Global Climate Change Contributors

Ted Liu | Elaine Zhang | Nguyen Ha | Baisakhi Sarkar | Diane Chiang

Contents

Abstract

In this analysis, we investigated how CO_2 emissions have changed in recent years and explored how forestry, socioeconomic, energy production, and urbanization metrics are associated with CO_2 emissions. We performed t-tests of the means comparing mean CO_2 emissions between 1990-2004 and 2005-2020 and found that they are not statistically different at the global level but are statistically different at individual and smaller group levels. We fitted linear regression models to determine the association with CO_2 emissions for the different metrics and found that each category of metrics had at least one indicator associated with CO_2 emissions. We conclude that forestry, socioeconomic, energy production, and urbanization have metrics that are both positively and negatively associated with CO_2 emissions. These results suggest that the factors for CO_2 emission contribution of a nation is complex, and mitigation of the climate change crisis warrants investigation into a multitude of factors.

Introduction

In 2023, Earth witnessed a record-breaking year in terms of temperature, marking the hottest year since the commencement of modern global temperature tracking in 1850 (Climate.gov, 2024a). This unprecedented surge in global temperatures not only impacts us but also affects other inhabitants of Planet Earth, as increased temperatures have led to increases in the acidity of our oceans (US EPA, 2016a) and an increase in certain diseases (US EPA, 2016b). The cause of the ongoing climate change crisis is indisputably human activities - it is estimated that human-related actions have led to a 0.8°C to 1.3°C increase in global temperatures between 1850 and 2019 (Climate.gov, 2024b).

Our analysis aims to examine recent data on greenhouse gas emissions, specifically carbon dioxide emissions. Using data from the World Bank and the International Monetary Fund (IMF), we examined whether the average CO_2 emissions globally have changed between the time periods of 1990-2005 and 2005-2020, and if there are any socioeconomic and environmental metrics associated with CO_2 emissions from 1990-2020. We sought to answer:

1. Have average CO_2 emissions increased since the late 20th century and early 21st century?
2. Are there socioeconomic indicators associated with CO_2 emissions?
3. Are CO_2 emissions associated with any forestry or natural disaster metrics?
4. Are there specific energy consumption or production methods correlated with CO_2 emissions?
5. Are there any relationships between urbanization and CO_2 emissions?

Methods

Data Description

Our research relies on the Environment, Social, and Governance Data provided by The World Bank, encompassing 71 indicators. These indicators include measures such as access to clean fuel, access to electricity, and fertility rate from 1960 to 2023. We primarily focused on CO_2 emissions (metric tons per capita), with the available data covering 1990 to 2020 for 193 countries. The World Bank acquired the data from official sources and made some adjustments to standardize indicator names.
The forestry metrics are from the International Monetary Fund's Climate Change Indicators Dashboard's Forest and Carbon dataset, sourced from FAOSTAT. The data includes forest area, land area, forest share, and carbon stock index for 225 countries from 1992 to 2020. The climate-related disasters frequency data is also from the International Monetary Fund, specifically the Climate Change Dashboard. The dataset encompasses various natural disasters, including landslides, droughts, extreme temperatures, and wildfires, across 215 countries and regions spanning from 1980 to 2022. To ensure consistency in our comparison, we focused on the frequency records from 1990 to 2020, aligning with the timeframe of our CO_2 emissions data. Moreover, we addressed missing data by zero-filling, essentially treating the absence of data as an indication that no such natural disaster occurred during the specific year.

Statistical Methods

Assessing Differences in Mean CO_2 Emission (Diane)

To understand changes in CO_2 emissions over time, we divided the years into two periods: 1990 to 2004, denoting the first half, and 2005 to 2020, representing the second half. We are interested in three different perspectives of comparing the mean CO_2 emissions, namely:

1. Global difference in mean CO_2 emissions between the two time periods.

2. Difference in mean CO_2 emissions between the two time periods for the 30 countries with the highest mean CO_2 emissions based on the data in the second time period.
3. Difference in mean CO_2 emissions between the two time periods for the 30 countries with the lowest mean CO_2 emissions based on the data in the second time period.
4. Individual countries of interest: Qatar, the United States, Burundi, and Ethiopia.

Given that we are examining the same set of countries for both periods, we employed a paired t-test on the mean without assuming equal variance to evaluate differences in mean CO_2 emissions globally and used a one-sample t-test for testing mean difference within specific groups. Additionally, we utilized paired t-tests to compare the mean differences in CO_2 emissions for individual countries of interest. It's worth noting that, to ensure equal time spans of 15 years each, we excluded the 2020 data for the paired t-tests. Furthermore, we assume that the matched pairs, representing the mean CO2 emissions (measured in metric tons per capita) within a country in the two time periods, are drawn from a population that follows a normal distribution.

In essence, for the primary test, we computed the mean emissions of each country from 1990 to 2005 and from 2005 to 2020, conducting paired t test on the mean difference to understand global changes in mean CO_2 emissions. Subsequently, we replicated this process for the top 30 countries with the highest emissions in the latter period and for the bottom 30 countries, employing one sample t-test on the mean difference to compare the means between the two timeframes. This approach aimed to differentiate the CO_2 emission trends among countries in various emission categories. Additionally, we singled out specific countries of interest, such as Qatar, the United States of America, Burundi, and Ethiopia, to comprehend the changes in CO_2 emissions at the country level.

Socioeconomic Factors (Ted)

Indicator Selection

Greenhouse gas emissions are often directly associated with industry and energy production such as coal / oil burning and vehicles. However, there could be other indicators related to the economy or society of a country that may impact their CO_2 emissions.

For our analysis on socioeconomic predictors, we chose variables that we believed were good measures of economic development and social progressiveness. For our indicators we chose:

- Prevalence of overweight (% of adults)
- GDP Growth (annual %)
- Population density (people per sq. km of land)
- Proportion of seats held by women in national parliaments
- School enrollment, primary (% gross)
- Voice and Accountability (an estimated score which describes the people / populations participation in government)
- Industry (including construction), value added (% of GDP).

Assumption Check and Model Selection

Prior to selecting a model, we verified if our data was fit for linear regression by verifying that our data followed the assumptions of linear regression:

1. Linear relationship between predictor and response
2. Predictors are independent.
3. Residual errors have a mean value of 0.
4. Residual errors have constant variance.

5. Residual errors are independent from each other.

We applied a log transformation to our response variable of CO_2 emissions to address issues of heteroskedasticity and standardized the variable for education spending to scale the values. Year and Nation may introduce variability in our model as CO_2 emissions can differ depending on the nation and year of the data due to unmeasured or unobserved differences. To control for this, we included Year and Nation into our linear regression model as predictors.
Since our regression model has log(CO_2) as its response variable, we assessed the practical significance of our findings by examining the percent change in CO_2 emissions per unit change in a predictor. This was done by applying the following:

$$\%\, change = \ (e^{\beta} - 1) * 10$$

Forestry-related Metrics (Elaine)

Indicator Selection

In the context of climate change, forests play an important role in both the mitigation and aggravation of greenhouse gas emissions. While forests absorb carbon dioxide, deforestation releases carbon dioxide (IUCN, 2021). As such, we decided to explore the association between forestry and carbon dioxide emissions in recent years. To make comparison standardized, we chose to use Forest Area (% of land area) and Agriculture, Forestry, and Fishing Value Added (% of GDP) as our forestry-related metrics and joined these with the CO_2 emissions data (measured in metric tons per capita). Missing values Agriculture, Forestry, and Fishing Value Added data were filled in using the country's average over the available data. Countries without any values were removed from analysis.

Data Aggregation

After preliminary exploration of the data, we decided to aggregate values over the period rather than keeping year as a variable. Logically, a country's forest share would not be able to fluctuate very much within a thirty-year period, as it is limited by a country's environmental conditions, such as land area and climate type. To check if any unexpected changes occurred from 1992-2020, we plotted the mean forest share at both the global and regional levels. We did not find any concerning trends blocking us from aggregating the values.

Assumption Check and Method Selection

To explore the relationship between forestry and carbon dioxide emissions, we decided to fit a linear regression model between the mean forest share, mean agriculture, forestry, and fishing value added to GDP, and mean CO_2 emissions. We checked the following assumptions of linear regression: linear relationship between predictor and response, homoscedasticity, and independence of predictors. In our preliminary exploratory analysis, we found that the relationship between CO_2 emissions and each of the forestry metrics did not appear to be linear. As such, we tested transformations of the CO_2 emissions variable. Through comparing the residual plots of both the inverse transformation model and log transformation model, we determined that the log transformation model was a better fit. We also checked the homoscedasticity assumption by plotting fitted values against residuals and residuals against predictors and found that the homoscedasticity assumption was violated. To remedy this, we decided to use jackknife to estimate the standard errors of the model coefficients. We then used the jackknife estimates of standard error to create 95% confidence intervals and calculate p-values.

Climate-related Disasters Frequency (Diane)

To assess the potential correlation between the frequency of natural disasters and CO_2 emissions (measured in metric tons per capita), we create a series of combined histograms depicting the counts of

natural disasters and line graphs illustrating CO_2 emissions for each selected country. Specifically, we focus on the United States, Canada, Congo, Dem. Rep., and the Central African Republic. to visually examine the trends. Since CO_2 emissions are more closely related to extreme temperature and wildfire, we focused on those two climate-related disasters for further understanding.

Energy Production and Consumption (Nguyen)

Indicator Selection

Energy production methods and sources vary in their impact on carbon dioxide (CO_2) emissions, shaping our environment and climate. Also, energy consumption directly impacts CO_2 emissions as it dictates the amount of energy produced to meet demand. Understanding how energy production and consumption influence overall emissions is crucial for mitigating climate change.

To analyze the relationship between different methods of energy production and CO_2 emissions, we choose indicators measuring the percentage of sources in production:

- Electricity production from coal sources (% of total)
- Electricity production from hydroelectric sources (% of total)
- Electricity production from natural gas sources (% of total)
- Electricity production from nuclear sources (% of total)
- Electricity production from oil sources (% of total)
- Electricity production from renewable sources, excluding hydroelectric (% of total)

To analyze the relationship between energy consumption and CO_2 emissions, we utilize these below indicators:

- Energy use (kg of oil equivalent per capita)
- Electric power consumption (kWh per capita)
- Fossil fuel energy consumption (% of total)
- Combustible renewables and waste (% of total energy)
- Alternative and nuclear energy (% of total energy use)
- Electric power transmission and distribution losses (% of output)
- CO_2 emissions (metric tons per capita)

Transforming Data

To analyze the relationship energy production and consumption with CO_2 emissions using data from the World Bank regardless of countries and time, we first transform the data. This involves transposing the data, with indicators serving as column names. Each row is uniquely identified by the country and year, facilitating comprehensive analysis of trends across nations.

Method Selection and Assumption Check

We employ linear regression to analyze relationships between energy production and consumption with CO_2 emissions. For valid confidence intervals of coefficients, we then verify if residual errors have constant variance.

Urbanization Metrics (Baisakhi)

Indicator Selection

In the context of climate change, urbanization plays a crucial role in both contributing to and mitigating greenhouse gas emissions. Urban areas are significant sources of carbon dioxide emissions due to high energy consumption, transportation activities, and industrial processes. Conversely, urbanization can also

lead to advancements in infrastructure and technology that promote energy efficiency and renewable energy adoption, thereby reducing carbon emissions.
To explore the association between urbanization and carbon dioxide emissions, we have chosen to use the indicator "Urban population (% of total population)." This indicator represents the proportion of a country's population living in urban areas and serves as a proxy for urbanization level. We will analyze how changes in urban population percentage relate to carbon dioxide emissions over recent years (CO2 emissions data (measured in metric tons per capita))

Data Preprocessing/Transformation

After exploring the datasets, we opted to aggregate values over the study period instead of analyzing individual years. Urbanization trends typically evolve gradually over time, shaped by economic development, population growth, and urban planning policies. By aggregating the data, our aim is to capture the overall impact of urbanization on carbon dioxide emissions while minimizing the influence of short-term fluctuations. We selected data from 1990 to 2020, as earlier data lacked consistency.
To begin the data cleaning and preprocessing step, we focused solely on the indicators relevant to our analysis: Urban Population (% of total population) and CO2 Emissions (metric tons per capita). We discarded all other indicators. Subsequently, we filtered out any null or NaN values and grouped the data by country name.
For our preliminary analysis, we visualized the relationship between Urban Population (% of total population) and CO2 Emissions using a scatterplot. This initial step allowed us to assess the assumptions of our analysis and check for any potential issues before proceeding with fitting a linear regression model.

Assumption Check and Method Selection

Our initial hypothesis posits that increased urbanization may lead to higher carbon dioxide emissions due to greater energy demand and transportation requirements in urban areas. To test this hypothesis, we plan to fit a linear regression model using urban population percentage as the predictor variable and carbon dioxide emissions as the response variable.
Before fitting the model, we checked the assumptions of linear regression, including the presence of a linear relationship between urban population percentage and carbon dioxide emissions, homoscedasticity of residuals, and independence of predictors. Our model was violating the homoscedasticity of residuals assumption – we will discuss more on this in the results section). Based on the analysis, we employed appropriate statistical techniques, such as robust standard errors or weighted least squares, to account for any violations of model assumptions and ensure the reliability of our findings.
In summary, our study aims to investigate the relationship between urbanization, as measured by urban population percentage of total population, and carbon dioxide emissions, contributing to a better understanding of the role of urban development in climate change mitigation and adaptation efforts.

Results

Assessing Differences in Mean CO_2 Emission (Diane)

The paired t-test, conducted without assuming equal variance to compare the global mean CO_2 emissions, resulted in a p-value of approximately 0.5689. This value exceeds the significance level of 0.05, suggesting that there is no sufficient evidence to conclude that the two global means are significantly different. The estimated global mean difference is approximately 0.0614, and we are 95% confident that the true global mean difference between the two time periods lies between -0.1507 and 0.2734. Our findings align with the visual representation below: the left figure illustrates the distributions of mean CO_2 emissions for the two time periods, and the right figure depicts the distribution of mean differences.

Notably, the two mean distributions of the two time periods exhibit similar patterns, and the mean difference distribution is centered around 0.

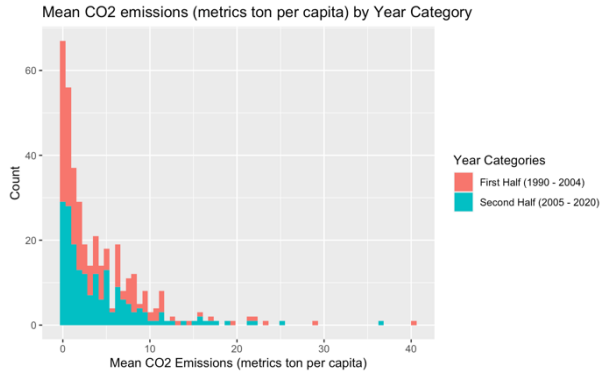


Figure 1. Global Mean CO2 Emissions Distributions.

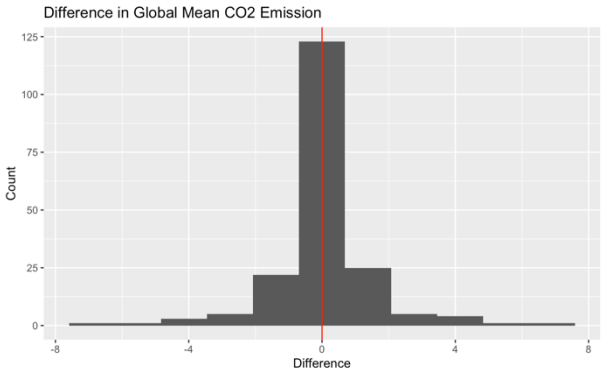


Figure 2. Difference in Global Mean CO2 Emissions

Additionally, our investigation continues with the mean CO_2 emissions distributions of the top 30 countries with the highest mean CO_2 emissions, based on data spanning from 2005 to 2020. Using a one-sample t-test to analyze the mean difference, the resulting p-value is 0.4544, surpassing the 0.05 threshold. Consequently, we do not have sufficient evidence to conclude that there is a difference in mean CO_2 emissions between the two time periods within the top 30 countries with the highest emissions per capita in metric tons. The visual representation of the distributions supports this conclusion, with the left figure displaying the mean distributions and the right figure illustrating the differences. The point estimate for the mean difference is approximately -0.4406, and with 95% confidence, we estimate that the true mean difference in CO_2 emissions between the two time periods lies within the range of -1.6289 to 0.7478.

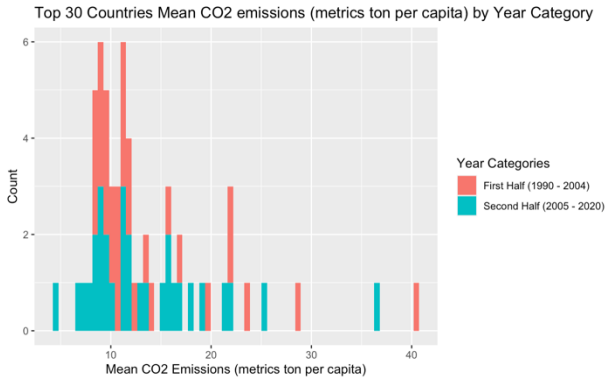


Figure 3. Mean CO2 Emissions Distributions For the 30 Countries with the most emission.

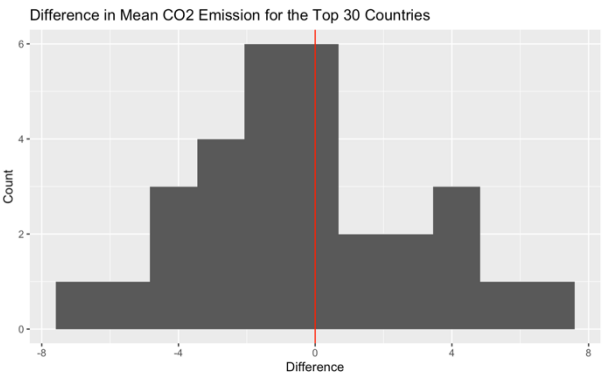


Figure 4. Difference in Mean CO2 Emissions for the Top 30 Countries.

We also investigated 30 countries with the least CO_2 emissions based on the second time period, similarly applying the one-sample t-test to assess the mean difference. The resulting p-value of 0.0018, being below the 0.05 significance level, leads us to conclude that a significant difference exists in mean CO_2 emissions within the 30 countries with the least CO_2 emissions between the two time periods. Specifically, the mean CO_2 emissions are higher for the period 2005 – 2020 compared to 1990 – 2004. The distribution of mean CO_2 emissions (bottom left) shows that the second half of the year period has higher mean CO_2 emissions compared to the first half, and the mean difference is also heavily skewed to the right (bottom right). The point estimate of the mean difference is 0.1227 and we are 95% confident that the true mean difference in CO_2 emissions between the two time periods is between 0.0497 and 0.1957.

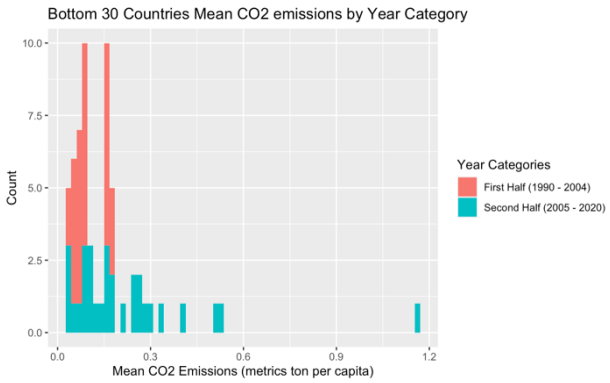


Figure 5. Mean CO2 Emissions Distributions For the 30 Countries with the least emission.

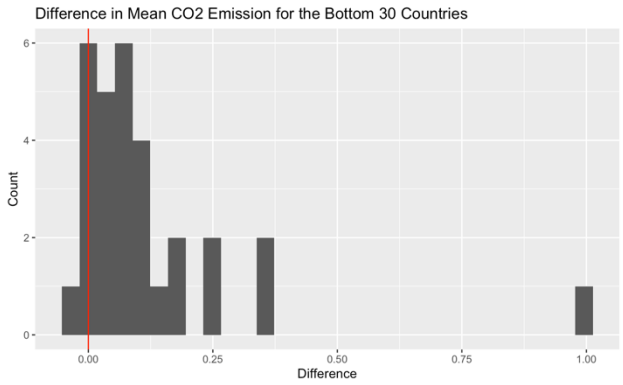


Figure 6. Difference in Mean CO2 Emissions for the Bottom 30 Countries.

Aside from exploring mean CO_2 emissions among different groups, we focus on four specific countries: Qatar, the United States of America, Burundi, and Ethiopia. We are interested in Qatar due to it having the highest mean CO_2 emissions in the second time period. The United States is of particular interest to us given our current location, while Burundi and Ethiopia were chosen as they represent countries with some of the lowest mean CO_2 emissions.

For Qatar (fig 9) in the 1990 – 2004 period, CO_2 emissions were most prevalent at around 45 metric tons per capita, shifting to approximately 35 metric tons per capita in the 2005 – 2020 period. The resulting paired t-test, with a p-value of 0.183 (greater than 0.05), indicates that we fail to reject the hypothesis that the true mean CO_2 emissions difference is 0. The point estimate stands at -3.53 metric tons per capita, and we are 95% confident that the true mean difference lies between -8.96 and 1.87 metric tons per capita.

Contrastingly, the United States (fig 10) exhibits a notable difference in means. In the first period, the mean centers around 20 metric tons per capita, whereas in the second period, it drops to around 16 metric tons per capita, with a broader spread. The associated paired t-test yields a p-value of 3.98e-05 (less than 0.05), indicating a significant difference. Specifically, the mean CO_2 emissions from 2005 – 2020 are lower than those from 1990 – 2004. The estimated mean difference is -2.9758 metric tons per capita, and we are 95% confident that the true difference lies between -4.061 and -1.89 metric tons per capita.

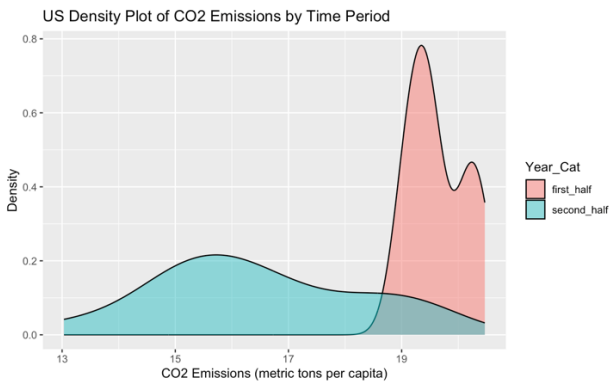


Figure 7. Qatar Mean CO2 Emissions Distributions for the Two Time Periods.

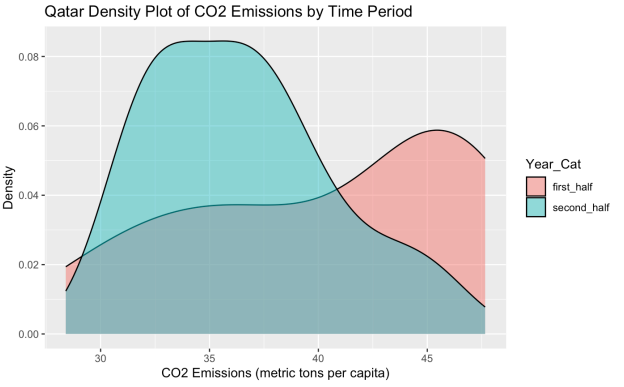


Figure 8. US Mean CO2 Emissions Distributions for the Two Time Periods.

Applying a similar approach to evaluate CO_2 emissions in Burundi and Ethiopia, we conducted a paired t-test for Burundi, resulting in a p-value of approximately 0.879. The estimated mean difference stands at about 0.00065, with a 95% confidence interval spanning from -0.0084 to 0.0097. Given the high p-value,

we lack sufficient evidence to conclude that the true mean difference in CO2 emissions for Burundi is non-zero.

Conversely, the paired t-test for Ethiopia yields a markedly low p-value of 9.478e-07. The point estimate is around 0.053, and with a 95% confidence interval ranging from 0.0393 to 0.0669 metric tons per capita, we confidently conclude that the mean difference is not zero.

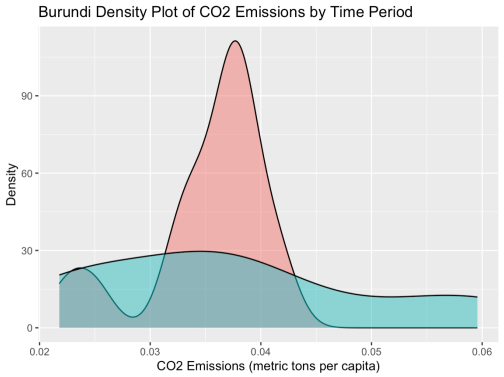


Figure 9. Burundi Mean CO2 Emissions Distributions for the Two Time Periods.

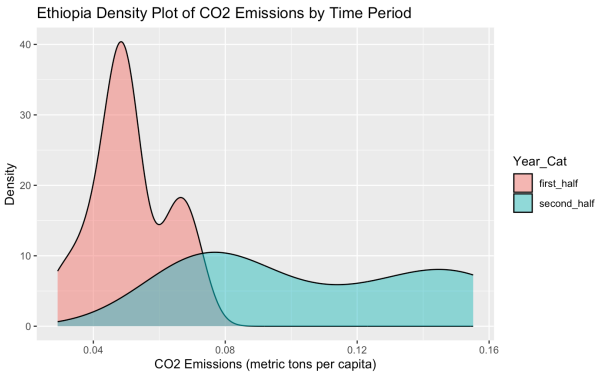


Figure 10. Ethiopia Mean CO2 Emissions Distributions for the Two Time Periods.

Socioeconomic Associations with CO_2 Emissions (Ted)

We fitted an initial linear regression, model_1, to our variables of interest to the CO_2 emissions.

$$CO_2 = \beta_0 + \beta_1(overweight) + \beta_2(gdp.growth) + \beta_3(pop.density) + \beta_4(prop.women) + \beta_5(citizen.voice) + \beta_6(scaled.edu.spend) + \beta_7(industry) + \beta_8(urban.pop)$$

We then evaluated the model to observe if the assumptions for linear regression hold for this initial model.

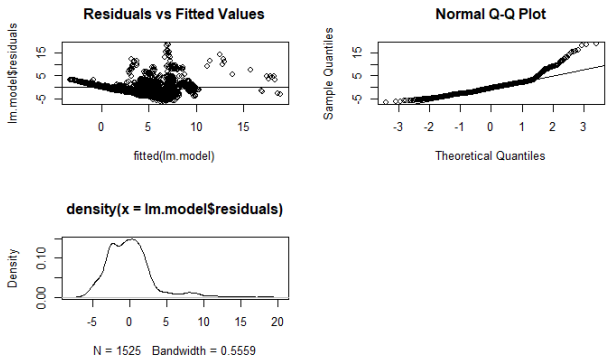


Figure 11. Model_1 Assumption Assessment

In Figure 11, we can observe that some assumptions for linear regression are violated in this initial model. We can observe that there are some signs of heteroscedasticity in our residuals vs fitted values plot and signs that the normality assumption may not hold in our Q-Q plot.

To address some issues of heteroscedasticity, we log transformed the response variable of CO_2 emissions and then fitted another model.

$$\log(CO_2) = \beta_0 + \beta_1(overweight) + \beta_2(gdp.growth) + \beta_3(pop.density) + \beta_4(prop.women) + \beta_5(citizen.voice) + \beta_6(scaled.edu.spend) + \beta_7(industry) + \beta_8(urban.pop)$$

We then evaluated this new model, model_2 to observe if the assumptions for linear regression are being violated or not.

