------------------------------------------------------------

# Print the column names
print(names(data))

# Print the structure of the dataframe
str(data)

# Summary statistics
summary(data)

# Check for missing values
print("Missing Values:")
print(colSums(is.na(data)))

# Count unique values in 'Area' column
print(paste("Number of unique areas:", length(unique(data$Area))))

# Count unique values in 'Element' column
print(paste("Unique elements:", unique(data$Element)))

# Correlation matrix (for numeric columns)
print("Correlation Matrix:")
print(cor(select_if(data, is.numeric), use = "complete.obs"))


# Global Temperature Changes Over the Years ------------------------------

# Reshape the data using gather() from tidyr
# The 'gather()' function is used to convert the data from wide format to
```

```
long format, making it easier to analyze and visualize.
gathered_data <- data %>%
  gather(key = "variable", value = "value",
      -Area.Code, -Area, -Element.Code, -Element, -Unit, -Months.Code,
-Months)

# Filter the data for rows with Element "Temperature change"
temperature_data <- gathered_data[gathered_data$Element == "Temperature
change", ]

# Calculate the average temperature change across all areas for each year
average_temperature_change <- aggregate(value ~ variable, data =
temperature_data, FUN = mean)

# Create a line plot with modified x-axis labels
glob_tmp_chng <- ggplot(average_temperature_change, aes(x = variable, y =
value, group = 1)) +
  geom_line() +  # Plotting the line
  labs(x = "Year", y = "Avg. Temp. Change", title = "Global Temp. Change by
Year") +  # Adding axis labels and plot title
  theme_minimal() +  # Using a minimal theme for the plot
  scale_x_discrete(
      breaks = c("Y1961", "Y1972", "Y1983", "Y1994", "Y2006", "Y2019"),  #
Specifying breaks for x-axis labels
      labels = c("Y1961", "Y1972", "Y1983", "Y1994", "Y2006", "Y2019")  #
Specifying labels for x-axis breaks
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotating
x-axis labels for better readability

plot.new()
glob_tmp_chng

# St. Dev. of Temperature Change ------------------------------------------

# Gather the data into long format
data_long <- data %>%
  gather(key = "Year", value = "Value", starts_with("Y"))

# Extract the year from the column names
data_long$Year <- gsub("Y", "", data_long$Year)

# Convert Year and Value columns to numeric
```

```
data_long$Year <- as.numeric(data_long$Year)
data_long$Value <- as.numeric(data_long$Value)

# Filter rows where Element is "Temperature change"
temperature_data <- data_long %>%
  filter(Element == "Temperature change")

# Calculate standard deviation for each year
std_data <- aggregate(Value ~ Year, data = temperature_data, FUN = sd)
names(std_data)[2] <- "std_dev"

# Plotting the data
st_dev_tmp_chng <- ggplot(std_data, aes(x = Year, y = std_dev)) +
  geom_line(color = "blue") +
  geom_point(color = "red") +
  geom_smooth(method = "lm", formula = 'y ~ x', se = FALSE, color =
"green") +
  labs(x = "Year", y = "Std. Dev. of Temp. Change") +
  ggtitle("Std. Dev. of Temp. Change by Year") +
  theme_minimal()

plot.new()
st_dev_tmp_chng

# Stacked Area Chart of Temp Change ---------------------------------------

# Gather the data into long format
data_long <- data %>%
  gather(key = "Year", value = "Value", starts_with("Y"))

# Extract the year from the column names
data_long$Year <- gsub("Y", "", data_long$Year)

# Convert Year and Value columns to numeric
data_long$Year <- as.numeric(data_long$Year)
data_long$Value <- as.numeric(data_long$Value)

# Filter the data for rows with Element "Temperature change"
data_long <- data_long[data_long$Element == "Temperature change", ]

# Create a stacked area chart without legend
stacked_tmp_chng <- ggplot(data = data_long, aes(x = Year, y = Value, fill
= Area)) +
```

```
  geom_area(show.legend = FALSE) +  # Set show.legend to FALSE to hide the
legend
  theme_minimal() +
  labs(x = "Year", y = "Temp. Changes; cont. by country", fill =
"Country/Territory") +
  ggtitle("Stacked Area Chart of Temp. Changes")

plot.new()
stacked_tmp_chng

# Distribution of Temperature Changes over the years ----------------------

# Assuming your dataframe is named "data"
# Extract columns for years from Y1961 to Y2019
years <- names(data)[which(names(data) %in% paste0("Y", 1961:2019))]

# Reshape the data from wide to long format using gather
gathered_data <- data %>%
  gather(key = "Year", value = "Value", !!years, na.rm = TRUE) %>%
  select(Area, Element, Year, Value)

# Filter the data for rows with Element "Temperature change"
gathered_data <- gathered_data[gathered_data$Element == "Temperature
change", ]

# Calculate upper and lower fences for each year
fences <- gathered_data %>%
  group_by(Year) %>%
  summarize(lower_fence = quantile(Value, 0.25) - 1.5 * IQR(Value),
            upper_fence = quantile(Value, 0.75) + 1.5 * IQR(Value))

# Create a box plot for each year without outliers
dist_tmp_chng <- ggplot(gathered_data, aes(x = Year, y = Value)) +
  geom_boxplot(outlier.shape = NA) +  # Exclude outliers from the plot+
  ylim(min(fences$lower_fence), max(fences$upper_fence)) +  # Set y-axis
limits based on fences
  labs(x = "Year", y = "Overall Temp. Change", title = "Dist. of
Temperature Changes") +
  scale_x_discrete(
      breaks = c("Y1961", "Y1972", "Y1983", "Y1994", "Y2006", "Y2019"),  #
Specifying breaks for x-axis labels
      labels = c("Y1961", "Y1972", "Y1983", "Y1994", "Y2006", "Y2019")  #
Specifying labels for x-axis breaks
```

```
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotating
x-axis labels for better readability

plot.new()
dist_tmp_chng

# Patchwork Plot ----------------------------------------------------------

# Combine all plots and add source text at the bottom
combined_plot <- glob_tmp_chng + st_dev_tmp_chng + stacked_tmp_chng +
dist_tmp_chng +
  plot_layout(guides = "collect") +
  plot_annotation(
      title = "Temperature Change Through the Years, By the Numbers",
      tag_prefix = "Plot ",
      caption = "Source:
https://www.kaggle.com/datasets/sevgisarac/temperature-change"
  )

# Print or save the updated combined plot
print(combined_plot)
```