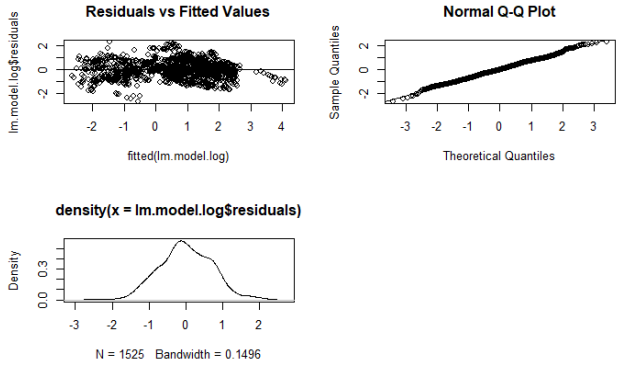


Figure 12. Model_2 Assumption Assessment

In Figure 12, we can observe that log transforming the response variable has addressed some of the issues with heteroscedasticity and normality. While this model seems to conform to the assumption of linear regression, a fundamental issue needs to be addressed – which is the fact that our data is longitudinal, nation-level data. Year and Country may have a relationship with CO_2 emissions. For example, there might be years and/or countries that may have different or variable CO_2 emissions for unobserved or unrecorded reasons that contribute to the variability in our data. Because we want to examine the contributions of our indicators of interest and not about timepoints and nations, we fitted another model, model_3 with the inclusion of Year and Country to account for the variability that Year and Country may have on CO_2 emissions.

$$\log(CO_2) = \beta_0 + \beta_1(overweight) + \beta_2(gdp.growth) + \beta_3(pop.density) + \beta_4(prop.women) + \beta_5(citizen.voice) + \beta_6(scaled.edu.spend) + \beta_7(industry) + \beta_8(urban.pop) + \beta_9(Year) + \beta_{10}I(Country)$$

Examining our final linear regression model, we find that, with log transformation of the response variable (CO_2 emissions) and inclusion of Year and Nation into our model provides the "best" model in terms of conforming with the assumptions for linear regression (Figure 11).

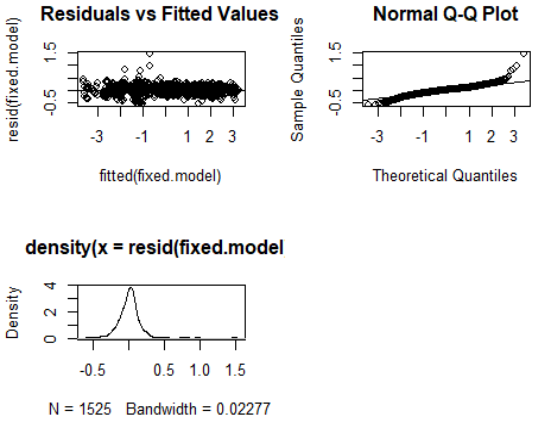


Figure 13. Model_3 Assumption Assessment

Table 1: Linear Regression Model Output (excluding Nation from output)

Coefficient	*Estimate*	*Lower*	*Upper*	*P_Value*
(Intercept)	37.1570685	22.2196594	52.0944776	0.0000012
Year	-0.0198028	-0.0273022	-0.0123033	0.0000003
overweight	0.0467205	0.0307469	0.0626941	0.0000000
gdp.growth	0.0025781	0.0001392	0.0050170	0.0382990
pop.density	0.0006291	0.0000673	0.0011910	0.0282135
prop.women	0.0019987	0.0000359	0.0039615	0.0459575
citizen.voice	0.0603953	0.0091009	0.1116898	0.0210497
scale_edu.spend	-0.0242037	-0.0474067	-0.0010006	0.0409176
industry	0.0131072	0.0100951	0.0161192	0.0000000

In Table 1, we can observe all the coefficients for each predictor – excluding the nation variables. We observed that, at a 0.05 significance level, all our predictors were statistically significant. With coefficients representing change in log(CO_2) emissions per unit change, we examined the percent change in CO_2 emissions per unit change for each predictor to assess practical significance.

Table 2: Percent Change

Coefficient	*Percent Change*	*Lower_CI*	*Upper_CI*	*P_Value*
Year	-1.9607987	-2.6932897	-1.2227936	0.0000003
overweight	4.7829063	3.1224430	6.4701062	0.0000000
gdp.growth	0.2581412	0.0139184	0.5029603	0.0382990
pop.density	0.0629337	0.0067290	0.1191700	0.0282135
prop.women	0.2000688	0.0035923	0.3969313	0.0459575
citizen.voice	6.2256401	0.9142391	11.8165951	0.0210497
scale_edu.spend	-2.3913110	-4.6300544	-0.1000147	0.0409176
industry	1.3193433	1.0146216	1.6249843	0.0000000
urban.pop	1.3372712	0.6268832	2.0526742	0.0002203

We then examined the practical significance of our findings by examining the percent change of CO_2 metric tons per capita for a unit change in a coefficient. Table 2 displays the percent change for each coefficient. From these results we can argue that the prevalence of overweight adults, voice and accountability metric, industry, and urban population are perhaps the more practically significant coefficients that are associated with CO_2 emissions and that education spending is negatively associated with CO_2 emissions.

Forestry-related Metrics (Elaine)

We fitted a multiple linear regression with interaction model, with mean forest share (% of land area), agriculture, forestry, and fishing value added (% of GDP) as the explanatory variables and the log of CO_2 emissions (metric tons per capita) as the response variable.

$$\log(CO_2) = \beta_0 + \beta_1(mean\ forest\ share) + \beta_2(mean\ agriculture,\ forestry,\ fishing\ value\ added) + \beta_3(interaction\ btwn\ mean\ forest\ share\ and\ mean\ forestry\ value\ add)$$

To check the constant variance assumptions, normality assumptions, and to assess the goodness-of-fit, we plotted the residual charts below as well as a quantile-quantile chart not pictured.

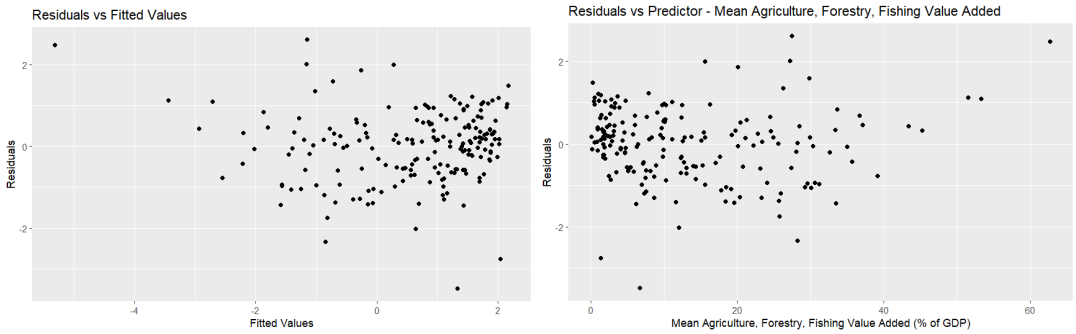


Figure 14: Forestry multiple linear regression model assumption assessment.

To check if the two forestry-related metrics had any signs of collinearity, we did a quick calculation of the correlation between the two and plotted the variables on a scatterplot. With a scatterplot that showed no apparent relationship between the two and a correlation value of 0.018, we felt it was safe to assume that the two predictors are independent and proceed with fitting the linear model.

Table 3: Forestry Multiple Linear Regression Model

Coefficient	*Estimate*	*Lower*	*Upper*	*P-Value*	*Percent Change*
(Intercept)	2.20012187	1.784031	2.616213	4.0842e-20	
Mean Forest Share	-0.00728047	-0.0160803	0.0015194	0.106605	-0.7254038
Mean Agriculture, Forestry, Fishing Value Added	-0.12429299	-0.1528105	-0.095775	4.8943e-15	-11.68789
Interaction Between Mean Forest Share and Mean Agriculture, Forestry, Fishing Value Added	0.000521439	-3.71896e-05	1.0800e-03	0.06894064	0.05215754

From the table above, we can see that of the three coefficients, only one, the coefficient for mean agriculture, forestry, fishing value added, is statistically significant at a significance level of 0.05. By taking the exponent of the coefficient estimates and calculating the percent change, we can interpret the coefficient as follows. A one-percent increase in mean agriculture, forestry, fishing value added to GDP is associated with a 11.69% decrease in mean metric tons of CO_2 emissions per capita, adjusting for other factors. Surprisingly, the forestry value of a country's GDP has a negative relationship with carbon dioxide emissions. This may be because countries with a larger dependency on the agriculture, forestry, and fishing industries generate less carbon dioxide emissions in general. The model shows that there is not enough evidence to say there is a relationship between forest share and carbon dioxide, which makes sense in context. Countries that have larger forest shares do not necessarily partake in more forest-related activities (deforestation, forestry), and therefore the carbon dioxide emissions do not have much of a relationship with forest share.

Climate-related Disaster Frequency (Diane)

To explore the potential correlation between natural disaster frequency and CO_2 emissions, we visually represent trends through histograms. As mentioned earlier, the climate-related disaster frequency data

encompasses events like droughts, floods, extreme temperatures, landslides, and storms. In line with the approach taken in the "Assessing Differences in Mean CO_2 Emission" section, we narrow our focus to four specific countries: the United States of America, Canada, Congo, Dem. Rep., and the Central African Republic. The selection of different countries is a result of variations in the availability of climate-related disaster data across nations with CO_2 emissions data.

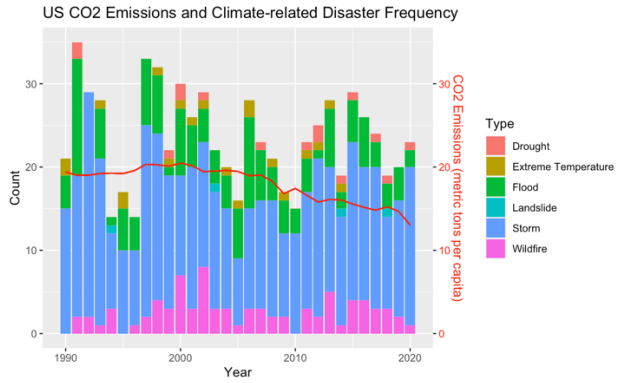


Figure 15. US CO_2 Emissions and Climate-Related Disaster Frequency

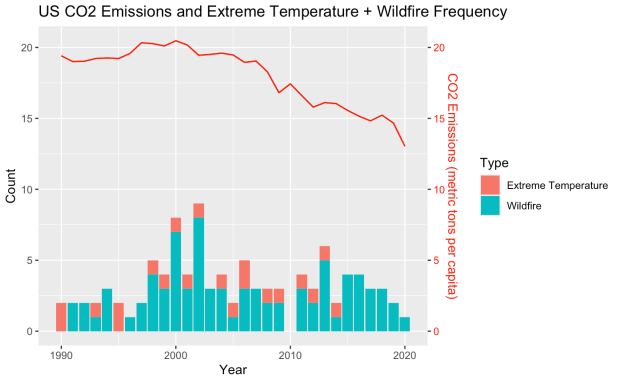


Figure 16. US CO_2 Emissions and Extreme Temperature + Wildfire Frequency

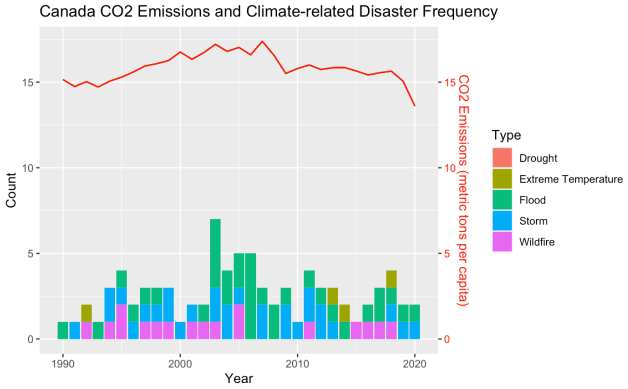


Figure 17. Canada CO_2 Emissions and Climate-Related Disaster Frequency

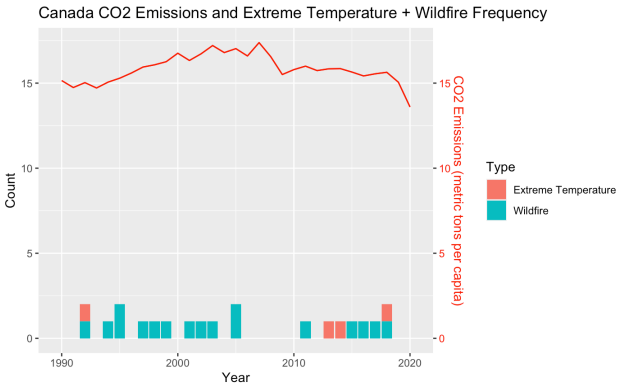


Figure 18. Canada CO_2 Emissions and Extreme Temperature + Wildfire Frequency

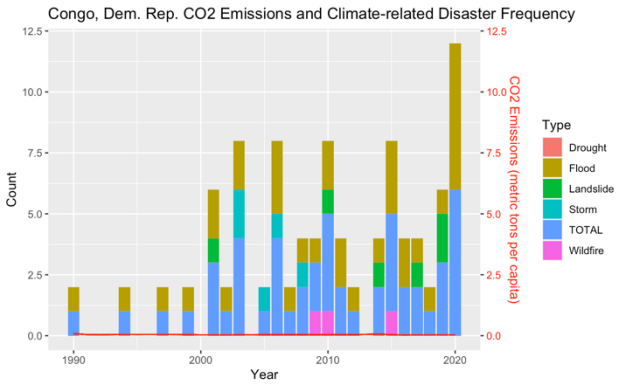


Figure 19. Congo Dem. Rep. CO_2 Emissions and Climate-Related Disaster Frequency

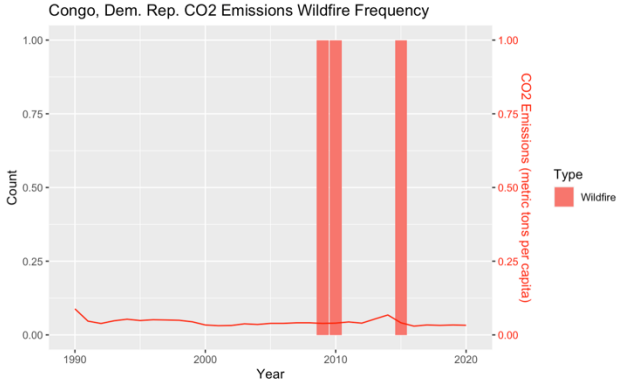


Figure 20. Congo Dem. Rep. CO_2 Emissions and Wildfire Frequency

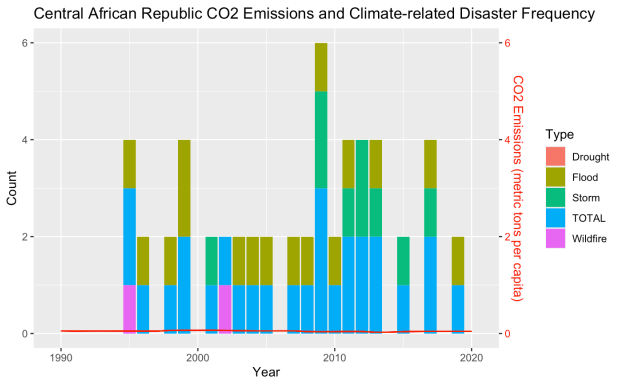


Figure 21. Central African Republic CO_2 Emissions and Climate-Related Disaster Frequency

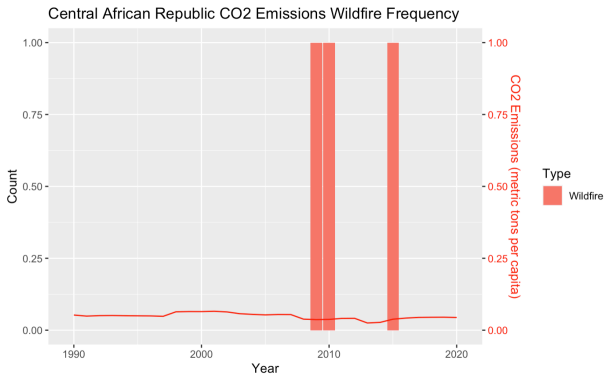


Figure 22. Central African Republic CO_2 Emissions and Wildfire Frequency

Based on the four countries shown here, CO_2 emissions and climate-related disasters frequency are not associated. Specifically, the pattern of related natural disaster is independent of the CO_2 emissions pattern and vice versa.

Energy Production and Consumption (Nguyen)

We fitted 2 linear regressions in this section. One uses energy production indicators as predictor variables and CO_2 emissions (metric tons per capita) as response variable, called *prod_mod*. The other uses energy consumption indicators as predictor variables and CO_2 emissions (metric tons per capita) as response variable, called *use_mod*.

In Table 4a, we can observe the coefficients for all predictors from energy production indicators of *prod_mod* model. In Table 4b, we can observe the coefficients for all predictors from energy consumption indicators of *use_mod* model. From Table 4a and 4b, we can see that, at the significance level of 0.05, all variables are significant, except for production from oil sources. Furthermore, the coefficient of electricity production from oil sources is negative. This contradicts a presumption we have that electricity production from oil sources would have high contribute to CO_2 emissions. It is also unusual for electricity production from renewable sources to have a positive coefficient. We suspect this is because 49% of rows in "Electricity production from oil sources (% of total)" has zero values (after imputation) and 63% of rows in "Electricity production from renewable sources, excluding hydroelectric (% of total)" has zero values.

Table 4a: Linear Regression Model Output of prod_mod

Coefficient	*Estimate*	*Standard Error*	*P-Value*
(Intercept)	1.87460	0.06613	< 2e-16
Electricity production from coal sources (% of total)	0.0381160	0.0021136	< 2e-16
Electricity production from hydroelectric sources (% of total)	-0.0181210	0.0017275	< 2e-16
Electricity production from natural gas sources (% of total)	0.0889180	0.0019592	< 2e-16
Electricity production from nuclear sources (% of total)	0.0965880	0.0044676	< 2e-16
Electricity production from oil sources (% of total)	-0.0000912	0.0022065	0.9670000
Electricity production from renewable sources, excluding hydroelectric (% of total)	0.0627350	0.0110858	1.5700E-08

Table 4b: Linear Regression Model Output of use_mod

Coefficient	*Estimate*	*Standard Error*	*P-Value*
(Intercept)	1.7690000	0.0483600	< 2.2e-16
Energy use (kg of oil equivalent per capita)	0.0014197	0.0000292	< 2.2e-16
Electric power consumption (kWh per capita)	0.0001268	0.0000188	1.69e-11
Fossil fuel energy consumption (% of total)	0.0155190	0.0011850	< 2e-16
Combustible renewables and waste (% of total energy)	-0.0323670	0.0022200	< 2e-16
Alternative and nuclear energy (% of total energy use)	-0.0879560	0.0055910	< 2e-16
Electric power transmission and distribution losses (% of output)	-0.0223150	0.0052270	1.9800e-05

We then graphically verify constant variance assumption to infer a confidence interval for coefficient estimates. Based on the visual, we can see that constant variance assumption does not for our data. So, we then use robust standard error to check the significance of our variables and find confidence intervals for our estimates.

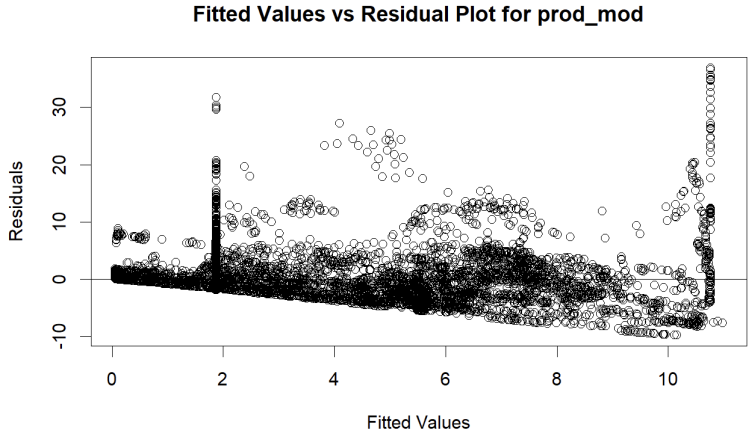


Figure 23. prod_mod Constant Variance Assumption Assessment

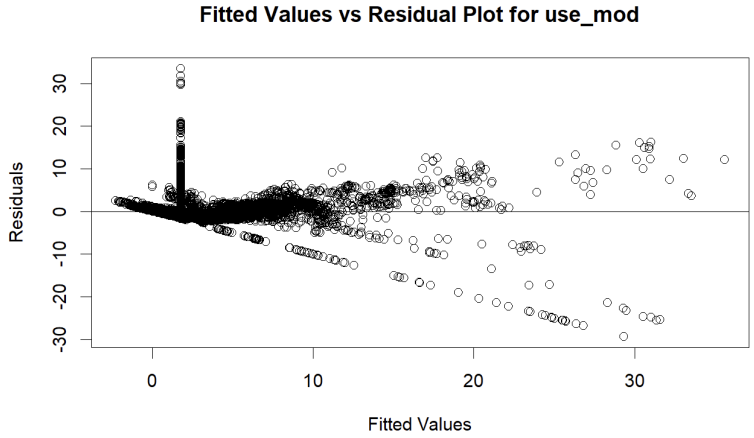


Figure 24. use_mod Constant Variance Assumption Assessment

In Table 5a, we can observe the coefficients and their robust standard error for all predictors from energy production indicators of *prod_mod* model. In Table 5b, we can observe the coefficients and their robust standard error for all predictors from energy consumption indicators of use_*mod* model. At the

significance level of 0.05, one variable that is not significant (compared to using regular standard error) is Electric power consumption (kWh per capita). This shows that our data indeed does not have constant variance.

Table 5a: Linear Regression Output with Robust standard of prod_mod

Coefficient	*Estimate*	*Standard Error*	*P-Value*
(Intercept)	1.87460	0.05243	< 2.2e-16
Electricity production from coal sources (% of total)	0.03812	0.00231	< 2.2e-16
Electricity production from hydroelectric sources (% of total)	-0.01812	0.00123	< 2.2e-16
Electricity production from natural gas sources (% of total)	0.08892	0.00409	< 2.2e-16
Electricity production from nuclear sources (% of total)	0.09659	0.00421	< 2.2e-16
Electricity production from oil sources (% of total)	-0.00009	0.00221	0.96710
Electricity production from renewable sources, excluding hydroelectric (% of total)	0.06274	0.00802	5.965e-15

Table 5b: Linear Regression Output with Robust standard of use_mod

Coefficient	*Estimate*	*Standard Error*	*P-Value*
(Intercept)	1.7693209	0.0535570	< 2.2e-16
Energy use (kg of oil equivalent per capita)	0.0014197	0.00014	< 2.2e-16
Electric power consumption (kWh per capita)	0.0001268	0.00009	0.13980
Fossil fuel energy consumption (% of total)	0.0155190	0.00256	1.392e-09
Combustible renewables and waste (% of total energy)	-0.0323670	0.00113	< 2.2e-16
Alternative and nuclear energy (% of total energy use)	-0.0879560	0.00797	< 2.2e-16
Electric power transmission and distribution losses (% of output)	-0.0223150	0.00417	9.088e-08

With the robust standard error, we can create 95% confidence intervals for our predictor variables. In Table 6a, we can observe the 95% confidence intervals for all predictors from energy production indicators of *prod_mod* model. In Table 6b, we can observe the 95% confidence intervals for all predictors from energy consumption indicators of use_*mod* model. Using robust standard to find 95% confidence intervals confirms that the coefficients for "Electricity production from oil sources (% of total)" and "Electric power consumption (kWh per capita)" could be zeros based on our data.

Table 6a: 95% Confidence Intervals for estimates of predictors from prod_mod

Coefficient	*Lower Limit*	*Upper Limit*
(Intercept)	1.77183	1.97738
Electricity production from coal sources (% of total)	0.03359	0.04265
Electricity production from hydroelectric sources (% of total)	-0.02053	-0.01571
Electricity production from natural gas sources (% of total)	0.08090	0.09693
Electricity production from nuclear sources (% of total)	0.08833	0.10485
Electricity production from oil sources (% of total)	-0.00443	0.00425
Electricity production from renewable sources, excluding hydroelectric (% of total)	0.04701	0.07846

Table 6b: 95% Confidence Intervals for estimates of predictors from use_mod

Coefficient	*Lower Limit*	*Upper Limit*
(Intercept)	1.66433684	1.87430497
Energy use (kg of oil equivalent per capita)	0.00115	0.00169
Electric power consumption (kWh per capita)	-0.00004	0.00030
Fossil fuel energy consumption (% of total)	0.01050	0.02054
Combustible renewables and waste (% of total energy)	-0.03458	-0.03015
Alternative and nuclear energy (% of total energy use)	-0.10359	-0.07233
Electric power transmission and distribution losses (% of output)	-0.03049	-0.01414

Urbanization Metrics (Baisakhi)

We conducted a simple linear regression analysis to investigate the relationship between urban population (% of total population) and CO_2 emissions (metric tons per capita). Upon initial evaluation of the model, we observed signs of heteroscedasticity as shown in Figure 26 below, indicating that the variance of the residuals was not constant across all levels of urban population percentage. To address this issue, we employed robust standard errors in our regression analysis. The null hypothesis for our hypothesis testing was that there is no correlation between urban population increase and CO_2 emissions increase for any country. Our alternative hypothesis was that there is an association between urban population and CO_2 emissions for any country.

Our Linear Regression Model is:

$$CO2\ Emissions = \beta_0 + \beta_1 * (Urban\ Population) + \epsilon$$

Where CO_2 Emissions is the response variable, β_0 is the intercept variable, β_1 is the slope for urban population, Urban population is the explanatory variable/independent variable and epsilon ϵ is the error variable. Figure 25 shows the scatterplot and the linear relationship between our 2 variables of interest and the red line shows the best-fit line/fitted line of our linear regression model.

After implementing robust standard errors, we re-evaluated the model and found that the heteroscedasticity issue was effectively mitigated. The use of robust standard errors provided more accurate estimates of the regression coefficients, allowing for more reliable inference regarding the relationship between urban population percentage and CO_2 emissions.

The results revealed that the intercept value was -3.2657, indicating the estimated CO_2 emissions when the urban population percentage is zero. Additionally, the slope value for urban population percentage was 0.1557, representing the change in CO_2 emissions for a one-unit increase in urban population percentage. Our test statistic value was 0.901619 and our p-value from the hypothesis testing using simple linear regression model is 8.169642682111703e-19. Since our p-value is extremely small, hence by using a significance level of 0.05 we can reject our null hypothesis and conclude that there is an association between urban population and CO_2 emissions for a country. Detailed results are provided below in tables 7 and 8.

Hence, to summarize we can say, our findings suggest that urban population percentage is statistically significant in predicting CO_2 emissions, even after accounting for heteroscedasticity. The robust standard errors helped to ensure the validity of our regression analysis and provided more robust estimates of the regression coefficients.

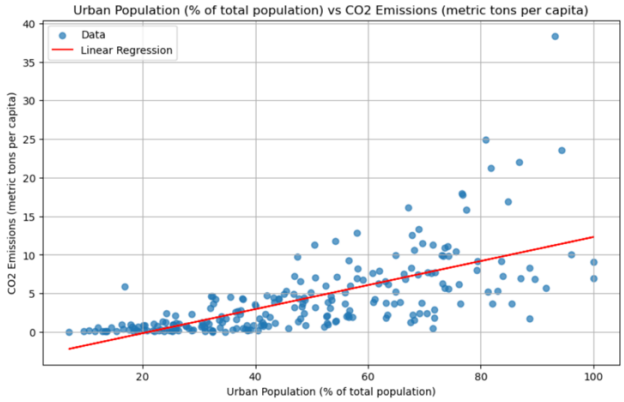


Figure 25. Urban Population (% of total population) VS. CO_2 Emissions (metric tons per capita)

Table 7: Linear Regression Results with Robust Standard Errors

Coefficient	*Estimate*	*Standard Error*	*Test Stats*	*P-Value*	*Lower Limit*	*Upper Limit*
(Intercept)	-3.2657	0.652	- 5.461	0.000000	- 4.545	- 1.987
Urban Population	0.1557	0.018	13.704	0.000000	0.121	0.190

Table 8: Linear Regression Results with Standard Errors

Coefficient	*Estimate*	*Standard Error*	*Test Stats*	*P-Value*	*Lower Limit*	*Upper Limit*
(Intercept)	-3.2657	0.598	- 5.005	0.000000	- 4.444	- 2.087
Urban Population	0.1557	0.011	8.858	0.000000	0.133	0.178

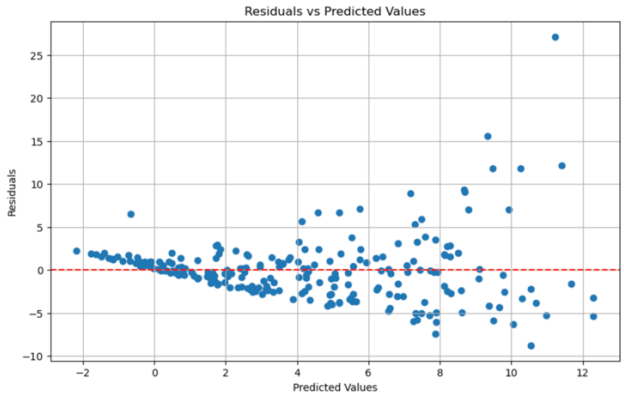


Figure 26. Residuals VS. Predicted Values of Linear Regression Model

Discussion

In this analysis, we aimed to investigate how CO_2 emissions have changed in recent years and how forestry, socioeconomic, energy production, and urbanization metrics are associated with CO_2 emissions. In analyzing the mean difference in CO_2 emissions, our conclusions vary when considering different perspectives. While examining the 30 countries with the least mean CO_2 emissions, we find evidence supporting a non-zero mean difference between the two time periods within this group, whereas we fail to

conclude the same when evaluating the global mean and means of the top 30 countries with the most emission. Additionally, the United States and Ethiopia exhibit significant differences in mean CO_2 emissions when comparing the two time periods. In contrast, Qatar and Burundi lack sufficient evidence to draw similar conclusions regarding changes in their mean CO_2 emissions over the specified time intervals.

Upon analysis of recent carbon dioxide emissions and forestry-related metrics, we only have sufficient evidence that agriculture, forestry, fishing value added to GDP has a relationship with carbon dioxide emissions. This conclusion may differ from analyses using a longer timeframe and can only reflect the past thirty years.

Examining socioeconomic indicators, we found that prevalence of overweight adults, the voice and accountability metric, industry value, and urban population were both statistically and practically significant indicators for increased CO_2 emissions. This aligns with the common notion that more developed countries are contributing more CO_2 emissions. Developed countries are more likely to have greater proportions of overweight adults in their population which may be a metric for overconsumption and may increase demand for industries that generate CO_2 emissions, such as the livestock industry. Similar conclusions can be said about voice and accountability metrics, industry value, and urban population. Interestingly, education spending was found to be negatively associated with CO_2 emissions as well as being statistically and practically significant. This perhaps shows that educating people about climate change and CO_2 emissions may have an impact which influences individual behavior to be more carbon conscious.

We have strong evidence that electricity production and consumption of gas combustions, including coal, natural gas, and fossil fuels, contribute to increasing CO_2 emissions. Although production from renewable sources is positively correlated with CO_2 emissions, energy consumption from combustible renewable sources has a negative coefficient, so the use of renewable sources might decrease CO_2. Same situation for nuclear energy, although the production is positively correlated with CO_2 emissions, the consumption is negatively related. It is possible that a discrepancy exists between energy production and the actual consumption of nuclear and renewable energy. It is advisable to promote hydroelectric power use to reduce CO2 emissions. Furthermore, higher energy consumption led to higher CO_2 emissions, but we are unsure if this is due to electricity consumption specifically.

From our investigation of urbanization metrics, we can say that urbanization is positively correlated with CO_2 emissions. Although we are unable to determine causation in this analysis, it would make sense that a higher proportion of urban population would have higher CO_2 emissions.

One notable limitation lies in the relatively brief time frame covered by our CO_2 emissions data, spanning from 1990 to 2020. A more comprehensive understanding of the consequences associated with a global increase in CO_2 emissions could be achieved with an extended dataset that encompasses a broader historical context. A future direction includes using these findings to predict future carbon dioxide emission trends based off the predictors analyzed in this report. It is also interesting to note that we do not necessarily know how long it takes for changes in socioeconomic and environmental factors to be reflected in carbon dioxide emissions. Perhaps the changes in carbon dioxide emissions that we see today are a result of actions taken much earlier in time.

Reference

2023 was the warmest year in the modern temperature record | NOAA Climate.gov. (2024a, January 17). http://www.climate.gov/news-features/featured-images/2023-was-warmest-year-modern-temperature-record

Climate Change: Global Temperature | NOAA Climate.gov. (2024b, January 18). http://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature

IUCN. (2021, February). *Forests and climate change*. https://www.iucn.org/resources/issues-brief/forests-and-climate-change

US EPA, O. (2016a, June 27). *Climate Change Indicators: Ocean Acidity* [Reports and Assessments]. https://www.epa.gov/climate-indicators/climate-change-indicators-ocean-acidity

US EPA, O. (2016b, July 1). *Climate Change Indicators: Health and Society* [Reports and Assessments]. https://www.epa.gov/climate-indicators/health-society

US EPA, O. (2021, April 16). *Impacts of Climate Change* [Overviews and Factsheets]. https://www.epa.gov/climatechange-science/impacts-climate-change

World Bank. (n.d.). *Methodologies*. https://datahelpdesk.worldbank.org/knowledgebase/articles/906531-methodologies

tliu2_analysis_code_only

Ted Liu

2024-03-09

```r
co2_df <- read.csv('./data/annual-co2-emissions-per-country.csv', stringsAsFactors=F)

mean_emissions_per_year <- co2_df %>%
  group_by(Year) %>%
  dplyr::summarize(MeanValue = mean(Annual.CO..emissions, na.rm = TRUE), .groups = 'drop') %>%
  filter(!is.na(MeanValue))

p1 <- ggplot(mean_emissions_per_year, aes(x = Year, y = MeanValue)) +
  geom_line() +
  theme_minimal() +
  labs(x = "Year", y = "Average Global CO2 Emmisions", title = "Trend of Emissions Over Time")

p1
```

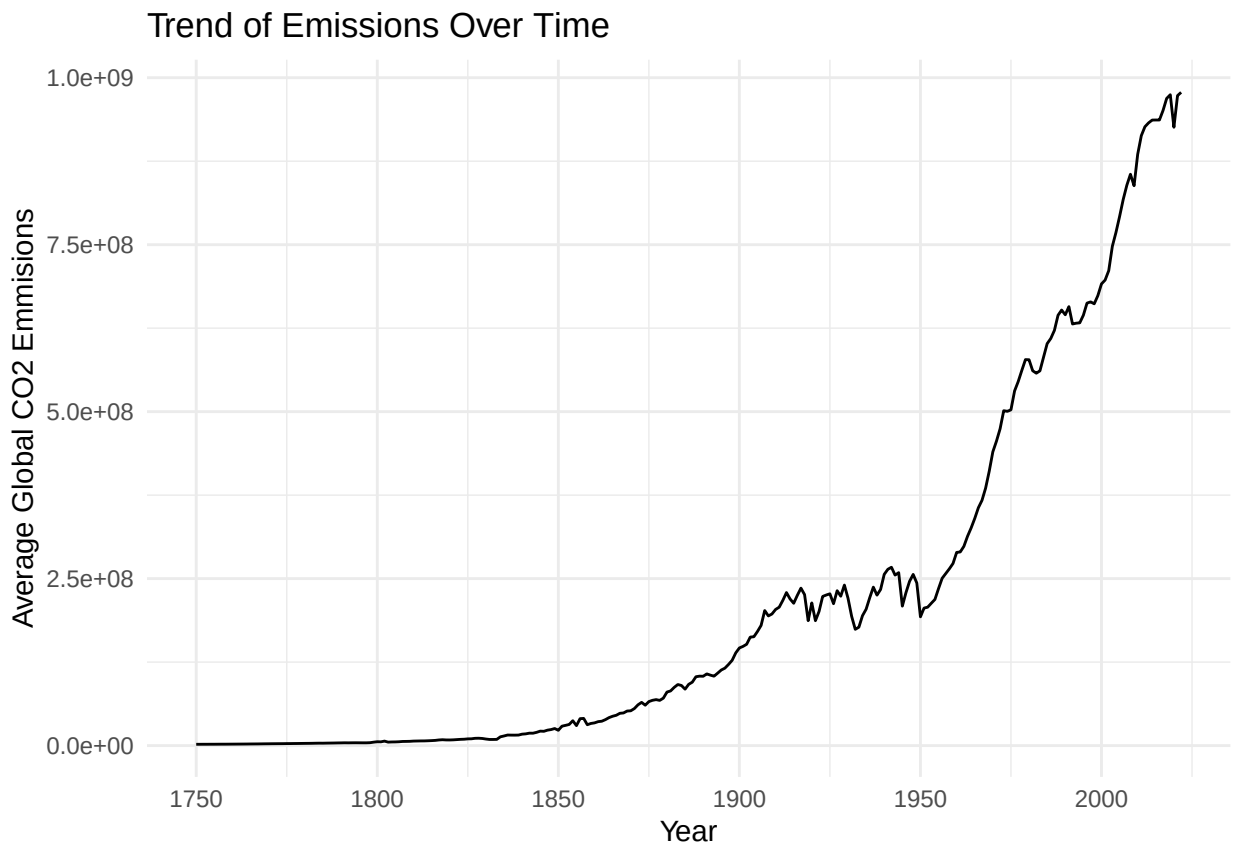


Figure 1: CO2 Emissions Over Time

```
wdi_src <- read.csv('./data/WDI_CSV/WDIData.csv', stringsAsFactors = F)
esg_src <- read.csv('./data/ESG_CSV/ESGData.csv', stringsAsFactors = F)
```

```
group_names <- c("Arab World", "Caribbean small states", "Central Europe and the Baltics",
                 "Early-demographic dividend", "East Asia & Pacific",
                 "Euro area", "Europe & Central Asia", "European Union",
                 "High income", "IBRD only", "IDA & IBRD total", "IDA blend",
                 "Low & middle income", "Middle East & North Africa",
                 "North America", "OECD members", "Post-demographic dividend",
                 "Pre-demographic dividend", "Small states", "South Asia",
                 "Sub-Saharan Africa", "Upper middle income", "World",
                 "East Asia & Pacific (excluding high income)", "East Asia & Pacific (IDA & IBRD)",
                 "Europe & Central Asia (excluding high income)", "Europe & Central Asia (IDA & IBRD)",
                 "Heavily indebted poor countries (HIPC)", "IDA only", "IDA total", "Late-demographic di
                 "Latin America & Caribbean", "Latin America & Caribbean (excluding high income)",
                 "Latin America & Caribbean (IDA & IBRD)", "Least developed countries: UN classificatio
                 "Low income", "Lower middle income", "Middle East & North Africa (excluding high incom
                 "Middle East & North Africa (IDA & IBRD)", "Middle income", "Other small states",
                 "Pacific island small states", "South Asia (IDA & IBRD)", "Sub-Saharan Africa (excludi
                 "Sub-Saharan Africa (IDA & IBRD)", "Africa Eastern and Southern", "East Asia & Pacific
                 "Middle East & North Africa (IDA & IBRD countries)", "Not classified", "Sub-Saharan Af
                 )
```

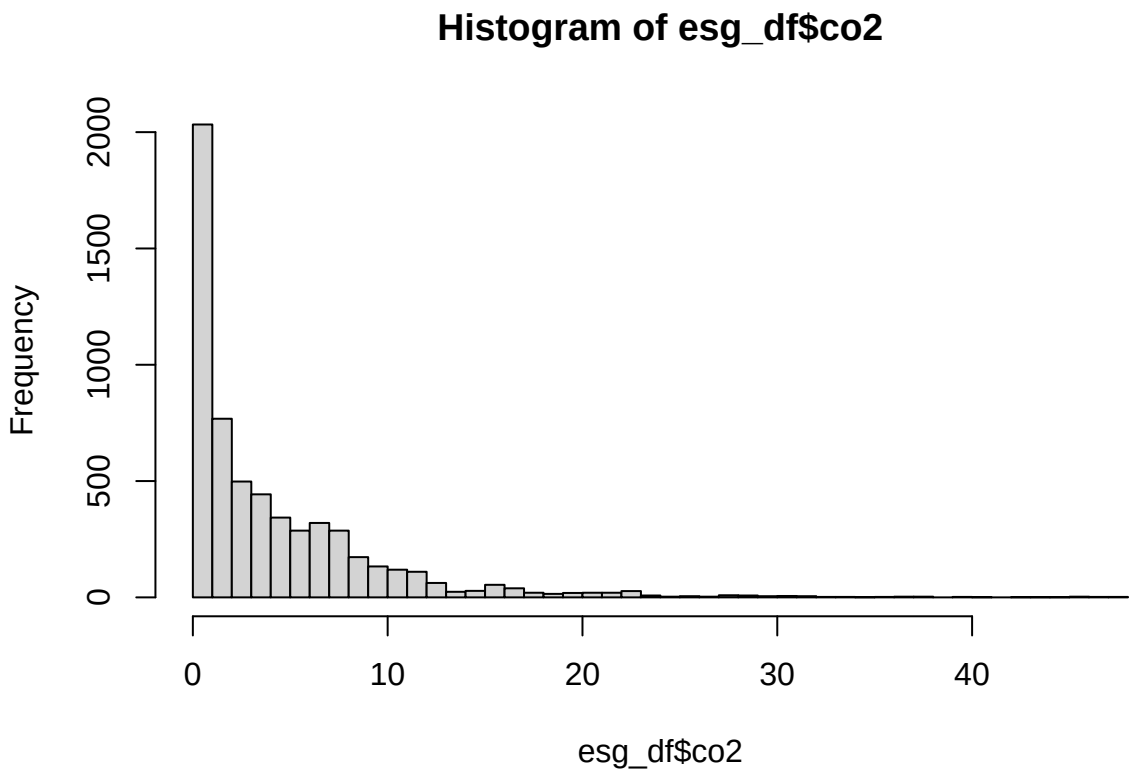
```r
esg_indicators <- c(
  'CO2 emissions (metric tons per capita)', "Proportion of seats held by women in national parliaments
  'GDP growth (annual %)', 'Population density (people per sq. km of land area)',
  'School enrollment, primary (% gross)', 'Voice and Accountability: Estimate', 'Rule of Law: Estimate'
)

wdi_indicators <- c(
  'CO2 emissions (metric tons per capita)', "Proportion of seats held by women in national parliaments
  'GDP growth (annual %)', 'Population density (people per sq. km of land area)',
  'School enrollment, primary (% gross)', 'Voice and Accountability: Estimate', 'Rule of Law: Estimate'
)

wdi_df <- wdi_src %>%
  filter(Indicator.Name %in% wdi_indicators) %>%
  filter(!(Country.Name %in% group_names)) %>%
  pivot_longer(cols = starts_with("X"), names_to = "Year", values_to = "Value", names_prefix = "X") %>%
  mutate(Year = as.numeric(sub("X", "", Year)))  %>%
  group_by(Country.Name, Country.Code, Year, Indicator.Name) %>%
  dplyr::summarise(Value = first(Value), .groups = 'drop') %>%
  pivot_wider(names_from = Indicator.Name, values_from = Value) %>%
  filter(!is.na(Year))

esg_df <- esg_src %>%
  filter(Indicator.Name %in% esg_indicators) %>%
  filter(!(Country.Name %in% group_names)) %>%
  pivot_longer(cols = starts_with("X"), names_to = "Year", values_to = "Value", names_prefix = "X") %>%
  mutate(Year = as.numeric(sub("X", "", Year)))  %>%
  group_by(Country.Name, Country.Code, Year, Indicator.Name) %>%
  dplyr::summarise(Value = first(Value), .groups = 'drop') %>%
  pivot_wider(names_from = Indicator.Name, values_from = Value) %>%
  filter(!is.na(Year))

wdi_df <- wdi_df %>%
  rename(
    co2 = `CO2 emissions (metric tons per capita)`,
    gdp.growth = `GDP per capita growth (annual %)`,
    gdp.capita = `GDP per capita (current US$)`,
    industry = `Industry (including construction), value added (% of GDP)`,
    econ.sustain = `Adjusted net savings, excluding particulate emission damage (% of GNI)`,
    urban.pop = `Urban population (% of total population)`,
    # access.electric = `Access to electricity (% of population)`,
    # r.and.d = `Research and development expenditure (% of GDP)`,
    transport.service = `Transport services (% of commercial service exports)`,
    # clean.cooking = `Access to clean fuels and technologies for cooking (% of population)`
    # agri.land = `Agricultural land (% of land area)`,
    pop.density = `Population density (people per sq. km of land area)`,
    # agri.land = `Agricultural land (% of land area)`,
    # agri.value = `Agriculture, forestry, and fishing, value added (% of GDP)`,
    renew.energy = `Renewable electricity output (% of total electricity output)`,
    prop.women = `Proportion of seats held by women in national parliaments (%)`,
    school.enroll = `School enrollment, primary (% gross)`,
    citizen.voice = `Voice and Accountability: Estimate`,
```

```
    rule.of.law = `Rule of Law: Estimate`,
    # clean.cooking = `Access to clean fuels and technologies for cooking (% of population)`
    edu.spend = `Adjusted savings: education expenditure (current US$)`,
    army = `Armed forces personnel (% of total labor force)`,
    corrupt.control = `Control of Corruption: Estimate`,
    reg.quality = `Regulatory Quality: Estimate`,
    pop.growth = `Population growth (annual %)`,
    net.income = `Adjusted net national income (annual % growth)`
  )


esg_df <- esg_df %>%
  rename(
    co2 = `CO2 emissions (metric tons per capita)`,
    overweight = `Prevalence of overweight (% of adults)`,
    # gdp.growth = `GDP per capita growth (annual %)`,
    # gdp.capita = `GDP per capita (current US$)`,
    pop.density = `Population density (people per sq. km of land area)`,
    # agri.land = `Agricultural land (% of land area)`,
    # agri.value = `Agriculture, forestry, and fishing, value added (% of GDP)`,
    renew.energy = `Renewable electricity output (% of total electricity output)`,
    prop.women = `Proportion of seats held by women in national parliaments (%)`,
    school.enroll = `School enrollment, primary (% gross)`,
    citizen.voice = `Voice and Accountability: Estimate`,
    rule.of.law = `Rule of Law: Estimate`,
    corrupt.control = `Control of Corruption: Estimate`,
    reg.quality = `Regulatory Quality: Estimate`
    # clean.cooking = `Access to clean fuels and technologies for cooking (% of population)`
  )

wdi_df$scale_edu.spend <- scale(wdi_df$edu.spend)

wdi_df$log_co2 <- log(wdi_df$co2)
esg_df$log_co2 <- log(esg_df$co2)

merge <- esg_df %>%
  left_join(wdi_df, by = c('Country.Name', 'Country.Code', 'Year'), suffix = c('.esg', '.wdi')) %>%
  mutate(
    co2 = coalesce(co2.esg, co2.wdi),
    log_co2 = coalesce(log_co2.esg, log_co2.wdi),
    # clean.cooking = coalesce(clean.cooking.esg, clean.cooking.wdi),
    # gdp.growth = coalesce(gdp.growth.esg, gdp.growth.wdi),
    # gdp.capita = coalesce(gdp.capita.esg, gdp.capita.wdi),
    pop.density = coalesce(pop.density.esg, pop.density.wdi),
    prop.women = coalesce(prop.women.esg, prop.women.wdi),
    school.enroll = coalesce(school.enroll.esg, school.enroll.wdi),
    rule.of.law = coalesce(rule.of.law.esg, rule.of.law.wdi),
    citizen.voice = coalesce(citizen.voice.esg, citizen.voice.wdi),
    corrupt.control = coalesce(corrupt.control.esg, corrupt.control.wdi),
    reg.quality = coalesce(reg.quality.esg, reg.quality.wdi)
  ) %>%
  select(-contains('.wdi'), -contains('.esg')) %>%
  filter(complete.cases(.))
```

```r
write.csv(merge, "merge.csv")
```

```r
lm.model <- lm(
    co2 ~ overweight + gdp.growth + pop.density + prop.women + citizen.voice + scale_edu.spend + industr
    data = merge
)
```

```r
hist(esg_df$co2, breaks = 50)
```




```r
par(mfrow=c(2, 2))
plot(fitted(lm.model), lm.model$residuals, main = "Residuals vs Fitted Values")
abline(0,0)

qqnorm(lm.model$residuals)
qqline(lm.model$residuals)

plot(density(lm.model$residuals))
# Reset to default
par(mfrow=c(1, 1))
```

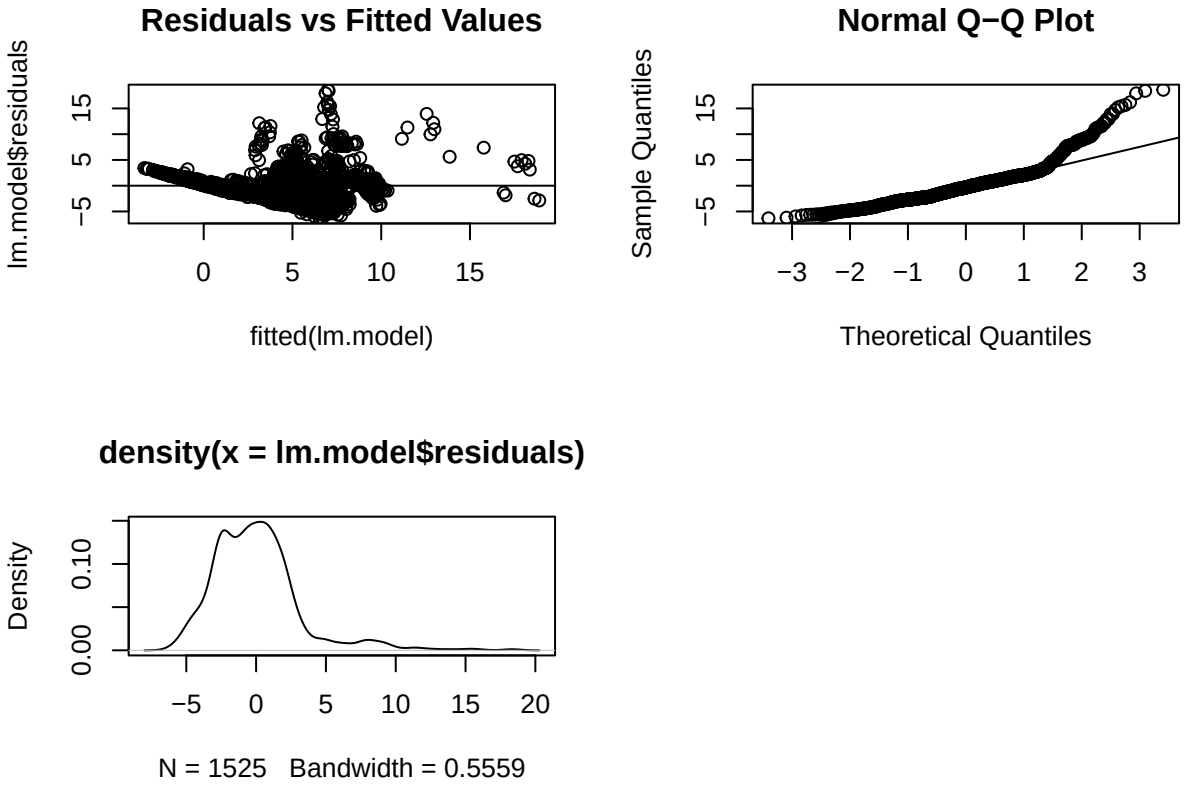


Figure 2: Assess Assumption Plots

```
lm.model.log <- lm(
  log_co2 ~ overweight + gdp.growth + pop.density + prop.women + citizen.voice + scale_edu.spend + indus
  data = merge
)
```

```
par(mfrow=c(2, 2))
plot(fitted(lm.model.log), lm.model.log$residuals, main = "Residuals vs Fitted Values")
abline(0,0)
qqnorm(lm.model.log$residuals)
qqline(lm.model.log$residuals)
plot(density(lm.model.log$residuals))
# Reset to default
par(mfrow=c(1, 1))
```

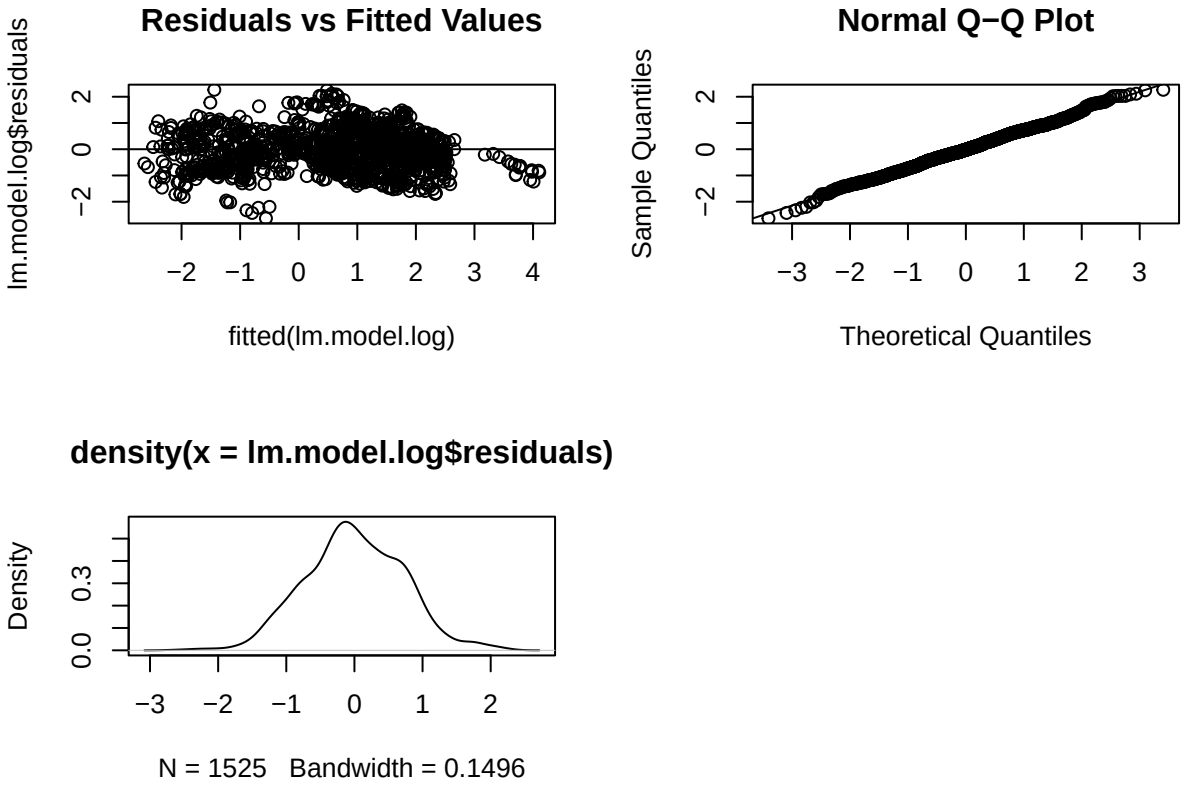


Figure 3: Assess Assumption Plots Log Transformed

```r
fixed.model <- lm(
  log_co2 ~ Year + overweight + gdp.growth + pop.density + prop.women + citizen.voice + scale_edu.spend
  data = merge
)
# summary(fixed.model, correlation=F)

# Exponentiate the relevant coefficients and calculate the percentage change
percentage_changes <- (exp(coef(fixed.model)) - 1) * 100
options(scipen = 999)
# print(percentage_changes)


## Calculate 95% confidence intervals for all model coefficients
all_conf_ints <- confint(fixed.model, level = 0.95)

# Get all coefficient names
all_coeff_names <- names(coef(fixed.model))

coeff_names <- all_coeff_names[!grepl("Country", all_coeff_names)]
coeff_names <- setdiff(coeff_names, "Year")
selected_conf_ints <- all_conf_ints[coeff_names, ]

exp_selected_conf_ints <- exp(selected_conf_ints) - 1
```

```r
# Convert to percentage change
percentage_change_conf_ints <- exp_selected_conf_ints * 100

print(selected_conf_ints)
```

```
##                          2.5 %       97.5 %
## (Intercept)     22.21965937248 52.094477602
## overweight       0.03074686346  0.062694067
## gdp.growth       0.00013917472  0.005016997
## pop.density      0.00006728745  0.001190991
## prop.women       0.00003592211  0.003961456
## citizen.voice    0.00910085250  0.111689799
## scale_edu.spend -0.04740669329 -0.001000647
## industry         0.01009508840  0.016119228
## urban.pop        0.00624926506  0.020318907
```

```r
print(percentage_change_conf_ints)
```

```
##                                2.5 %                              97.5 %
## (Intercept)     446555512768.923828125 421060168956687784768204 4.0000000
## overweight                3.122443026                          6.4701062
## gdp.growth                0.013918440                          0.5029603
## pop.density               0.006728971                          0.1191700
## prop.women                0.003592275                          0.3969313
## citizen.voice             0.914239118                         11.8165951
## scale_edu.spend          -4.630054446                         -0.1000147
## industry                  1.014621570                          1.6249843
## urban.pop                 0.626883245                          2.0526742
```

```r
par(mfrow=c(2, 2))
plot(fitted(fixed.model), resid(fixed.model), main = "Residuals vs Fitted Values")
abline(0,0)
qqnorm(resid(fixed.model))
qqline(resid(fixed.model))
plot(density(resid(fixed.model)))
# Reset to default
par(mfrow=c(1, 1))
```

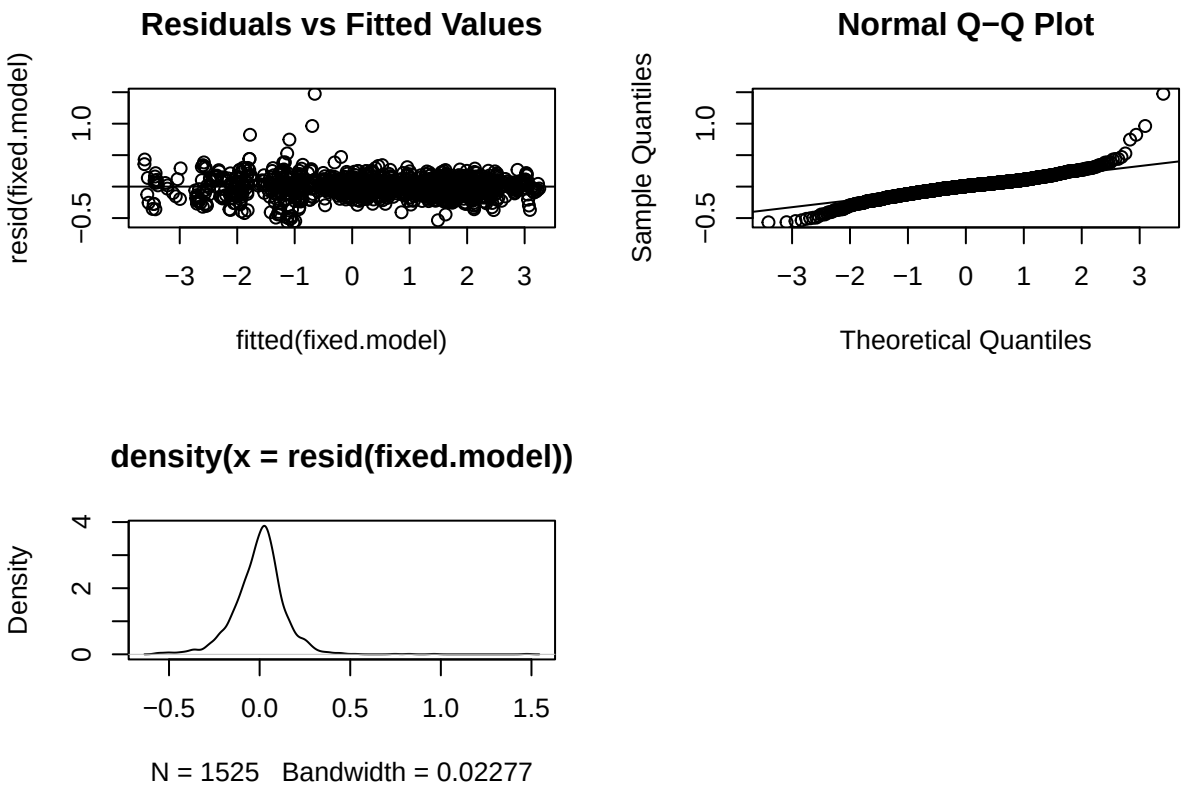


Figure 4: Assess Assumption Plots Mixed Effects

```r
# Calculate 95% confidence intervals
conf_ints <- confint(fixed.model, level = 0.95)

model_summary <- summary(fixed.model)

coefficients <- coef(fixed.model)
coeff_names <- names(coefficients)
p_values <- summary(fixed.model)$coefficients[,4]

excluded_terms <- grep("Country", coeff_names, value = TRUE)
coeff_names <- setdiff(coeff_names, excluded_terms)

selected_coeffs <- coefficients[coeff_names]
selected_conf_ints <- conf_ints[coeff_names, ]
selected_p_values <- p_values[coeff_names]

coeffs_df <- data.frame(
  Coefficient = coeff_names,
  Estimate = selected_coeffs,
  Lower = selected_conf_ints[, 1],
  Upper = selected_conf_ints[, 2],
  P_Value = selected_p_values
)
```

```r
# Use knitr::kable() to create the table
knitr::kable(coeffs_df, format = "simple", caption = "Model Coefficients Including Year and P-values, E
  kableExtra::kable_styling(latex_options = c("striped", "scale_down"))
```

```
## Warning in kableExtra::kable_styling(., latex_options = c("striped",
## "scale_down")): Please specify format in kable. kableExtra can customize either
## HTML or LaTeX outputs. See https://haozhu233.github.io/kableExtra/ for details.
```

Table 1: Model Coefficients Including Year and P-values, Excluding Country Code Related Terms

	Coefficient	Estimate	Lower	Upper	P_Value
(Intercept)	(Intercept)	37.1570685	22.2196594	52.0944776	0.0000012
Year	Year	-0.0198028	-0.0273022	-0.0123033	0.0000003
overweight	overweight	0.0467205	0.0307469	0.0626941	0.0000000
gdp.growth	gdp.growth	0.0025781	0.0001392	0.0050170	0.0382990
pop.density	pop.density	0.0006291	0.0000673	0.0011910	0.0282135
prop.women	prop.women	0.0019987	0.0000359	0.0039615	0.0459575
citizen.voice	citizen.voice	0.0603953	0.0091009	0.1116898	0.0210497
scale_edu.spend	scale_edu.spend	-0.0242037	-0.0474067	-0.0010006	0.0409176
industry	industry	0.0131072	0.0100951	0.0161192	0.0000000
urban.pop	urban.pop	0.0132841	0.0062493	0.0203189	0.0002203

557 Group Project - Forestry Metrics

Elaine Zhang

2024-03-10

Data Setup & Cleaning

```
#load necessary libraries
library(ggplot2)
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
```

```
library(tidyr)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.2
```

```
## corrplot 0.92 loaded
```

```
#get regions list from world bank
regions <- read.csv("Countries and Regions.csv", header = T)
regions <- regions %>% rename("Country Code" = "Country.Code")
regions <- regions %>% filter(Region != "") #filter out regions from country list
```

```
#read forest data
forest <- read.csv("IMF Forest and Carbon.csv", header = T)
#filter data to just forest share
forest_filter <- subset(forest, forest$Indicator == "Share of forest area")
#pivot table
forest_pivot <- pivot_longer(forest_filter, cols = starts_with("F"),
                             names_to = "Year",  values_to = "Forest Share")
#clean up column names
forest_pivot$Year <- as.numeric(sub("F", "", forest_pivot$Year))
forest_data <- forest_pivot[, c("Country", "ISO3", "Year", "Forest Share", "Unit")]
forest_data <- forest_data %>% rename("Country Code" = "ISO3")
#join with regions and clean out regions
forest_data <- inner_join(forest_data, regions, by = "Country Code")
forest_data <- forest_data[, c("Country", "Country Code", "Year", "Forest Share", "Unit", "Region")]
#check for na values
sum(is.na(forest_data$`Forest Share`))
```

```
## [1] 0
```

```
#no nulls here!
```

```
#read in CO2 data
co2 <- read.csv("CO2.csv", header = T)
#pivot table
co2_pivot <- pivot_longer(co2, cols = starts_with("X"),
                          names_to = "Year",  values_to = "CO2")
co2_pivot$Year <- as.numeric(sub("X", "", co2_pivot$Year))
#rename and clean up columns
co2_data <- co2_pivot %>% rename("Country" = "Country.Name", "Country Code" = "Country.Code")
co2_data <- co2_data[, c("Country", "Country Code", "Year", "CO2")]
#check for blank values
sum(is.na(co2_data$CO2))
```

```
## [1] 63
```

```
#remove the countries with no CO2 data -> Monaco, San Marino
co2_data <- co2_data[complete.cases(co2_data),]
#join regions
co2_data <- co2_data %>% inner_join(regions, by = "Country Code")
co2_data <- co2_data[, c("Country", "Country Code", "Year", "CO2", "Region")]
```

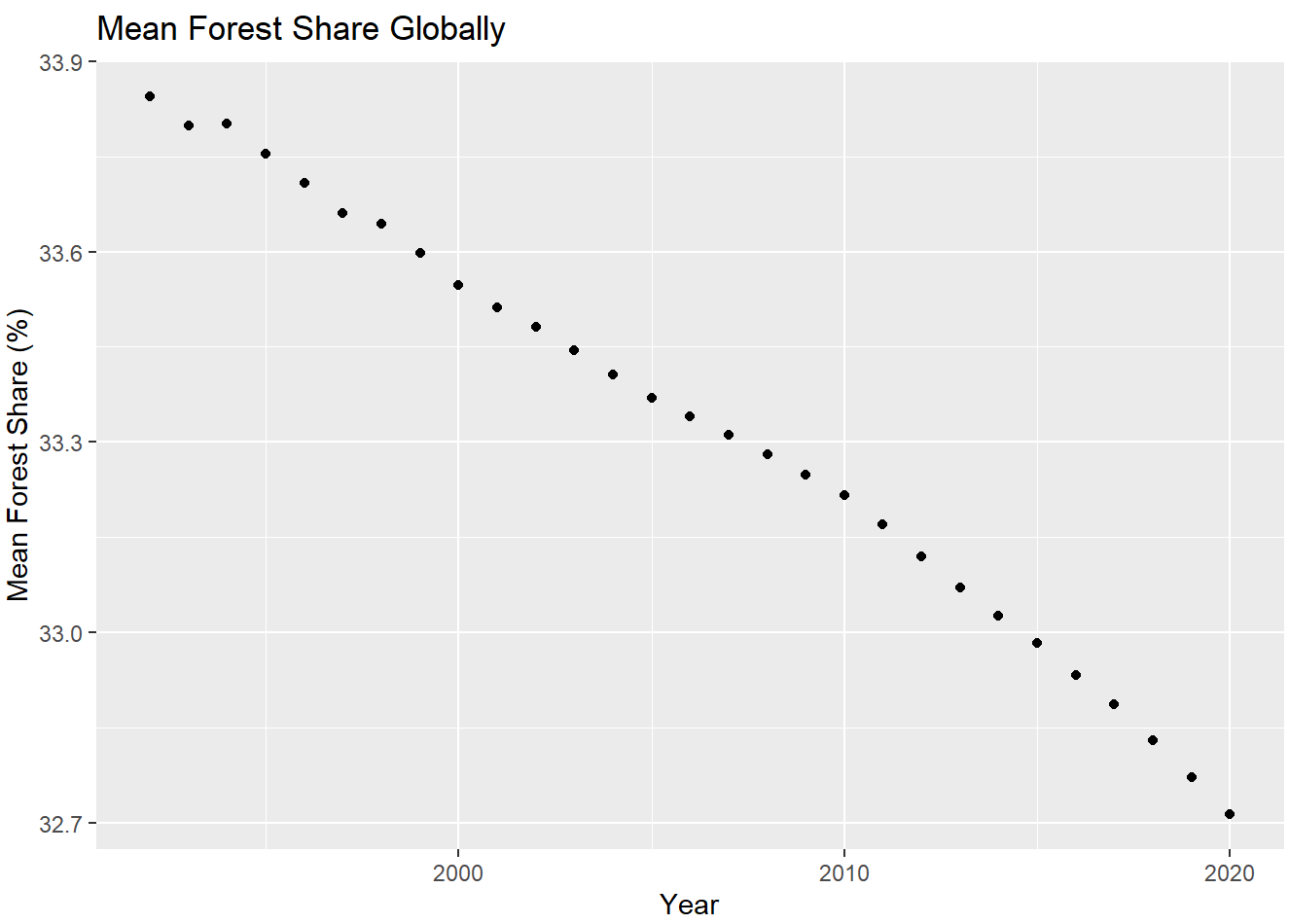
```
#read in agriculture GDP data
agri <- read.csv("Agriculture Value Added.csv", header = T)
#pivot table
agri_pivot <- pivot_longer(agri, cols = starts_with("X"),
                           names_to = "Year",  values_to = "Agriculture, Forestry, Fishing Value Added (% of
GDP)")
#clean up column names
agri_pivot$Year <- as.numeric(sub("X", "", agri_pivot$Year))
agri_data <- agri_pivot %>% rename("Country" = "Country.Name", "Country Code" = "Country.Code")
agri_data <- agri_data[, c("Country", "Country Code", "Year", "Agriculture, Forestry, Fishing Value Added
(% of GDP)")]
#add region
agri_data <- agri_data %>% inner_join(regions, by = "Country Code")
agri_data <- agri_data[, c("Country", "Country Code", "Year", "Agriculture, Forestry, Fishing Value Added
(% of GDP)", "Region")]
#filter data to the years in the CO2 data
agri_data <- inner_join(co2_data, agri_data, by = c("Country", "Year"))
agri_data <- agri_data[, c("Country", "Country Code.x", "Year", "Agriculture, Forestry, Fishing Value Added
(% of GDP)", "Region.x"),]
agri_data <- agri_data %>% rename("Country Code" = "Country Code.x", "Region" = "Region.x")
#fill in blank values with the mean for that country
agri_data <- agri_data %>% group_by(Country) %>% mutate(`Agriculture, Forestry, Fishing Value Added (% of G
DP)` = ifelse(is.na(`Agriculture, Forestry, Fishing Value Added (% of GDP)`), mean(`Agriculture, Forestry,
Fishing Value Added (% of GDP)`, na.rm = TRUE), `Agriculture, Forestry, Fishing Value Added (% of GDP)`))
```

```
#combining the datasets
data <- inner_join(inner_join(co2_data, forest_data, by = c("Country Code", "Year")), agri_data, by = c("Co
untry Code", "Year"))
#clean up columns
data <- data[, c("Country.x", "Country Code", "Year", "CO2", "Forest Share", "Agriculture, Forestry, Fishin
g Value Added (% of GDP)", "Region.x")]
data <- data %>% rename("Country" = "Country.x", "Region" = "Region.x")
# data[!complete.cases(data),] #no GDP indicator for N.Korea
data <- data[complete.cases(data),]
```

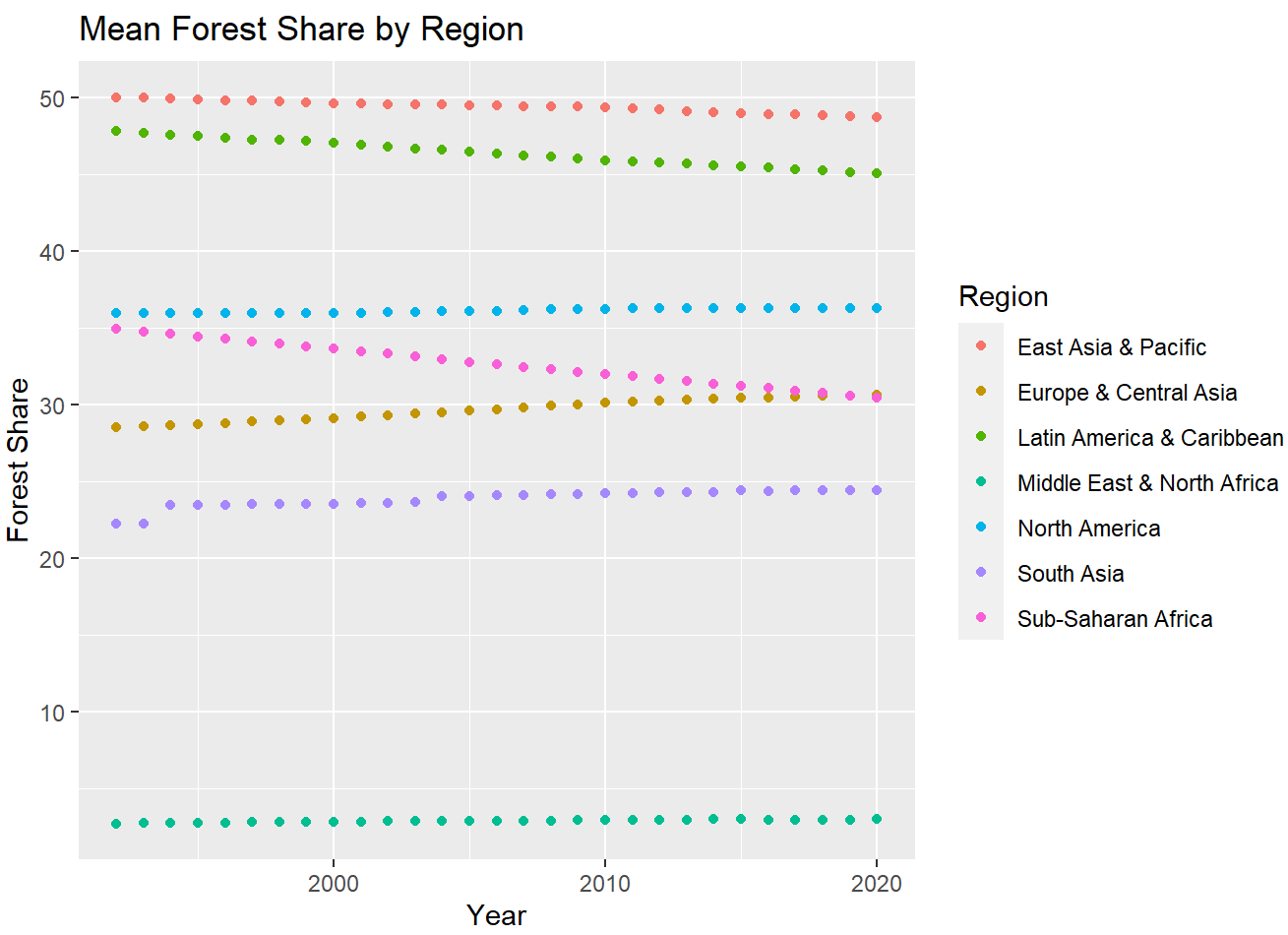
```
#create global/regional/country-level datasets
data_country <- data %>% group_by(Country) %>%
  summarize('Mean Agriculture, Forestry, Fishing Value Added (% of GDP)' = mean(`Agriculture, Forestry, Fis
hing Value Added (% of GDP)`), 'Mean Forest Share' = mean(`Forest Share`), 'Mean CO2 Emissions' = mean(`CO2
`), .groups="keep")
```

Exploratory Analysis

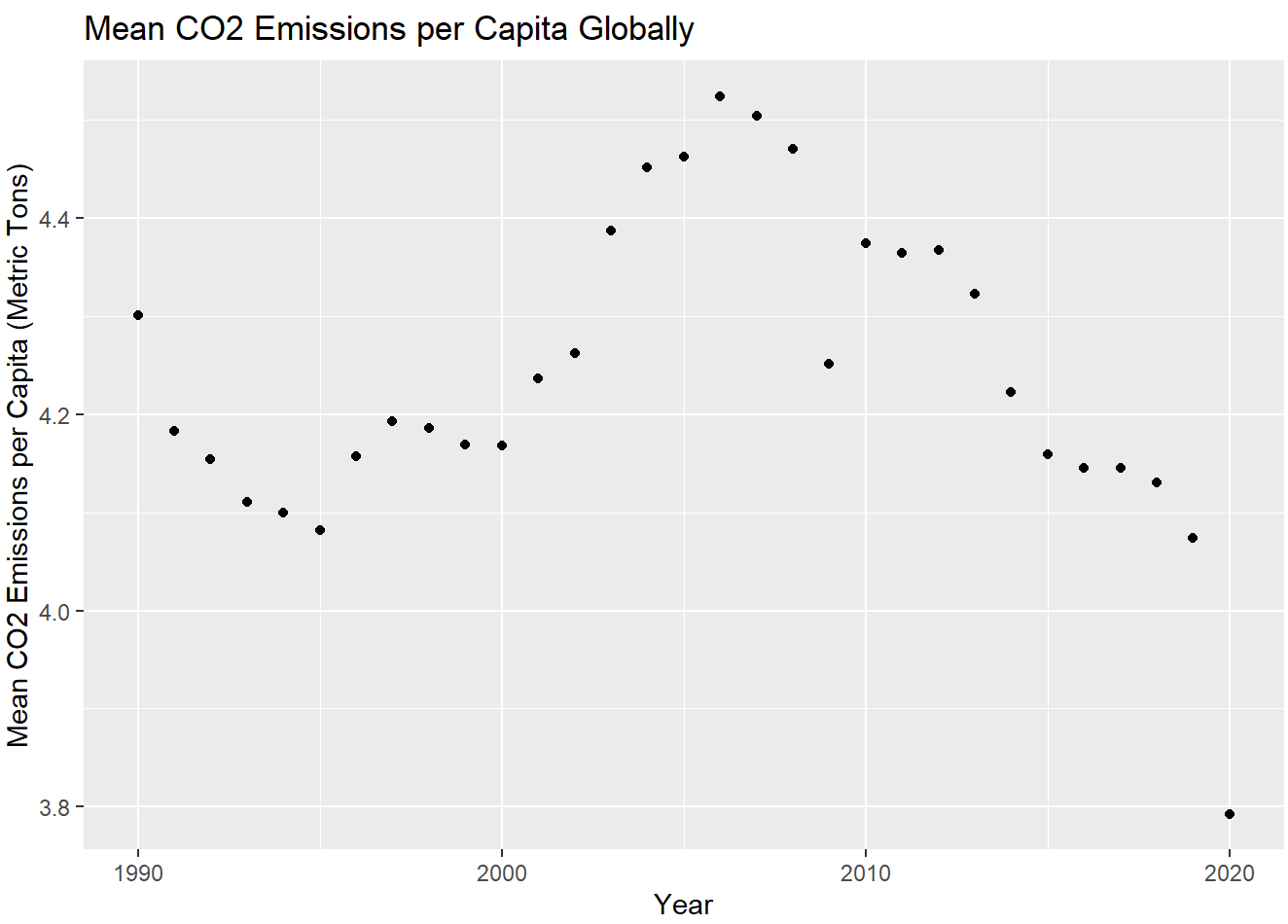
```
#overall global trend for forest share
forest_global <- forest_data%>% group_by(Year) %>% summarize('Mean Forest Share' = mean(`Forest Share`))

ggplot(forest_global, aes(x = Year, y = `Mean Forest Share`)) +
  geom_point(data = forest_global) +
  labs(title = "Mean Forest Share Globally", x = "Year", y = "Mean Forest Share (%)")
```



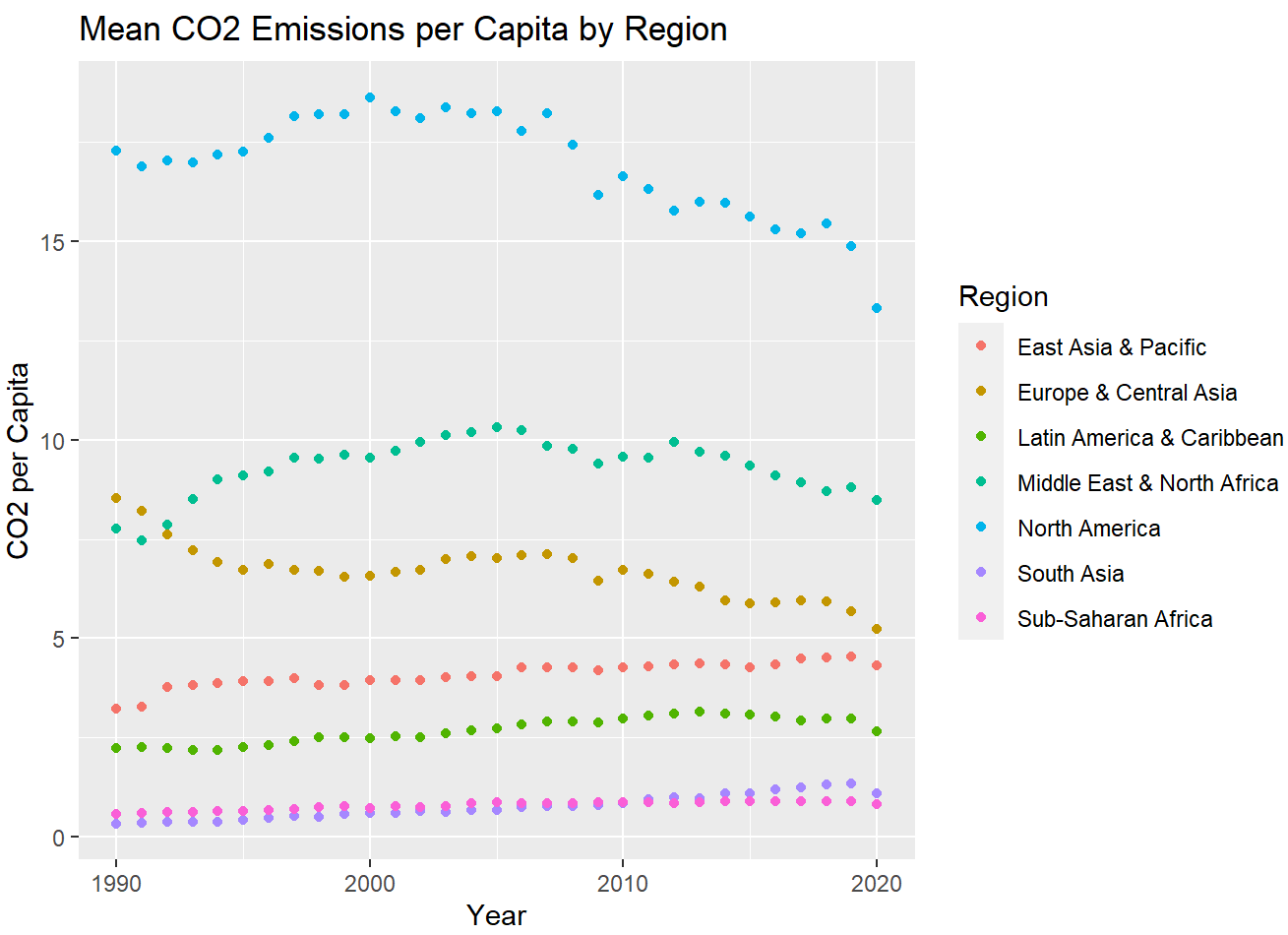

```
#look at forest share trends over time by region
forest_region <- na.omit(forest_data) %>% group_by(Region, Year) %>% summarize('Mean Forest Share' = mean(`
Forest Share`), .groups = "keep")

#plot them
ggplot(forest_region, aes(x = Year, y = `Mean Forest Share`, color = Region, linetype = Region)) +
  geom_point(data = forest_region) +
  labs(title = "Mean Forest Share by Region", x = "Year", y = "Forest Share")
```



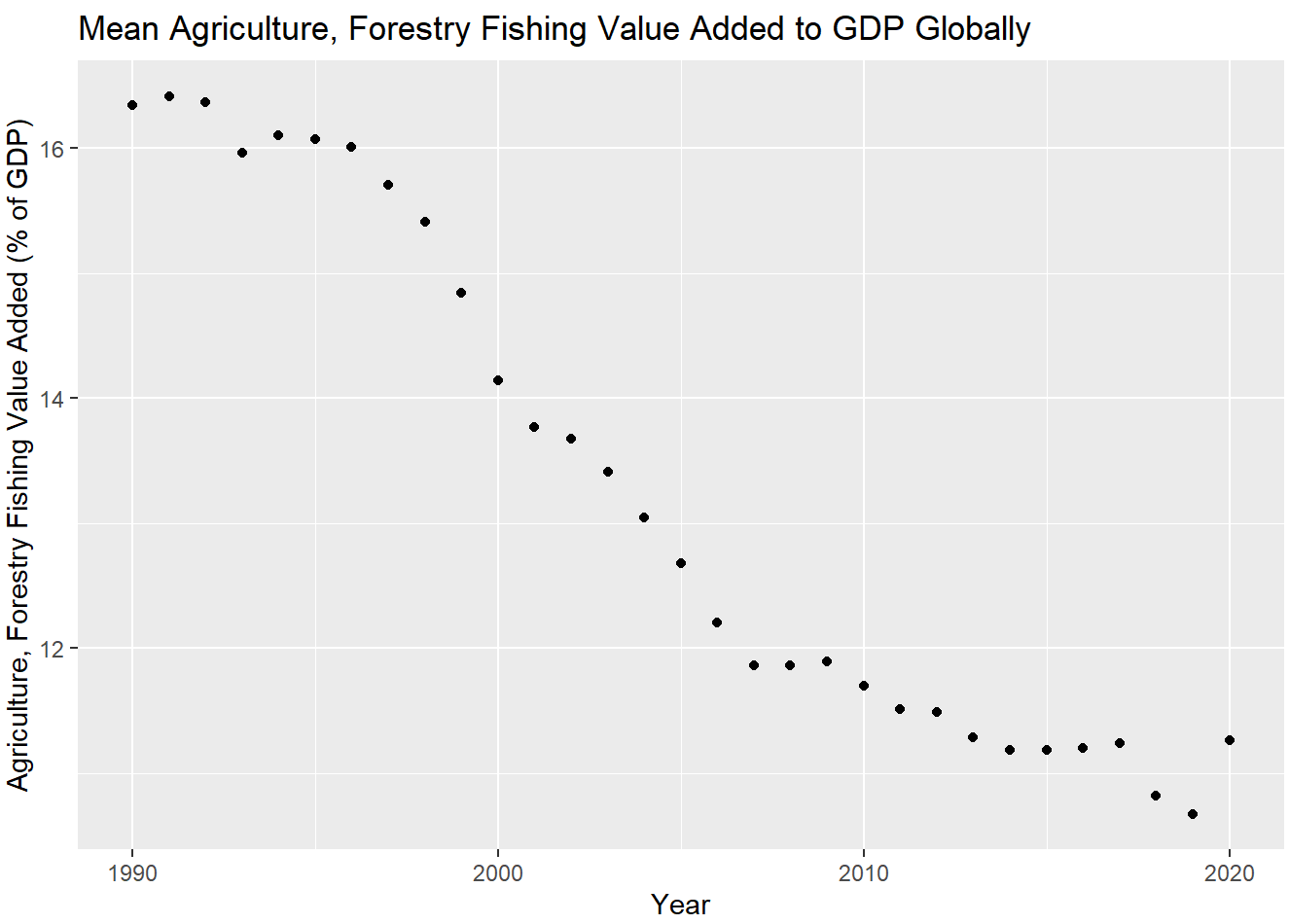

```
#overall global trend for forest share
co2_global <- co2_data%>% group_by(Year) %>% summarize('Mean CO2 Emissions per Capita' = mean(`CO2`))

ggplot(co2_global, aes(x = Year, y = `Mean CO2 Emissions per Capita`)) +
  geom_point(data = co2_global) +
  labs(title = "Mean CO2 Emissions per Capita Globally", x = "Year", y = "Mean CO2 Emissions per Capita (Me
tric Tons)")
```



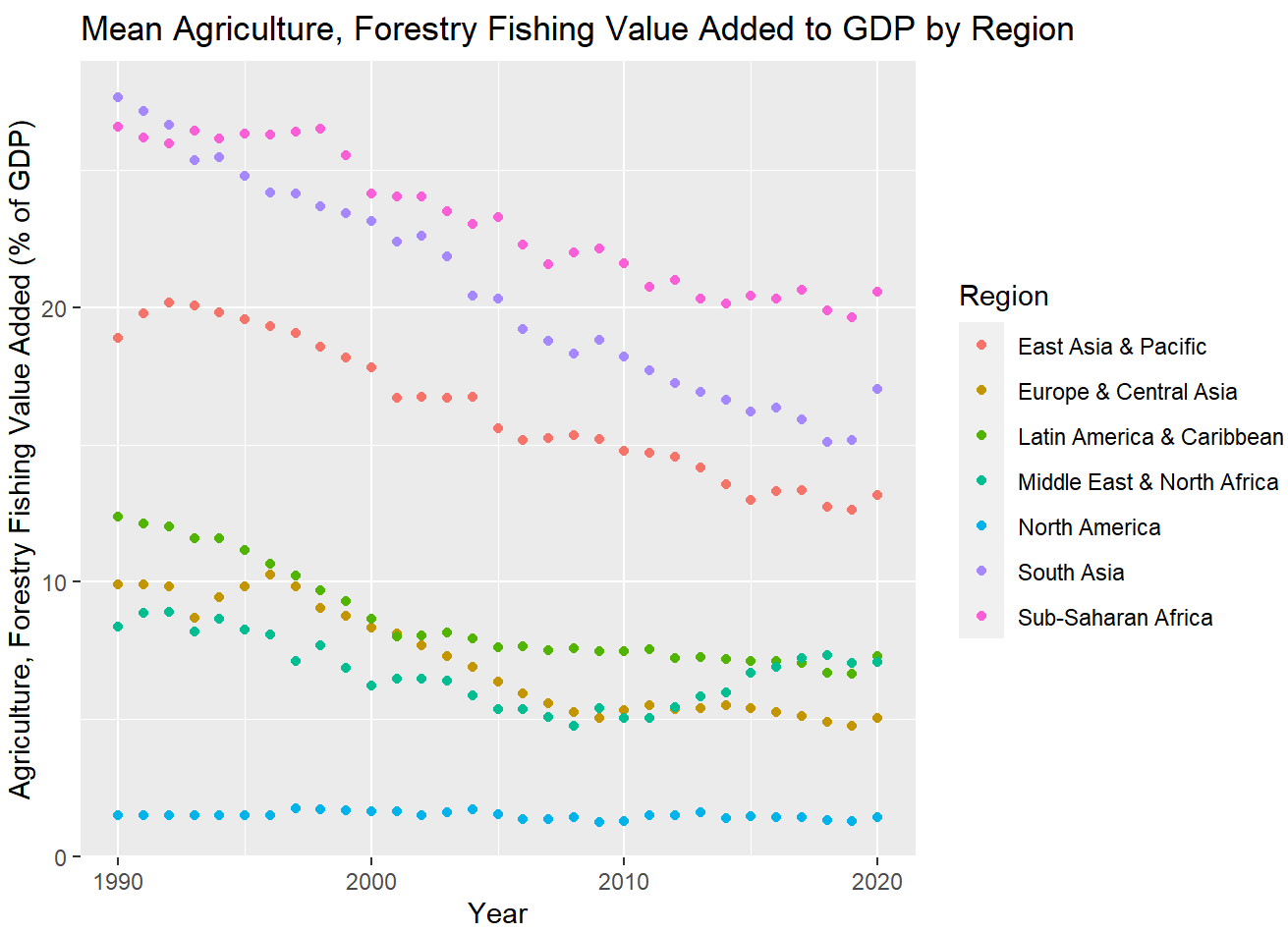

```
#look at co2 trends over time by region
co2_region <- na.omit(co2_data) %>% group_by(Region, Year) %>% summarize('Mean CO2 Emissions' = mean(`CO2
`), .groups = "keep")

#plot them
ggplot(co2_region, aes(x = Year, y = co2_region$`Mean CO2 Emissions`, color = Region, linetype = Region)) +
  geom_point(data = co2_region) +
  labs(title = "Mean CO2 Emissions per Capita by Region", x = "Year", y = "CO2 per Capita")
```




```
#overall global trend for agri GDP%
agri_global <- na.omit(agri_data) %>% group_by(Year) %>% summarize('Mean Agriculture, Forestry Fishing Valu
e Added to GDP' = mean(`Agriculture, Forestry, Fishing Value Added (% of GDP)`))

ggplot(agri_global, aes(x = Year, y = `Mean Agriculture, Forestry Fishing Value Added to GDP`)) +
  geom_point(data = agri_global) +
  labs(title = "Mean Agriculture, Forestry Fishing Value Added to GDP Globally", x = "Year", y = "Agricultu
re, Forestry Fishing Value Added (% of GDP)")
```

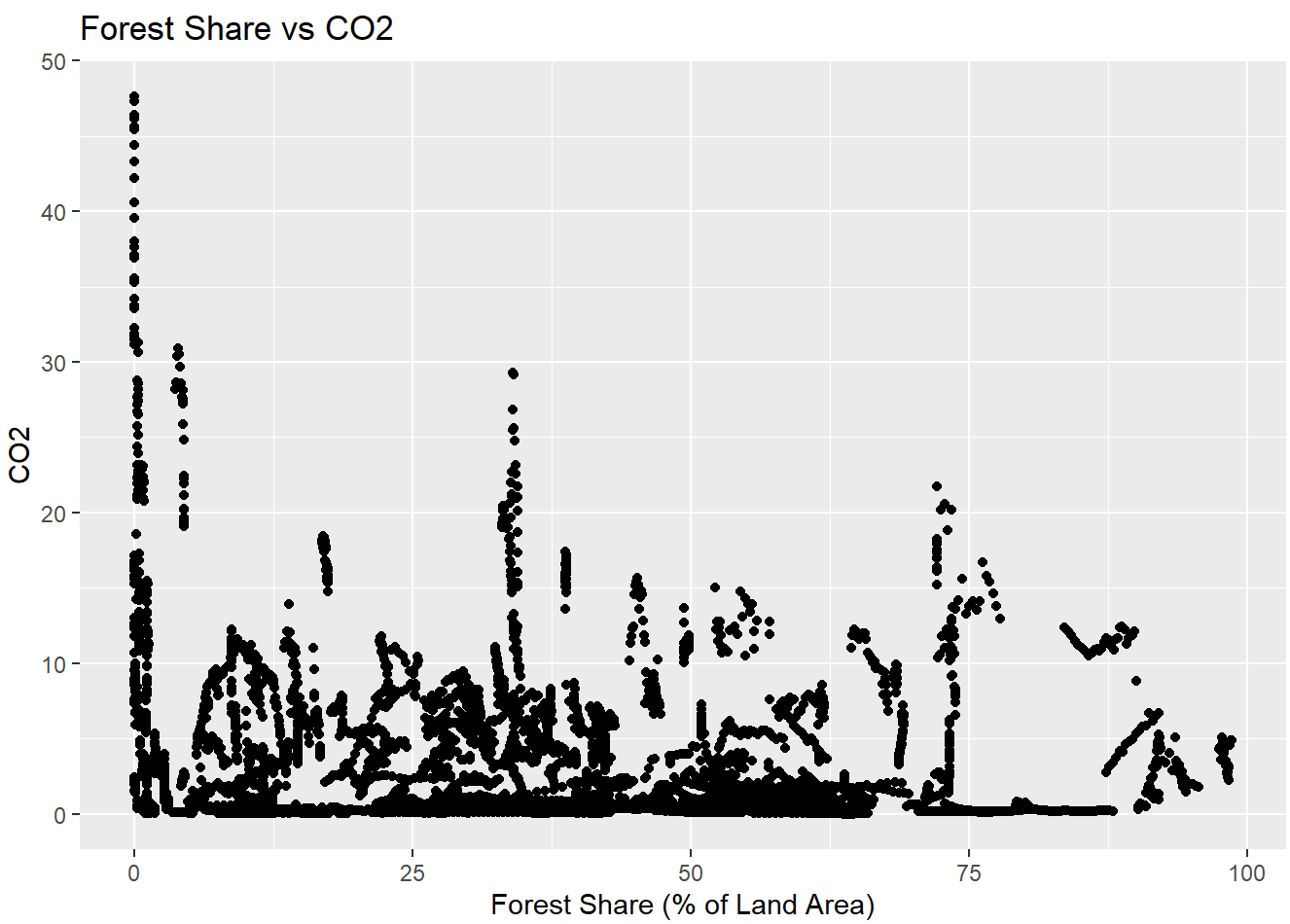



```
#look at agriculture/forestry/fishing value added trends over time by region
agri_region <- na.omit(agri_data) %>% group_by(Region, Year) %>% summarize('Mean Agriculture, Forestry Fish
ing Value Added to GDP' = mean(`Agriculture, Forestry, Fishing Value Added (% of GDP)`), .groups = "keep")

ggplot(agri_region, aes(x = Year, y = `Mean Agriculture, Forestry Fishing Value Added to GDP`, color = Regi
on, linetype = Region)) +
  geom_point(data = agri_region) +
  labs(title = "Mean Agriculture, Forestry Fishing Value Added to GDP by Region", x = "Year", y = "Agricult
ure, Forestry Fishing Value Added (% of GDP)")
```



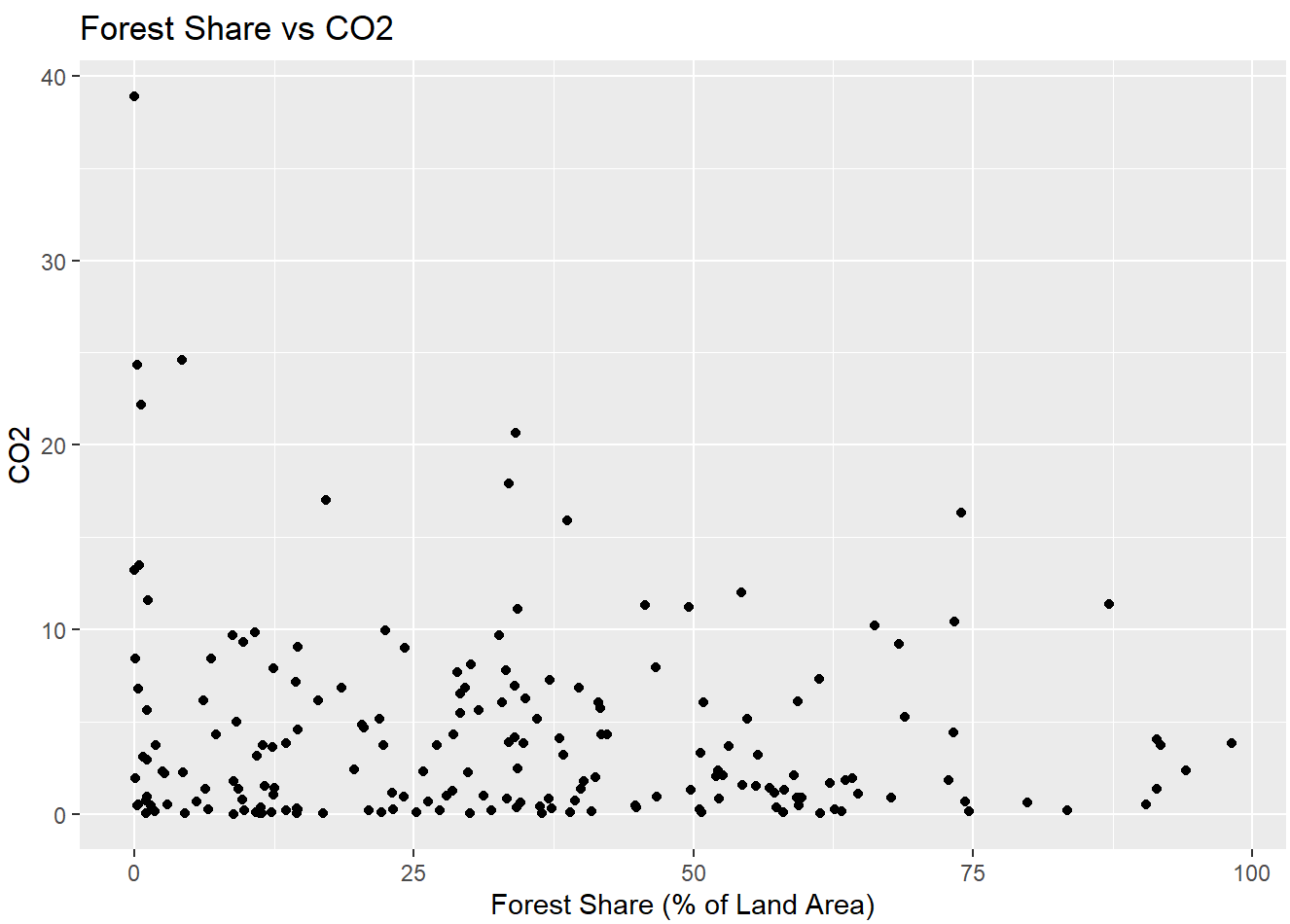
Linear Regression

```
#scatterplot of forest share vs co2 - ungrouped
ggplot(data, aes(x = `Forest Share`, y = `CO2`)) +
  geom_point(data = data) +
  labs(title = "Forest Share vs CO2", x = "Forest Share (% of Land Area)", y = "CO2")
```

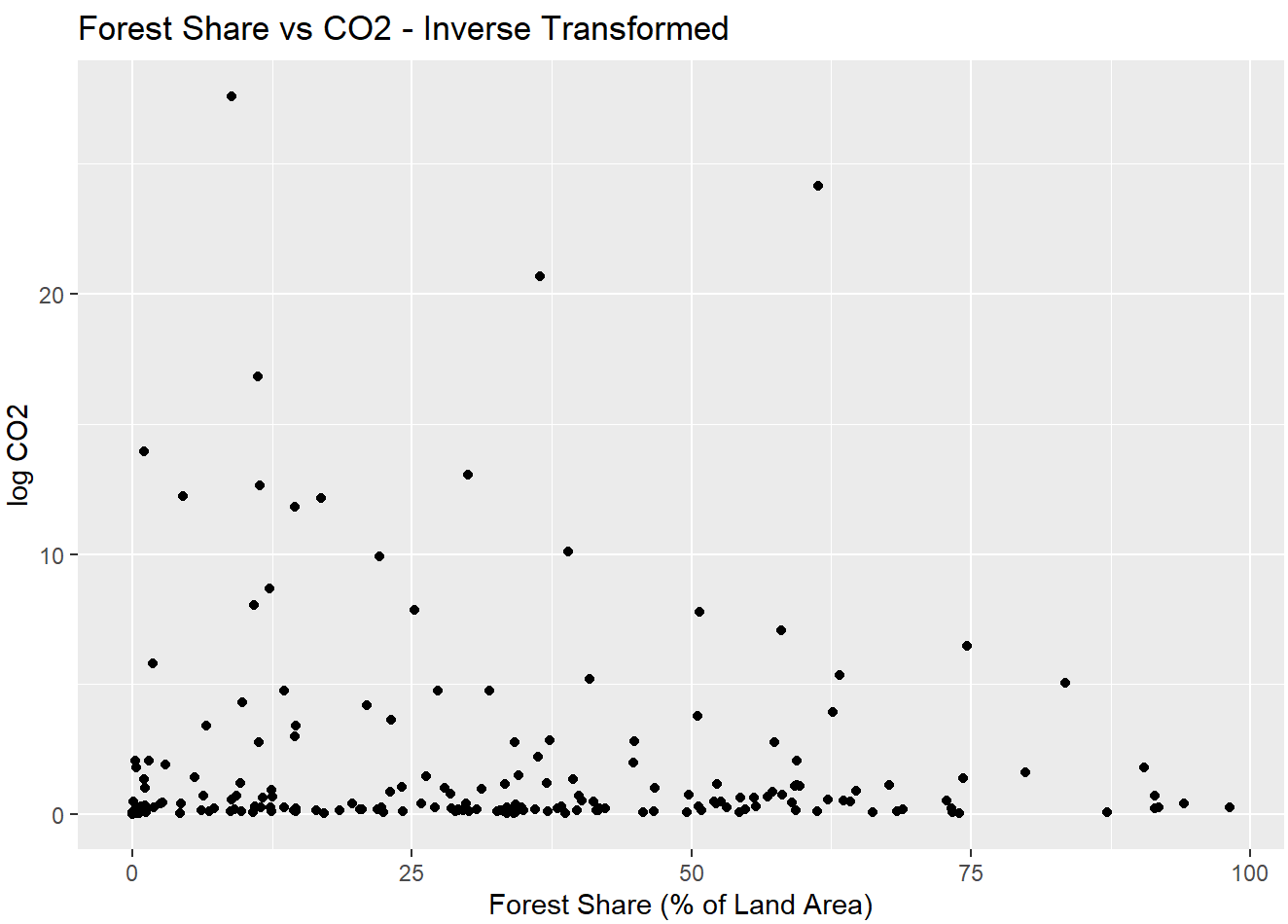



```
#this kinda doesn't make sense.. the clusters are likely countries
```

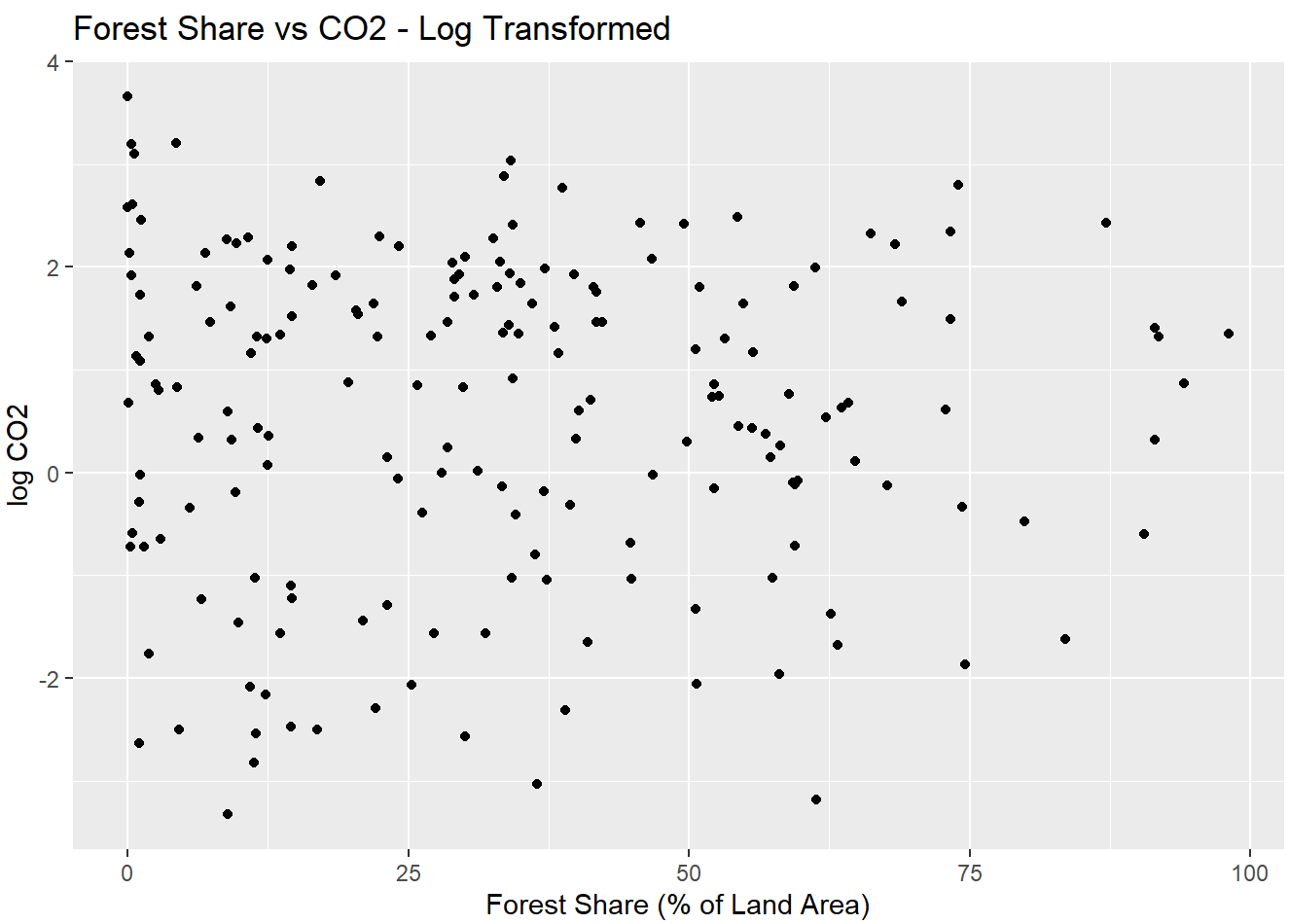
```
#scatterplot of forest share vs co2 but using the mean of the forest shares over all years
forest_country <- data %>% group_by(Country) %>% summarize('Mean CO2 Emissions' = mean(`CO2`), 'Mean Forest
Share' = mean(`Forest Share`), .groups="keep")

ggplot(forest_country, aes(x = `Mean Forest Share`, y = `Mean CO2 Emissions`)) +
  geom_point(data = forest_country) +
  labs(title = "Forest Share vs CO2", x = "Forest Share (% of Land Area)", y = "CO2")
```



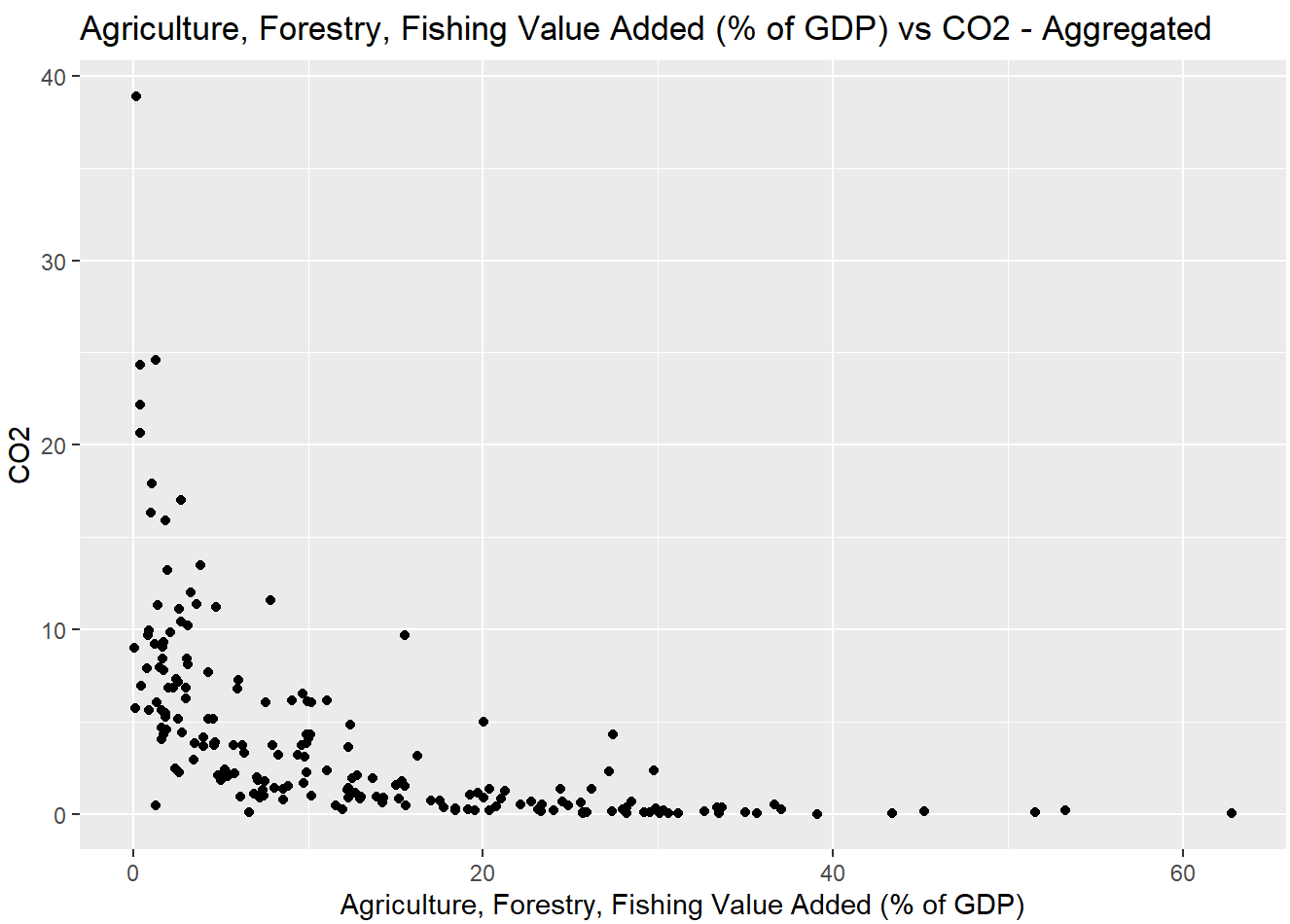

```
ggplot(forest_country, aes(x = `Mean Forest Share`, y = 1/(`Mean CO2 Emissions`))) +
  geom_point(data = forest_country) +
  labs(title = "Forest Share vs CO2 - Inverse Transformed", x = "Forest Share (% of Land Area)", y = "log C
O2")
```



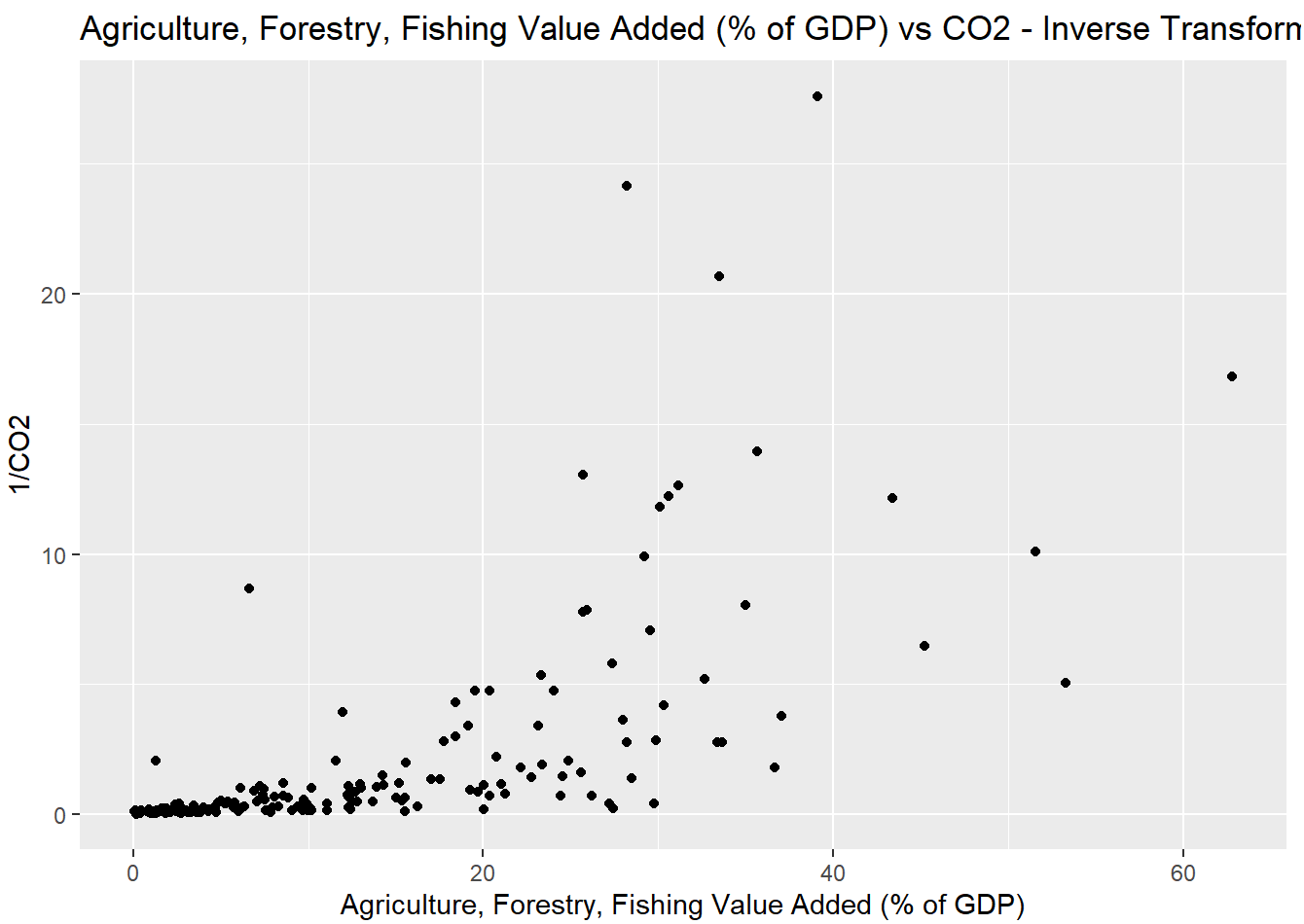

```
ggplot(forest_country, aes(x = `Mean Forest Share`, y = log(`Mean CO2 Emissions`))) +
  geom_point(data = forest_country) +
  labs(title = "Forest Share vs CO2 - Log Transformed", x = "Forest Share (% of Land Area)", y = "log CO2")
```



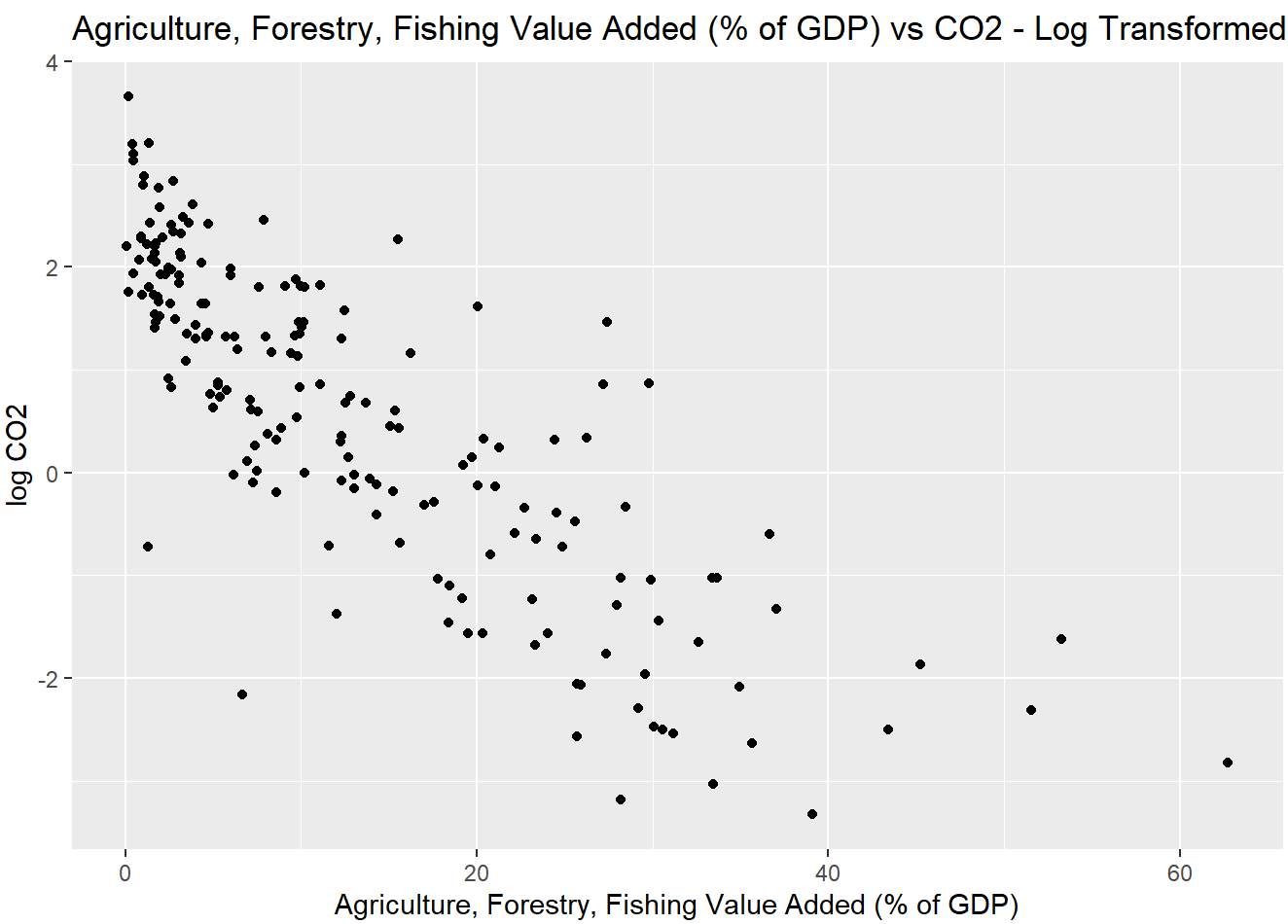

```
#scatterplot of agri gdp% vs co2 but using the mean over all years
agri_country <- data %>% group_by(Country) %>% summarize('Mean CO2 Emissions' = mean(`CO2`), 'Mean Agricult
ure, Forestry, Fishing Value Added (% of GDP)' = mean(`Agriculture, Forestry, Fishing Value Added (% of GD
P)`), .groups="keep")

ggplot(agri_country, aes(x = `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, y = `Mean CO2 Em
issions`)) +
  geom_point(data = agri_country) +
  labs(title = "Agriculture, Forestry, Fishing Value Added (% of GDP) vs CO2 - Aggregated", x = "Agricultur
e, Forestry, Fishing Value Added (% of GDP)", y = "CO2")
```



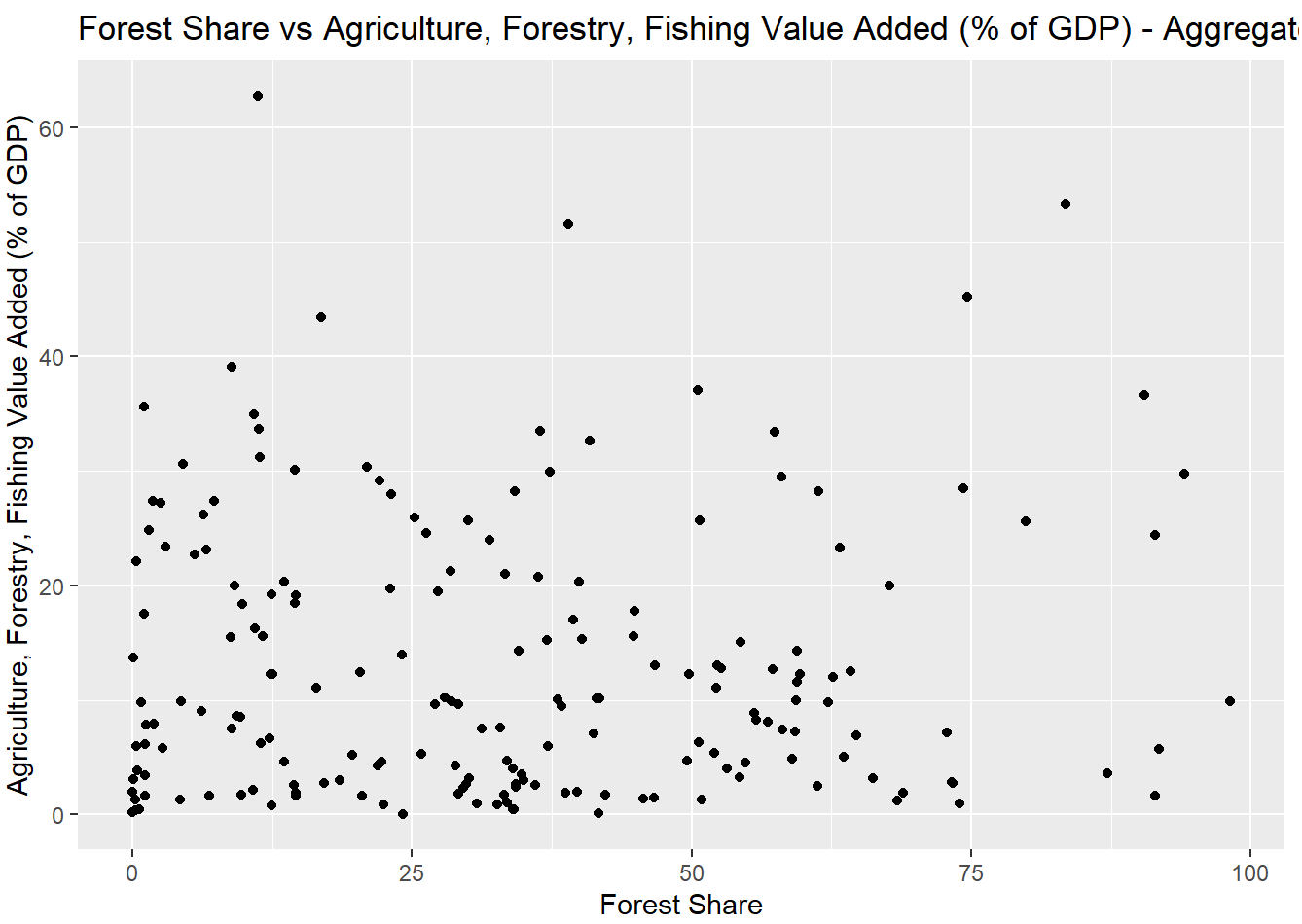

```
ggplot(agri_country, aes(x = `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, y = 1/`Mean CO2 Emissions`)) +
  geom_point(data = agri_country) +
  labs(title = "Agriculture, Forestry, Fishing Value Added (% of GDP) vs CO2 - Inverse Transformed", x = "Agriculture, Forestry, Fishing Value Added (% of GDP)", y = "1/CO2")
```




```
ggplot(agri_country, aes(x = `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, y = log(`Mean CO
2 Emissions`))) +
  geom_point(data = agri_country) +
  labs(title = "Agriculture, Forestry, Fishing Value Added (% of GDP) vs CO2 - Log Transformed", x = "Agric
ulture, Forestry, Fishing Value Added (% of GDP)", y = "log CO2")
```




```
#scatterplot of forest share vs agri gdp % using the mean of the forest shares over all years
agri_forest <- data %>% group_by(Country) %>% summarize('Mean Agriculture, Forestry, Fishing Value Added (% of GDP)' = mean(`Agriculture, Forestry, Fishing Value Added (% of GDP)`), 'Mean Forest Share' = mean(`Forest Share`), .groups="keep")

ggplot(agri_forest, aes(x = `Mean Forest Share`, y = `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`)) +
  geom_point(data = agri_forest) +
  labs(title = "Forest Share vs Agriculture, Forestry, Fishing Value Added (% of GDP) - Aggregated", x = "Forest Share", y = "Agriculture, Forestry, Fishing Value Added (% of GDP)")
```



Inverse Transformation Linear Regression

```
# transform CO2
data_transformed <- data_country
data_transformed$`Mean CO2 Emissions` <- 1/data_country$`Mean CO2 Emissions`
```

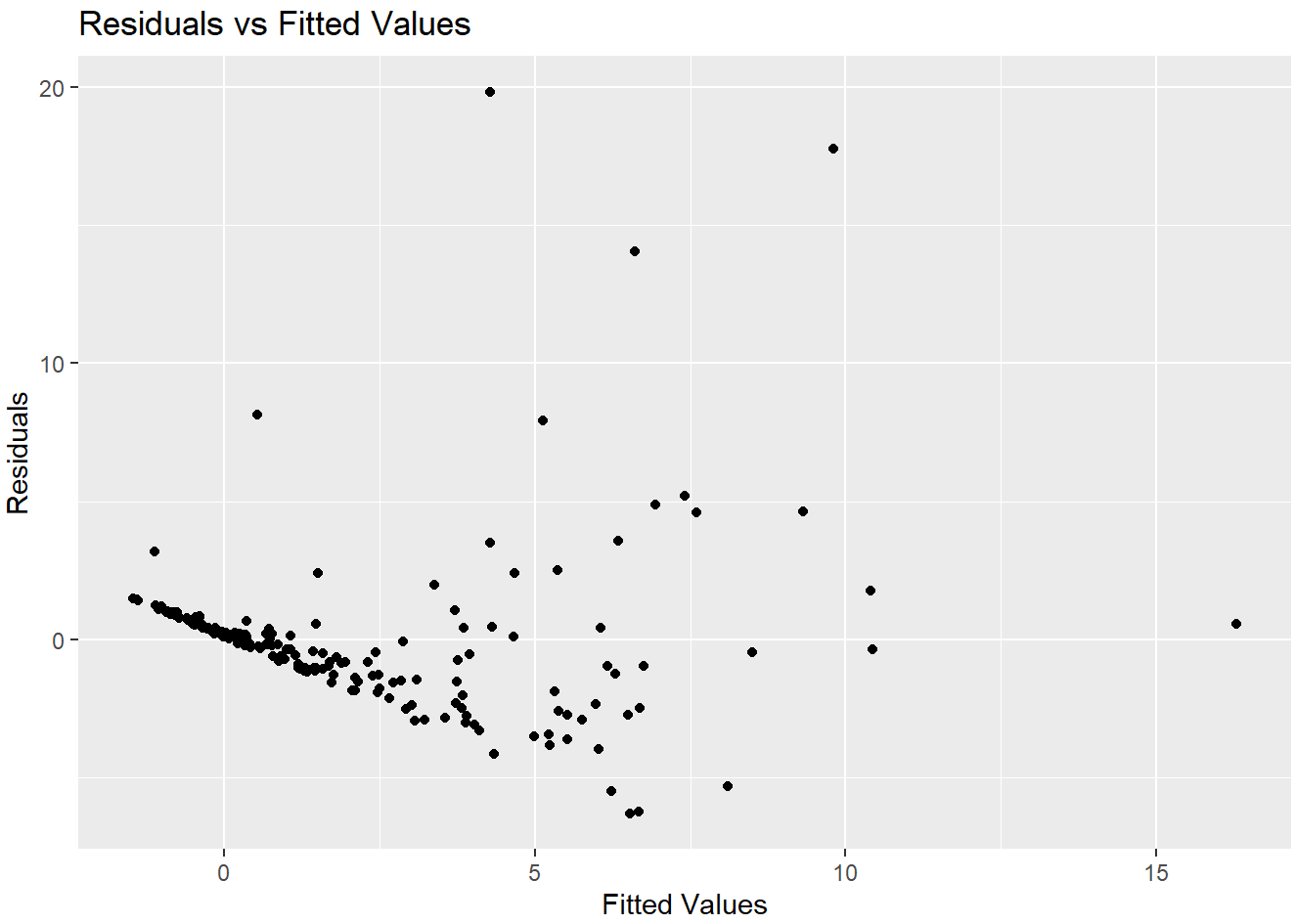
```
#fit a multiple regression model with interaction, aggregated by country over all 30 years (can't break out by countries)
mod <- lm(`Mean CO2 Emissions` ~ `Mean Forest Share`*`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, data = data_transformed)
summary(mod)
```

```
##
## Call:
## lm(formula = `Mean CO2 Emissions` ~ `Mean Forest Share` * `Mean Agriculture, Forestry, Fishing Value Add
ed (% of GDP)`,
##     data = data_transformed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.2986 -1.1403  0.0131  0.6598 19.8447
##
## Coefficients:
##                                                                                 Estimate
## (Intercept)                                                                    -1.5203050
## `Mean Forest Share`                                                             0.0164943
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                    0.3056336
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)` -0.0022125
##                                                                                Std. Error
## (Intercept)                                                                     0.5489996
## `Mean Forest Share`                                                             0.0132420
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                    0.0294071
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`  0.0006657
##                                                                                t value
## (Intercept)                                                                     -2.769
## `Mean Forest Share`                                                              1.246
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                    10.393
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`  -3.324
##                                                                                Pr(>|t|)
## (Intercept)                                                                     0.00619
## `Mean Forest Share`                                                             0.21448
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                    < 2e-16
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`  0.00107
##
## (Intercept)                                                                    **
## `Mean Forest Share`
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                   ***
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)` **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.989 on 185 degrees of freedom
## Multiple R-squared:  0.4821, Adjusted R-squared:  0.4737
## F-statistic:  57.4 on 3 and 185 DF,  p-value: < 2.2e-16
```

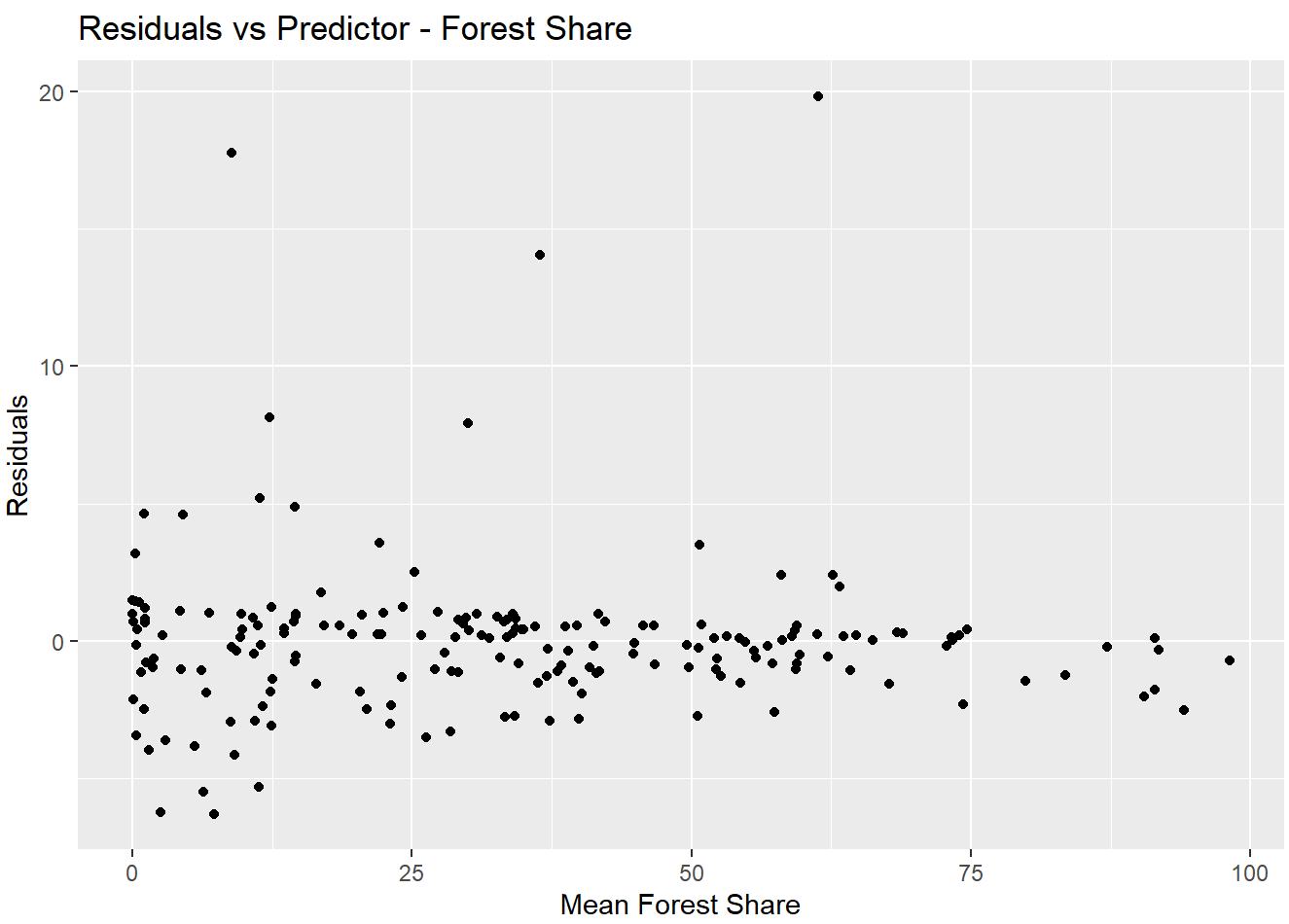
```
mod$coefficients
```

```
##                                                                  (Intercept)
##                                                                -1.520305005
##                                                           `Mean Forest Share`
##                                                                 0.016494339
##                     `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                                 0.305633611
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                                -0.002212485
```

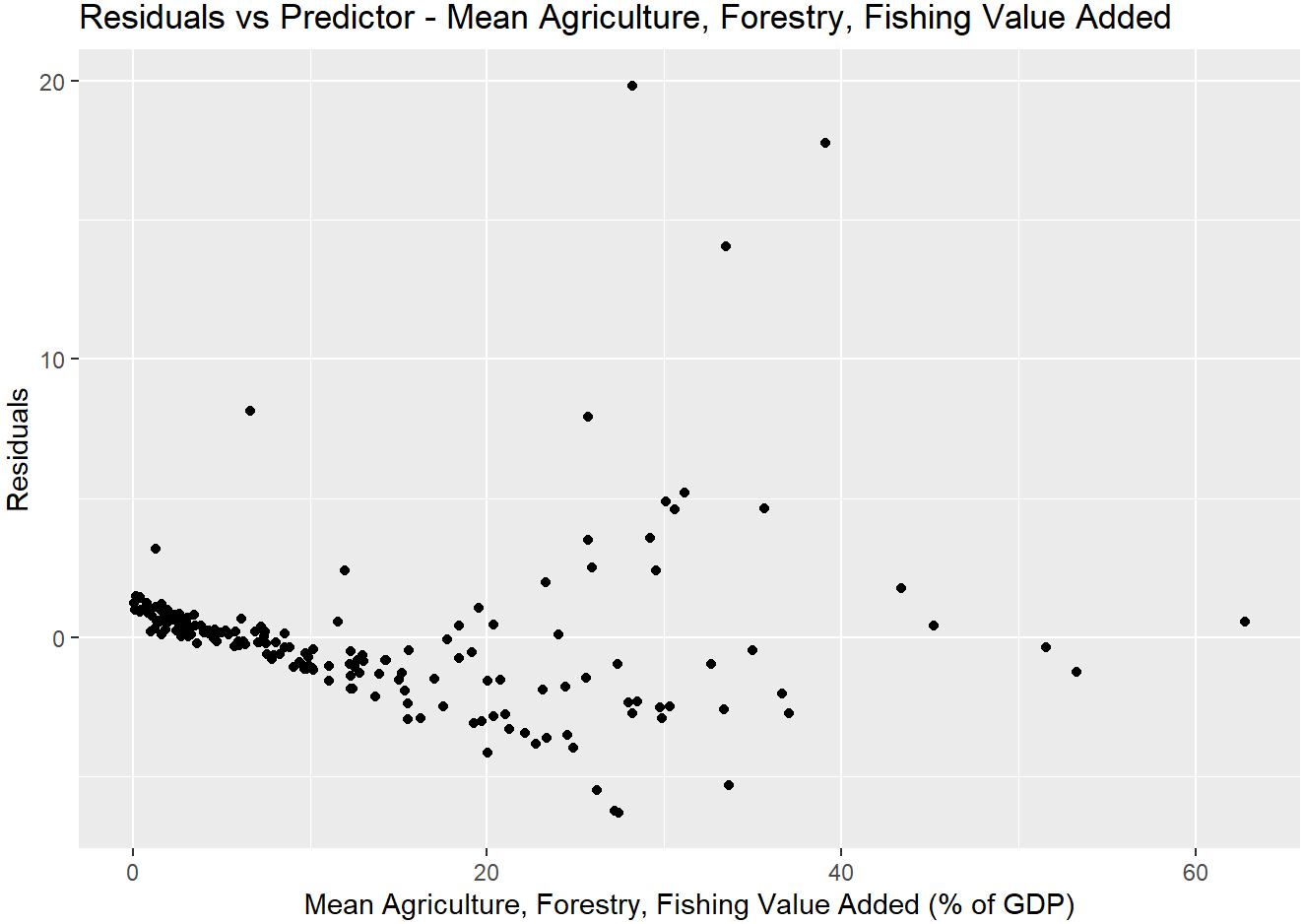
```
#check assumptions for aggregated model
#residuals vs fitted values
ggplot(data_transformed, aes(x = mod$fitted.values, y = mod$residuals)) +
  geom_point(data = data_transformed) +
  labs(title = "Residuals vs Fitted Values", x = "Fitted Values", y = "Residuals")
```



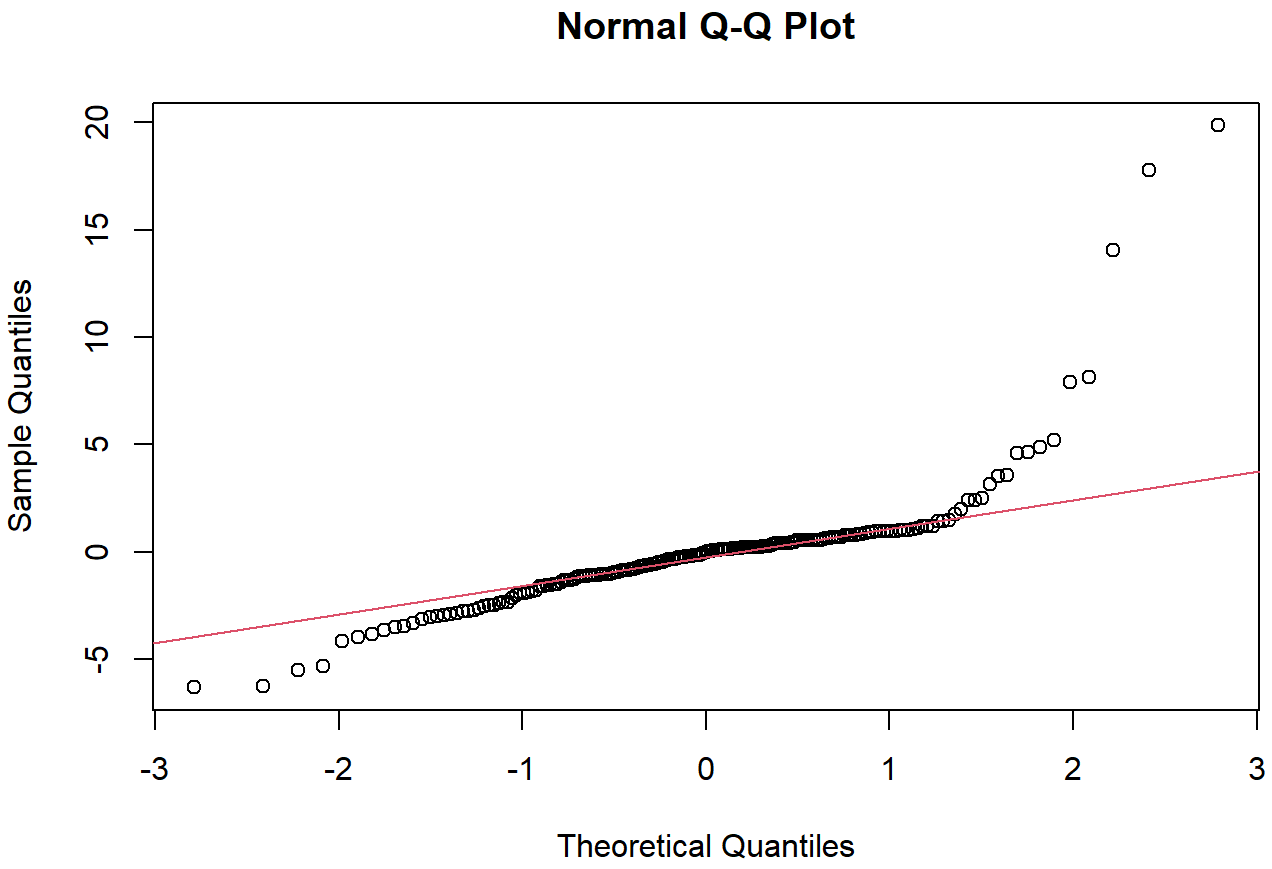

```
#residuals vs predictor
ggplot(data_transformed, aes(x = `Mean Forest Share`, y = mod$residuals)) +
  geom_point(data = data_transformed) +
  labs(title = "Residuals vs Predictor - Forest Share", x = "Mean Forest Share", y = "Residuals")
```




```
#residuals vs predictor
ggplot(data_transformed, aes(x = `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, y = mod$resi
duals)) +
  geom_point(data = data_transformed) +
  labs(title = "Residuals vs Predictor - Mean Agriculture, Forestry, Fishing Value Added", x = "Mean Agricu
lture, Forestry, Fishing Value Added (% of GDP)", y = "Residuals")
```




```
#qq plot
residuals <- residuals(mod)
qqnorm(residuals)
qqline(residuals, col = 2)
```



Log Transformation Linear Regression

```
# transform CO2
data_log <- data_country
data_log$`Mean CO2 Emissions` <- log(data_country$`Mean CO2 Emissions`)
```

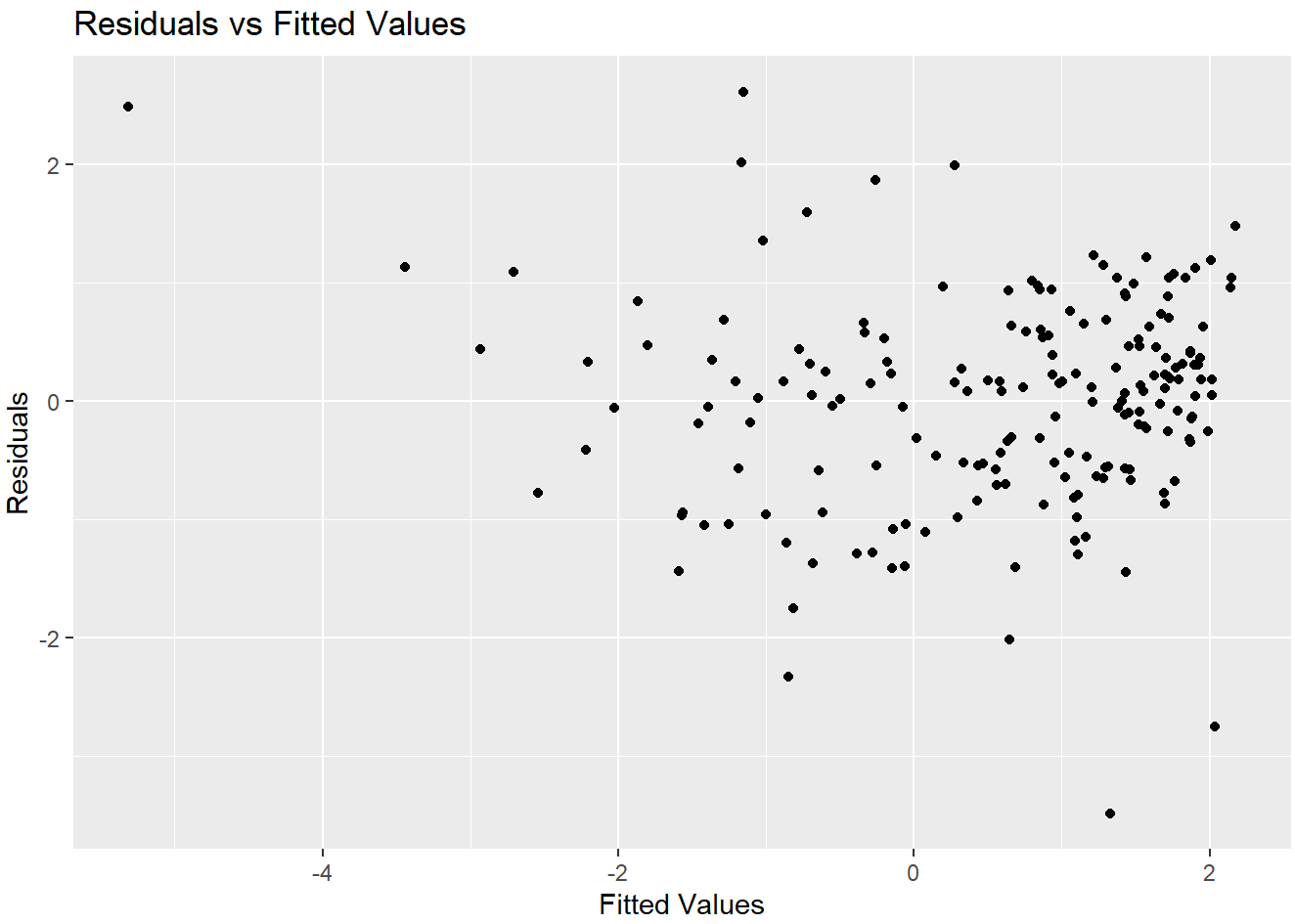
```
#fit a multiple regression model with interaction, aggregated by country over all 30 years (can't break out by countries)
mod_log <- lm(`Mean CO2 Emissions` ~ `Mean Forest Share`*`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, data = data_log)
summary(mod_log)
```

```
##
## Call:
## lm(formula = `Mean CO2 Emissions` ~ `Mean Forest Share` * `Mean Agriculture, Forestry, Fishing Value Add
ed (% of GDP)`,
##     data = data_log)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4863 -0.5645  0.0862  0.5291  2.6178
##
## Coefficients:
##                                                                                 Estimate
## (Intercept)                                                                    2.2001219
## `Mean Forest Share`                                                           -0.0072805
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                  -0.1242930
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`  0.0005214
##                                                                               Std. Error
## (Intercept)                                                                    0.1630675
## `Mean Forest Share`                                                            0.0039332
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                   0.0087347
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`  0.0001977
##                                                                               t value
## (Intercept)                                                                    13.492
## `Mean Forest Share`                                                            -1.851
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                  -14.230
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`   2.637
##                                                                               Pr(>|t|)
## (Intercept)                                                                    < 2e-16
## `Mean Forest Share`                                                            0.06576
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                   < 2e-16
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`  0.00907
##
## (Intercept)                                                                   ***
## `Mean Forest Share`                                                           .
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`                  ***
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)` **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8877 on 185 degrees of freedom
## Multiple R-squared:  0.6807, Adjusted R-squared:  0.6755
## F-statistic: 131.5 on 3 and 185 DF,  p-value: < 2.2e-16
```

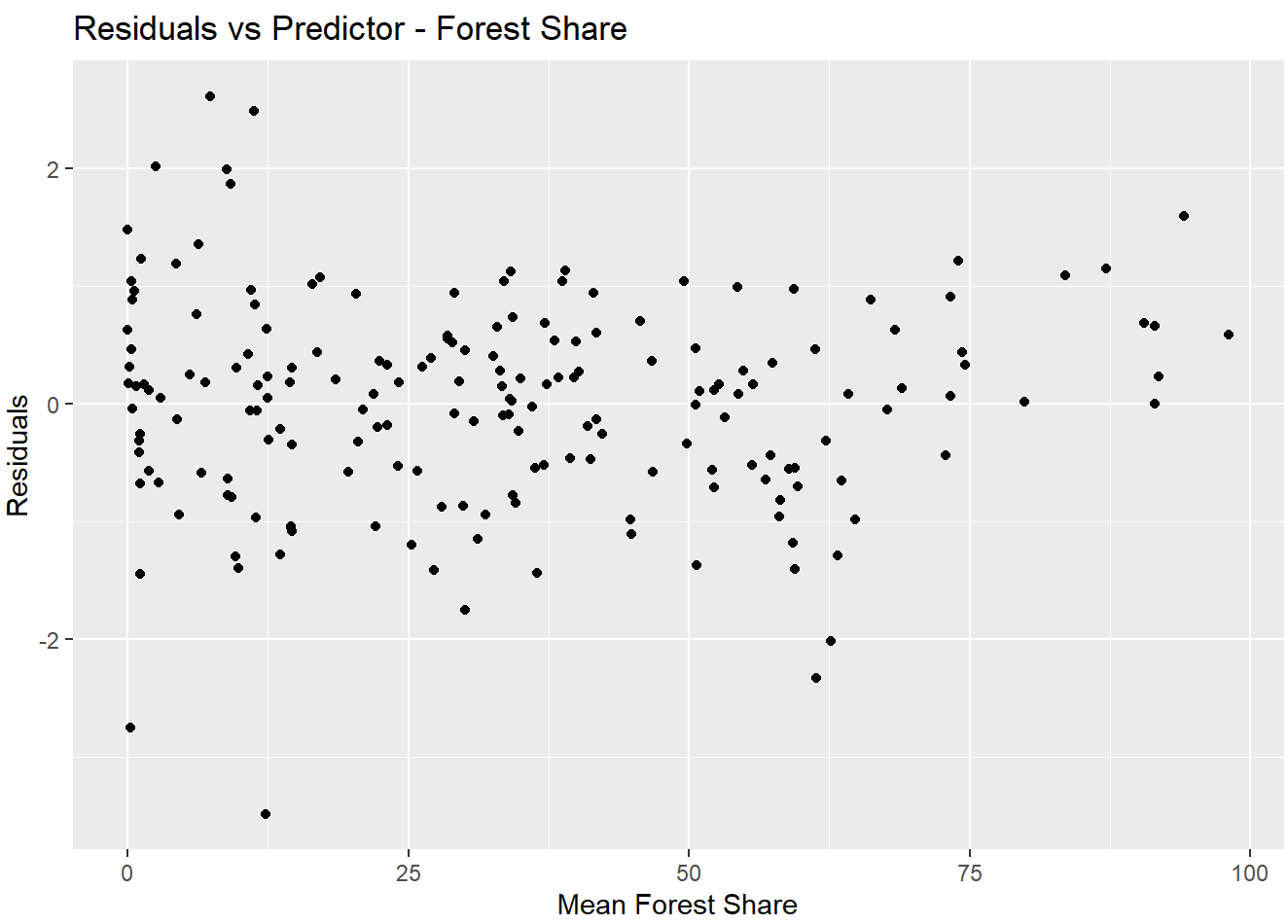
```
mod_log$coefficients
```

```
##                                                                 (Intercept)
##                                                               2.2001218715
##                                                          `Mean Forest Share`
##                                                               -0.0072804763
##                     `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                              -0.1242929910
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                               0.0005214394
```

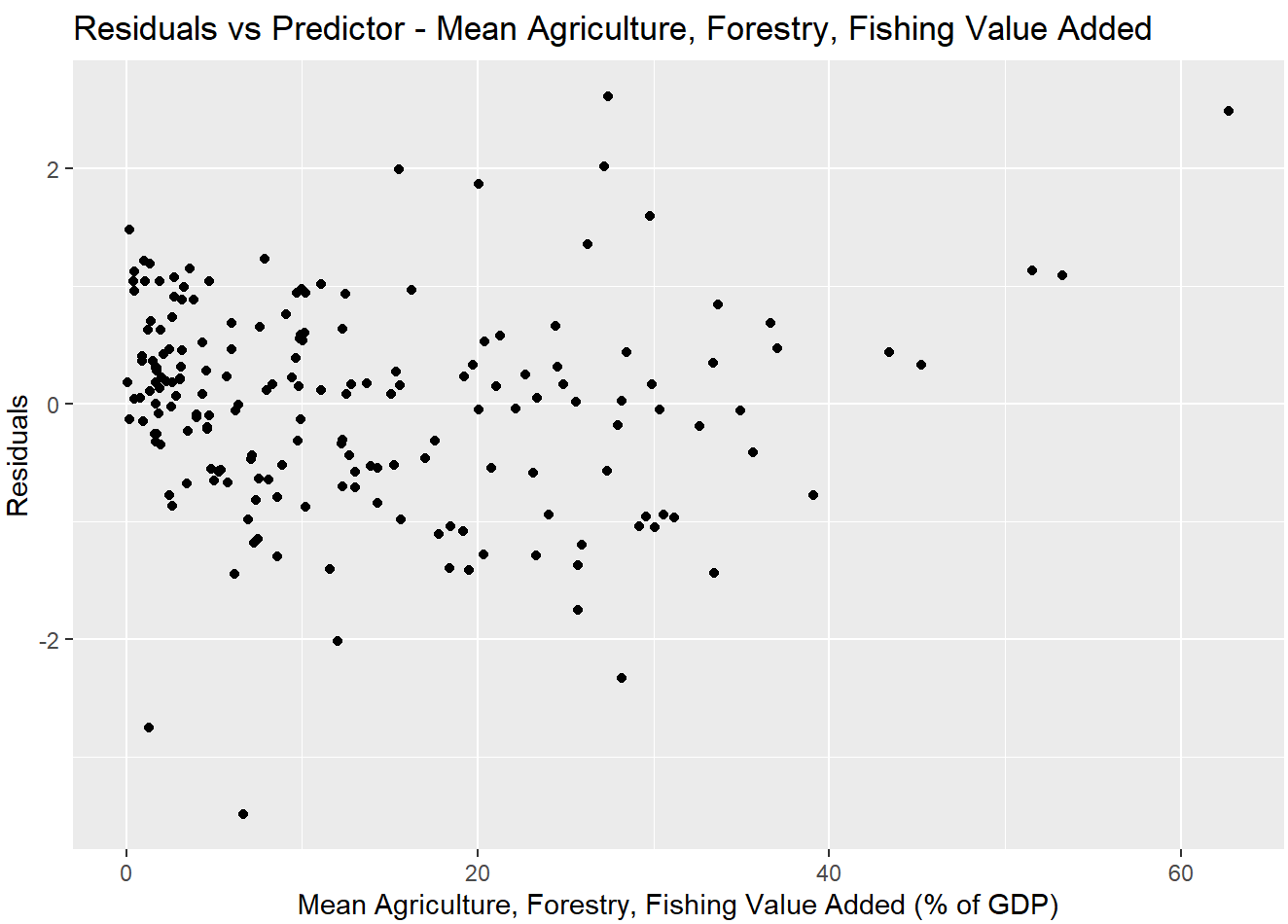
```
#check assumptions for aggregated model
#residuals vs fitted values
ggplot(data_log, aes(x = mod_log$fitted.values, y = mod_log$residuals)) +
  geom_point(data = data_log) +
  labs(title = "Residuals vs Fitted Values", x = "Fitted Values", y = "Residuals")
```



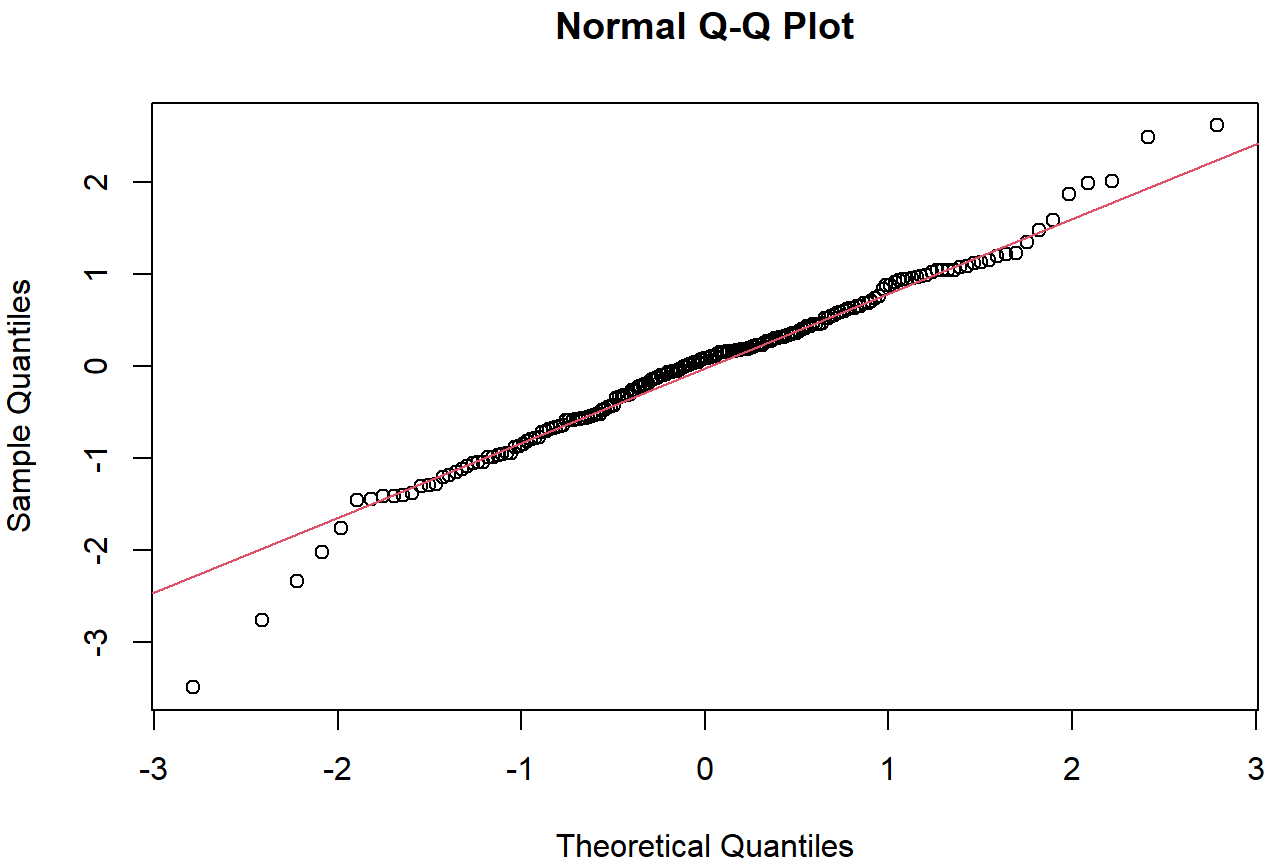

```
#residuals vs predictor
ggplot(data_log, aes(x = `Mean Forest Share`, y = mod_log$residuals)) +
  geom_point(data = data_log) +
  labs(title = "Residuals vs Predictor - Forest Share", x = "Mean Forest Share", y = "Residuals")
```




```
#residuals vs predictor
ggplot(data_log, aes(x = `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, y = mod_log$residuals)) +
  geom_point(data = data_log) +
  labs(title = "Residuals vs Predictor - Mean Agriculture, Forestry, Fishing Value Added", x = "Mean Agriculture, Forestry, Fishing Value Added (% of GDP)", y = "Residuals")
```




```
#qq plot
residuals <- residuals(mod_log)
qqnorm(residuals)
qqline(residuals, col = 2)
```




```
#jackknife estimation of se, 95% confidence intervals
n <- nrow(data_log)
beta0jack <- rep(0, n)
beta1jack <- rep(0, n)
beta2jack <- rep(0, n)
beta3jack <- rep(0, n)

for (i in 1:n){
  lmstar <- lm(`Mean CO2 Emissions` ~ `Mean Forest Share`*`Mean Agriculture, Forestry, Fishing Value Added
(% of GDP)`, data = data_log, subset = -i)
  beta0jack[i] <- lmstar$coefficients[1]
  beta1jack[i] <- lmstar$coefficients[2]
  beta2jack[i] <- lmstar$coefficients[3]
  beta3jack[i] <- lmstar$coefficients[4]
}

varhat0 <- (n-1)/n*sum((beta0jack-mean(beta0jack))^2)
sehat0 <- sqrt(varhat0)
varhat1 <- (n-1)/n*sum((beta1jack-mean(beta1jack))^2)
sehat1 <- sqrt(varhat1)
varhat2 <- (n-1)/n*sum((beta2jack-mean(beta2jack))^2)
sehat2 <- sqrt(varhat2)
varhat3 <- (n-1)/n*sum((beta3jack-mean(beta3jack))^2)
sehat3 <- sqrt(varhat3)

#create confidence intervals for each coefficient
beta0ci <- c(mod_log$coefficients[1]-1.96*sehat0, mod_log$coefficients[1]+1.96*sehat0)
beta0ci
```

```
## (Intercept) (Intercept)
##    1.784031    2.616213
```

```
beta1ci <- c(mod_log$coefficients[2]-1.96*sehat1, mod_log$coefficients[2]+1.96*sehat1)
beta1ci
```

```
## `Mean Forest Share` `Mean Forest Share`
##        -0.016080357         0.001519404
```

```
beta2ci <- c(mod_log$coefficients[3]-1.96*sehat2, mod_log$coefficients[3]+1.96*sehat2)
beta2ci
```

```
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                   -0.15281055
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                   -0.09577544
```

```
beta3ci <- c(mod_log$coefficients[4]-1.96*sehat3, mod_log$coefficients[4]+1.96*sehat3)
beta3ci
```

```
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                                    -3.718962e-05
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                                     1.080069e-03
```

```
#p-values with jackknife estimates of se
p <- function(beta, se, xnum, n) {
  t_stat <- beta/se
  df <- n-(xnum+1)
  p_value <- 2 * pt(abs(t_stat), df, lower.tail = FALSE)
  p_value
  return(p_value)
}

beta0p <- p(mod_log$coefficients[1], sehat0, 4, nrow(data_log))
beta0p
```

```
##  (Intercept)
## 4.084218e-20
```

```
beta1p <- p(mod_log$coefficients[2], sehat1, 4, nrow(data_log))
beta1p
```

```
## `Mean Forest Share`
##            0.106605
```

```
beta2p <- p(mod_log$coefficients[3], sehat2, 4, nrow(data_log))
beta2p
```

```
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                4.894391e-15
```

```
beta3p <- p(mod_log$coefficients[4], sehat3, 4, nrow(data_log))
beta3p
```

```
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                                      0.06894064
```

```
#transform coefficients back to original scale as percent change
b0 <- exp(mod_log$coefficients[1])
(b0-1)*100
```

```
## (Intercept)
##    802.6113
```

```
b1 <- exp(mod_log$coefficients[2])
(b1-1)*100
```

```
## `Mean Forest Share`
##          -0.7254038
```

```
b2 <- exp(mod_log$coefficients[3])
(b2-1)*100
```

```
## `Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                  -11.68789
```

```
b3 <- exp(mod_log$coefficients[4])
(b3-1)*100
```

```
## `Mean Forest Share`:`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`
##                                                                      0.05215754
```

```
#calculate correlation between the 2 x variables for collinearity investigation
cor(data_transformed$`Mean Agriculture, Forestry, Fishing Value Added (% of GDP)`, data_transformed$`Mean F
orest Share`)
```

```
## [1] 0.01818023
```

analysis

2024-02-24

Data

```
data <- read.csv("data/clean_data.csv", header=T)
header <- colnames(data)
```

Column names

```
energy_prod_cols <- c(
  "Electricity.production.from.coal.sources....of.total.",
  "Electricity.production.from.hydroelectric.sources....of.total.",
  "Electricity.production.from.natural.gas.sources....of.total.",
  "Electricity.production.from.nuclear.sources....of.total.",
  "Electricity.production.from.oil.sources....of.total.",
  "Electricity.production.from.oil..gas.and.coal.sources....of.total.",
  "Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total."
)

energy_use_cols <- c(
  "Energy.use..kg.of.oil.equivalent.per.capita.",
  "Electric.power.consumption..kWh.per.capita.",
  "Fossil.fuel.energy.consumption....of.total.",
  "Combustible.renewables.and.waste....of.total.energy.",
  "Alternative.and.nuclear.energy....of.total.energy.use.",
  "Electric.power.transmission.and.distribution.losses....of.output."
)

response_col <- "CO2.emissions..metric.tons.per.capita."

other_cols <- c("Country", "Year")
```

Linear Regression: Energy Production vs CO2 emissions

```
prod_mod <- lm(
  CO2.emissions..metric.tons.per.capita. ~
  Electricity.production.from.coal.sources....of.total. +
  Electricity.production.from.hydroelectric.sources....of.total. +
  Electricity.production.from.natural.gas.sources....of.total. +
  Electricity.production.from.nuclear.sources....of.total. +
```

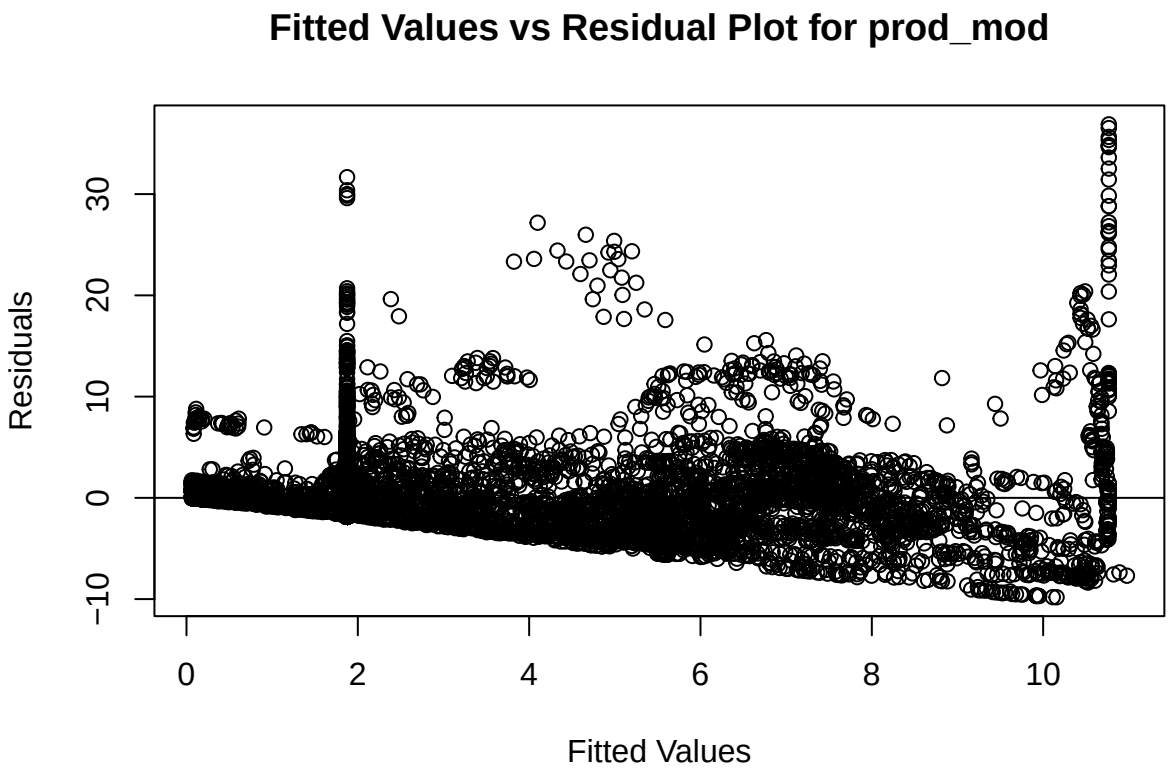
```
  Electricity.production.from.oil.sources....of.total. +
  Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total.,
  data = data
)

prod_summary <- summary(prod_mod)
```

Linear Regression: Energy Use vs CO2 emissions

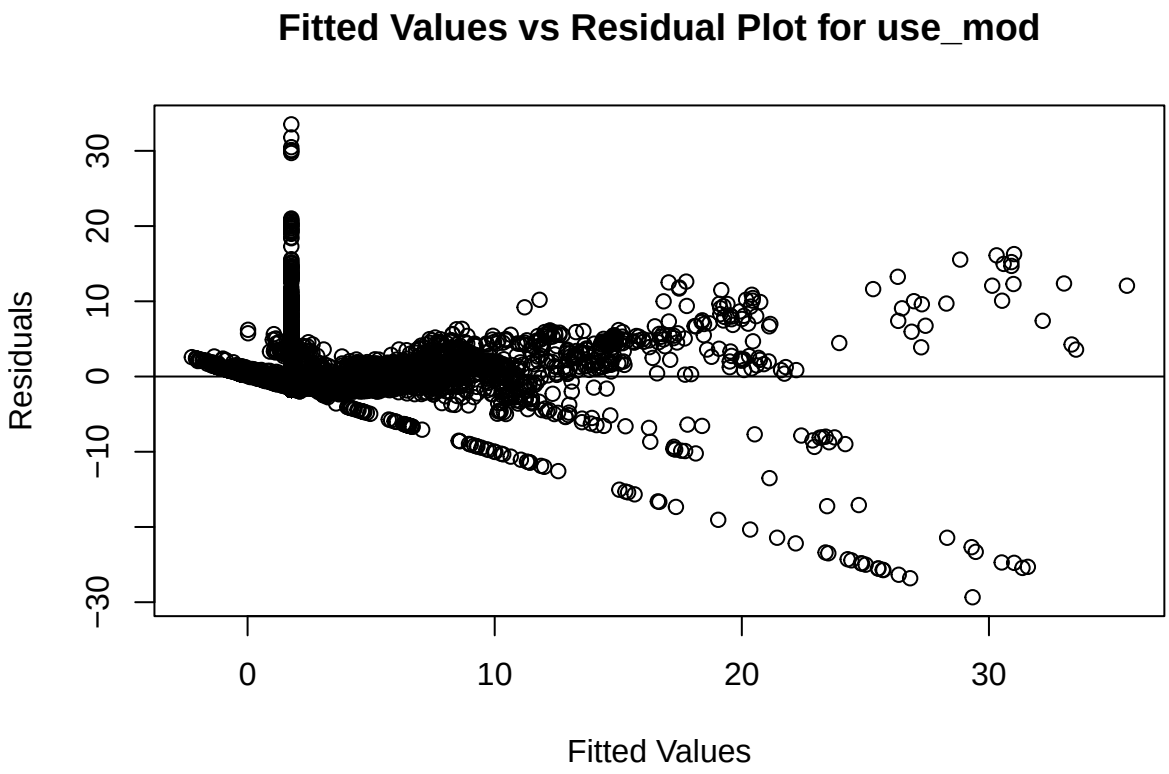
```
use_mod <- lm(
  CO2.emissions..metric.tons.per.capita. ~
  Energy.use..kg.of.oil.equivalent.per.capita. +
  Electric.power.consumption..kWh.per.capita. +
  Fossil.fuel.energy.consumption....of.total. +
  Combustible.renewables.and.waste....of.total.energy. +
  Alternative.and.nuclear.energy....of.total.energy.use. +
  Electric.power.transmission.and.distribution.losses....of.output.,
  data = data
)

use_summary <- summary(use_mod)
```

Constant Variance check

```
plot(prod_mod$fitted.values, residuals(prod_mod),
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Fitted Values vs Residual Plot for prod_mod")
abline(h = 0)
```




```
plot(use_mod$fitted.values, residuals(use_mod),
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Fitted Values vs Residual Plot for use_mod")
abline(h = 0)
```



Summary: Coefficient table

```
sum(data$Electricity.production.from.oil.sources....of.total.==0)/nrow(data)
```

```
## [1] 0.48838
```

```
sum(
  data$Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total.==0)/
  nrow(data)
```

```
## [1] 0.6349966
```

Robust standard eror

```
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
```

```
library("lmtest")
```

```
## Warning: package 'lmtest' was built under R version 4.3.3
```

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

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

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(sandwich)
```

```
## Warning: package 'sandwich' was built under R version 4.3.3
```

```
df <- nrow(data) - (6 + 1)
alpha <- 0.05

prod_CI_table <-  coeftest(prod_mod, vcov = vcovHC)[,1:2]
prod_CI_df <- as.data.frame(prod_CI_table)
colnames(prod_CI_df) <- c("Estimate", "Std_Error")
prod_CI_df <- prod_CI_df %>%
  mutate(lower_limit = Estimate - abs(qt(alpha/2, df))*Std_Error)
prod_CI_df <- prod_CI_df %>%
  mutate(upper_limit = Estimate + abs(qt(alpha/2, df))*Std_Error)

use_CI_table <-  coeftest(use_mod, vcov = vcovHC)[,1:2]
use_CI_df <- as.data.frame(use_CI_table)
colnames(use_CI_df) <- c("Estimate", "Std_Error")
use_CI_df <- use_CI_df %>%
  mutate(lower_limit = Estimate - abs(qt(alpha/2, df))*Std_Error)
use_CI_df <- use_CI_df %>%
  mutate(upper_limit = Estimate + abs(qt(alpha/2, df))*Std_Error)

prod_CI_df
```

```
##                                                                                 Estimate
## (Intercept)                                                                1.874604e+00
## Electricity.production.from.coal.sources....of.total.                      3.811635e-02
## Electricity.production.from.hydroelectric.sources....of.total.            -1.812128e-02
## Electricity.production.from.natural.gas.sources....of.total.               8.891818e-02
## Electricity.production.from.nuclear.sources....of.total.                   9.658797e-02
## Electricity.production.from.oil.sources....of.total.                      -9.120482e-05
## Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total.  6.273503e-02
##                                                                                Std_Error
```

```
## (Intercept)                                                                    0.052431922
## Electricity.production.from.coal.sources....of.total.                          0.002311526
## Electricity.production.from.hydroelectric.sources....of.total.                 0.001230192
## Electricity.production.from.natural.gas.sources....of.total.                   0.004088524
## Electricity.production.from.nuclear.sources....of.total.                       0.004213969
## Electricity.production.from.oil.sources....of.total.                           0.002212579
## Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total. 0.008023937
##                                                                                 lower_limit
## (Intercept)                                                                    1.771825076
## Electricity.production.from.coal.sources....of.total.                          0.033585219
## Electricity.production.from.hydroelectric.sources....of.total.                -0.020532747
## Electricity.production.from.natural.gas.sources....of.total.                   0.080903714
## Electricity.production.from.nuclear.sources....of.total.                       0.088327604
## Electricity.production.from.oil.sources....of.total.                          -0.004428378
## Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total.  0.047006234
##                                                                                 upper_limit
## (Intercept)                                                                    1.977382800
## Electricity.production.from.coal.sources....of.total.                          0.042647483
## Electricity.production.from.hydroelectric.sources....of.total.                -0.015709818
## Electricity.production.from.natural.gas.sources....of.total.                   0.096932644
## Electricity.production.from.nuclear.sources....of.total.                       0.104848340
## Electricity.production.from.oil.sources....of.total.                           0.004245969
## Electricity.production.from.renewable.sources..excluding.hydroelectric....of.total.  0.078463831
```

```
use_CI_df
```

```
##                                                                  Estimate
## (Intercept)                                                 1.7693209052
## Energy.use..kg.of.oil.equivalent.per.capita.                0.0014196787
## Electric.power.consumption..kWh.per.capita.                 0.0001267691
## Fossil.fuel.energy.consumption....of.total.                 0.0155192417
## Combustible.renewables.and.waste....of.total.energy.       -0.0323674417
## Alternative.and.nuclear.energy....of.total.energy.use.     -0.0879561788
## Electric.power.transmission.and.distribution.losses....of.output. -0.0223147682
##                                                                Std_Error
## (Intercept)                                                5.355689e-02
## Energy.use..kg.of.oil.equivalent.per.capita.               1.393371e-04
## Electric.power.consumption..kWh.per.capita.                8.584658e-05
## Fossil.fuel.energy.consumption....of.total.                2.559684e-03
## Combustible.renewables.and.waste....of.total.energy.       1.130487e-03
## Alternative.and.nuclear.energy....of.total.energy.use.     7.973223e-03
## Electric.power.transmission.and.distribution.losses....of.output. 4.172097e-03
##                                                              lower_limit
## (Intercept)                                                 1.6643368418
## Energy.use..kg.of.oil.equivalent.per.capita.                0.0011465452
## Electric.power.consumption..kWh.per.capita.                -0.0000415103
## Fossil.fuel.energy.consumption....of.total.                 0.0105016617
## Combustible.renewables.and.waste....of.total.energy.       -0.0345834604
## Alternative.and.nuclear.energy....of.total.energy.use.     -0.1035855655
## Electric.power.transmission.and.distribution.losses....of.output. -0.0304930556
##                                                              upper_limit
## (Intercept)                                                 1.8743049686
## Energy.use..kg.of.oil.equivalent.per.capita.                0.0016928122
## Electric.power.consumption..kWh.per.capita.                 0.0002950486
```

```
## Fossil.fuel.energy.consumption....of.total.                          0.0205368217
## Combustible.renewables.and.waste....of.total.energy.                -0.0301514230
## Alternative.and.nuclear.energy....of.total.energy.use.              -0.0723267921
## Electric.power.transmission.and.distribution.losses....of.output. -0.0141364807
```

In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [ ]:
```
df = pd.read_csv('data/data.csv')

# Rename columns
df = df.rename(columns=lambda x: x.split("[")[0])
df = df.rename(columns=lambda x: int(x.strip()) if x.strip().isdigit() else x)

# Convert missing values to NaN
df.replace('..', np.nan, inplace=True)
non_num_cols = ['Series Name', 'Series Code', 'Country Name', 'Country Code']
num_cols = [col for col in df.columns if col not in non_num_cols]
num_cols_1990 = [x for x in num_cols if x>=1990]
df[num_cols] = df[num_cols].astype(float)

# Impute missing values with 0s
df = df.apply(lambda row: row.fillna(0))

# Remove Indicator & Country Code
data = df.loc[:,[x for x in df.columns if 'Code' not in str(x)]]
```

In [ ]:
```
# Get data after 1990
numerical_data_1990 = data.iloc[:,2:].loc[:, data.iloc[:,2:].columns >= 1990]
data_1990 = pd.concat([data[['Series Name', 'Country Name']], numerical_data_1990], ax
```

Transpose data

In [ ]:
```
selected_cols = [     'Series Name',      1990,
                      1991,               1992,               1993,
                      1994,               1995,               1996,
                      1997,               1998,               1999,
                      2000,               2001,               2002,
                      2003,               2004,               2005,
                      2006,               2007,               2008,
                      2009,               2010,               2011,
                      2012,               2013,               2014,
                      2015,               2016,               2017,
                      2018,               2019,               2020,
                      2021,               2022, ]
```

In [ ]:
```
ordered_cols = ['Country', 'Year',
                'Electricity production from coal sources (% of total)',
                'Electricity production from hydroelectric sources (% of total)',
                'Electricity production from natural gas sources (% of total)',
                'Electricity production from nuclear sources (% of total)',
                'Electricity production from oil sources (% of total)',
                'Electricity production from oil, gas and coal sources (% of total)',
                'Electricity production from renewable sources, excluding hydroelectri
                'Energy use (kg of oil equivalent per capita)',
                'Electric power consumption (kWh per capita)',
                'Fossil fuel energy consumption (% of total)',
                'Combustible renewables and waste (% of total energy)',
                'Alternative and nuclear energy (% of total energy use)',
```

```
                'Electric power transmission and distribution losses (% of output)',
                'CO2 emissions (metric tons per capita)'
               ]
```

In [ ]:
```
final_df = pd.DataFrame(columns=ordered_cols)

countries = data_1990['Country Name'].unique()
for country in countries:
    country_data = data_1990.loc[data_1990['Country Name']==country,selected_cols].set
    cntry_df = country_data.T
    cntry_df['Country'] = country
    cntry_df['Year'] = cntry_df.index
    cntry_df = cntry_df[ordered_cols]
    cntry_df.reset_index(drop=True, inplace=True)
    cntry_df.to_csv(f'countries/{country}.csv')

    final_df = pd.concat([final_df, cntry_df], ignore_index=True)
```

In [ ]:
```
final_df.to_csv('data/clean_data.csv')
```

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
import statsmodels.api as sm
from scipy.stats import t
```

In [2]:
```
df1 = pd.read_csv("UrbanPopulation.csv")
df2 = pd.read_csv("Co2Emmissions.csv")
```

In [3]:
```
df1.head()
```

Out[3]:

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	
0	Aruba	ABW	Urban population (% of total population)	SP.URB.TOTL.IN.ZS	5.077600e+01	5.076100e+01	5.074600e+01	5.073(
1	Aruba	ABW	Urban population	SP.URB.TOTL	2.772800e+04	2.833000e+04	2.876400e+04	2.9157
2	Africa Eastern and Southern	AFE	Urban population (% of total population)	SP.URB.TOTL.IN.ZS	1.456381e+01	1.481141e+01	1.506925e+01	1.5347
3	Africa Eastern and Southern	AFE	Urban population	SP.URB.TOTL	1.903382e+07	1.987235e+07	2.077079e+07	2.1737
4	Afghanistan	AFG	Urban population (% of total population)	SP.URB.TOTL.IN.ZS	8.401000e+00	8.684000e+00	8.976000e+00	9.2760

5 rows × 67 columns

In [4]:
```
df2.head()
```

Out[4]:

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	...	2014
0	Arab World	ARB	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	NaN	NaN	NaN	NaN	NaN	NaN	...	4.439511
1	Caribbean small states	CSS	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	NaN	NaN	NaN	NaN	NaN	NaN	...	5.582121
2	Central Europe and the Baltics	CEB	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	NaN	NaN	NaN	NaN	NaN	NaN	...	6.139710
3	Early-demographic dividend	EAR	CO2 emissions (metric	EN.ATM.CO2E.PC	NaN	NaN	NaN	NaN	NaN	NaN	...	2.172532

			tons per capita)									
4	East Asia & Pacific	EAS	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	NaN	NaN	NaN	NaN	NaN	NaN	...	5.987349

5 rows × 68 columns

In [5]:
```
df1.drop(['Country Code', 'Indicator Code'], axis=1, inplace=True)
```

In [6]:
```
df2.drop(['Country Code', 'Indicator Code'], axis=1, inplace=True)
```

In [7]:
```
df1.reset_index(inplace=True)

df1.set_index(['Country Name', 'Indicator Name'], inplace=True)
df1 = df1.stack().unstack(level='Indicator Name').reset_index()
```

In [8]:
```
df1.rename(columns={'level_1': 'Year'}, inplace=True)
df1 = df1[~((df1['Year'] == 'level_0') | (df1['Year'] == "index"))]
df1.fillna(0, inplace=True)
df1
```

Out[8]:

Indicator Name	Country Name	Year	Urban population	Urban population (% of total population)
1	Afghanistan	1960	724373.0	8.401
2	Afghanistan	1961	763336.0	8.684
3	Afghanistan	1962	805062.0	8.976
4	Afghanistan	1963	849446.0	9.276
5	Afghanistan	1964	896820.0	9.586
...	...	...	...	...
16830	Zimbabwe	2018	4848158.0	32.209
16831	Zimbabwe	2019	4945719.0	32.210
16832	Zimbabwe	2020	5052214.0	32.242
16833	Zimbabwe	2021	5166388.0	32.303
16834	Zimbabwe	2022	5287038.0	32.395

16569 rows × 4 columns

In [9]:
```
df1.drop(['Urban population'], axis=1, inplace=True)
df1 = df1.groupby(['Country Name'])['Urban population (% of total population)'].aggregat
df1
```

Out[9]:
```
Country Name
Afghanistan                    18.861063
Africa Eastern and Southern    25.406360
Africa Western and Central     31.056023
Albania                        41.172413
Algeria                        53.304683
                                 ...
West Bank and Gaza             65.560413
World                          44.098230
Yemen, Rep.                    22.545984
Zambia                         35.965429
```

```
Zimbabwe                          26.496381
Name: Urban population (% of total population), Length: 263, dtype: float64
```

In [10]:
```
df2.reset_index(inplace=True)

df2.set_index(['Country Name', 'Indicator Name'], inplace=True)
df2 = df2.stack().unstack(level='Indicator Name').reset_index()
```

In [11]:
```
df2.rename(columns={'level_1': 'Year'}, inplace=True)
df2 = df2[~((df2['Year'] == 'level_0') | (df2['Year'] == "index"))]
df2.fillna(0, inplace=True)
df2
```

Out[11]:

Indicator Name	Country Name	Year	CO2 emissions (metric tons per capita)
1	Afghanistan	1990	0.191389
2	Afghanistan	1991	0.180674
3	Afghanistan	1992	0.126517
4	Afghanistan	1993	0.109106
5	Afghanistan	1994	0.096638
...	...	...	...
7580	Zimbabwe	2016	0.723062
7581	Zimbabwe	2017	0.663069
7582	Zimbabwe	2018	0.735435
7583	Zimbabwe	2019	0.663338
7584	Zimbabwe	2020	0.530484

7346 rows × 3 columns

In [12]:
```
df2 = df2.groupby(['Country Name'])['CO2 emissions (metric tons per capita)'].aggregate(
df2
```

Out[12]:
```
Country Name
Afghanistan    0.173175
Albania        1.297776
Algeria        3.061466
Andorra        6.991472
Angola         0.887885
                 ...
Viet Nam       1.322077
World          4.263295
Yemen, Rep.    0.735432
Zambia         0.261004
Zimbabwe       1.023901
Name: CO2 emissions (metric tons per capita), Length: 237, dtype: float64
```

In [13]:
```
new_df = pd.merge(df1, df2, on='Country Name')
new_df
```

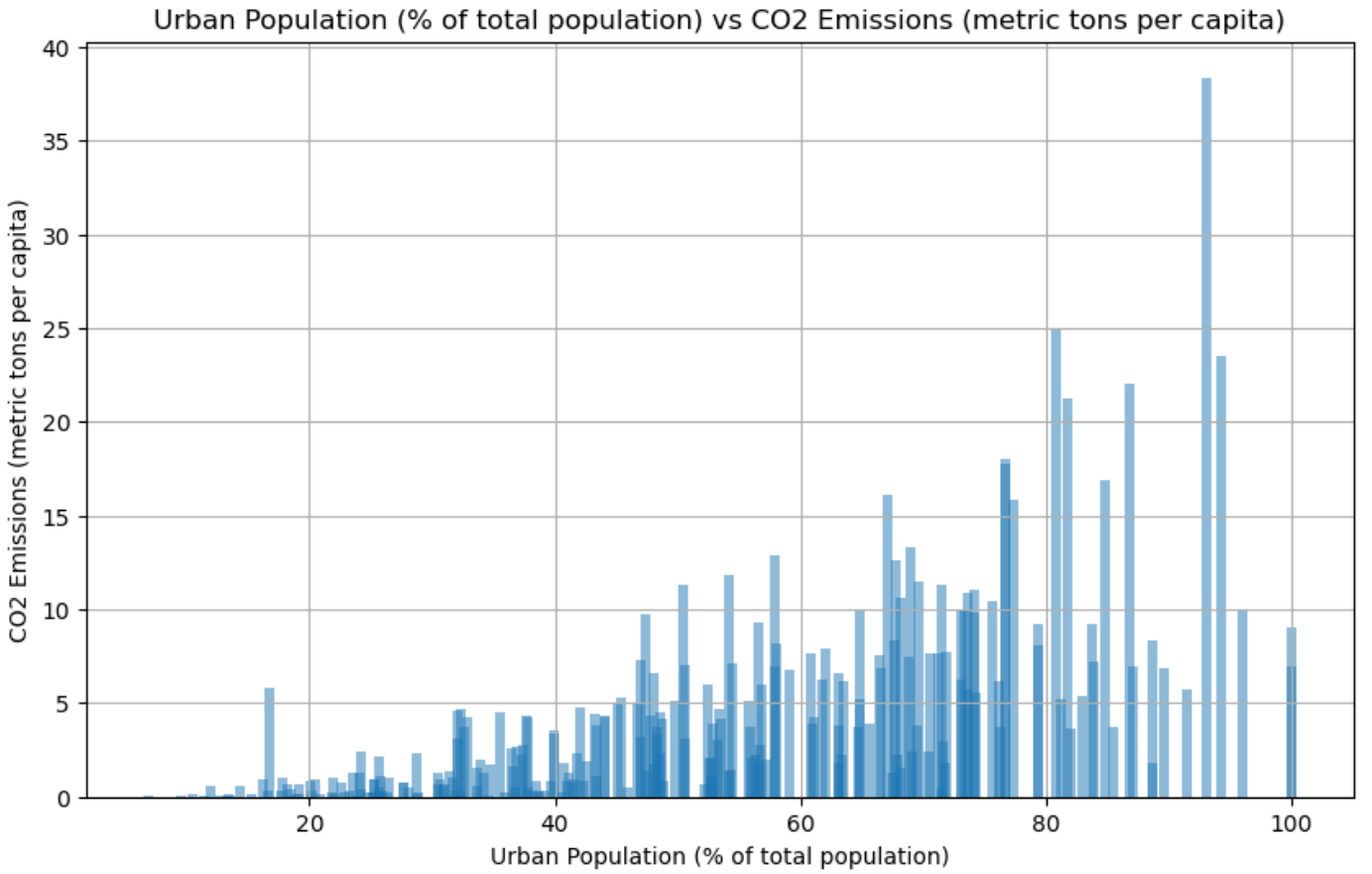
Out[13]:

Country Name	Urban population (% of total population)	CO2 emissions (metric tons per capita)
Afghanistan	18.861063	0.173175
Albania	41.172413	1.297776
Algeria	53.304683	3.061466

Andorra	87.068143	6.991472
Angola	38.499413	0.887885
...	...	...
Viet Nam	23.563984	1.322077
World	44.098230	4.263295
Yemen, Rep.	22.545984	0.735432
Zambia	35.965429	0.261004
Zimbabwe	26.496381	1.023901

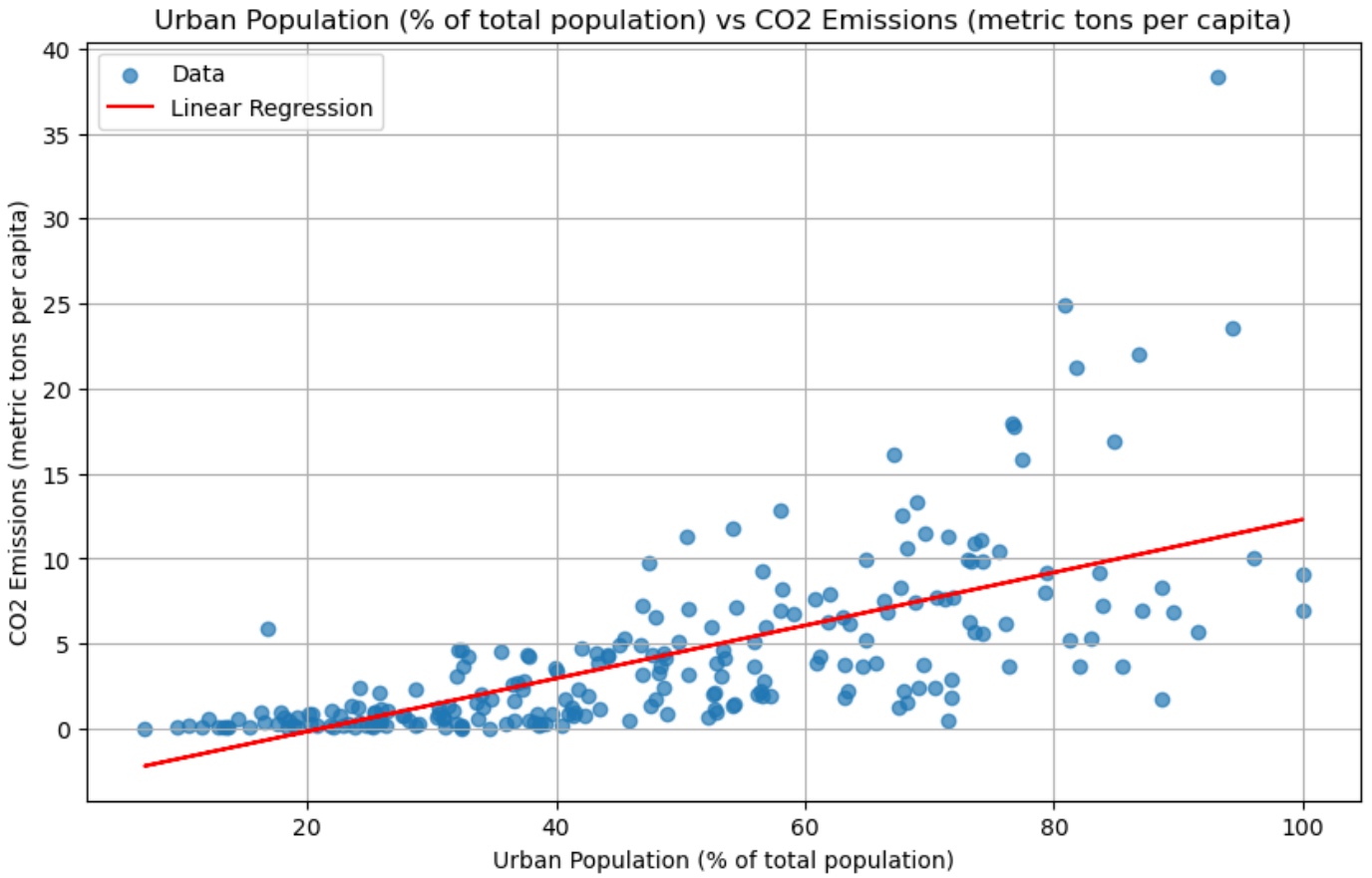
232 rows × 2 columns

In [26]:
```
plt.figure(figsize=(10, 6))
plt.bar(new_df['Urban population (% of total population)'], new_df['CO2 emissions (metri
plt.title('Urban Population (% of total population) vs CO2 Emissions (metric tons per ca
plt.xlabel('Urban Population (% of total population)')
plt.ylabel('CO2 Emissions (metric tons per capita)')
plt.grid(True)
plt.show()
```



In [15]:
```
x = new_df[['Urban population (% of total population)']]
y = new_df['CO2 emissions (metric tons per capita)']

# Creating and fitting the linear regression model
model = LinearRegression()
model.fit(x, y)

# Getting the coefficients of the linear regression model
slope = model.coef_[0]
intercept = model.intercept_
```

```
# Predicting the pollution values using the fitted model
predictions = model.predict(x)

plt.figure(figsize=(10, 6))
plt.scatter(x, y, alpha=0.7, label='Data')
plt.plot(x, predictions, color='red', label='Linear Regression')
plt.title('Urban Population (% of total population) vs CO2 Emissions (metric tons per ca
plt.xlabel('Urban Population (% of total population)')
plt.ylabel('CO2 Emissions (metric tons per capita)')
plt.legend()
plt.grid(True)
plt.show()

print(f"Regression Line Equation: CO2 Emissions (metric tons per capita) = {slope:.2f} *
```




```
Regression Line Equation: CO2 Emissions (metric tons per capita) = 0.16 * Urban Populati
on (% of total population) + -3.27
```

In [16]:

```
x = new_df[['Urban population (% of total population)']]
y = new_df['CO2 emissions (metric tons per capita)']

# Adding constant term for intercept
x = sm.add_constant(x)

# Creating and fitting the linear regression model using statsmodels
model = sm.OLS(y, x).fit()

# Performing hypothesis testing for the coefficient of population
print(model.summary())
```

```
                                OLS Regression Results
==============================================================================
==========
Dep. Variable:     CO2 emissions (metric tons per capita)   R-squared:
0.449
Model:                                               OLS   Adj. R-squared:
0.447
```

```
Method:                                 Least Squares   F-statistic:
187.8
Date:                                Fri, 08 Mar 2024   Prob (F-statistic):
1.20e-31
Time:                                        22:37:23   Log-Likelihood:
-636.42
No. Observations:                                 232   AIC:
1277.
Df Residuals:                                     230   BIC:
1284.
Df Model:                                           1
Covariance Type:                            nonrobust
==============================================================================================
                                                 coef    std err          t      P>|t|
[0.025      0.975]
----------------------------------------------------------------------------------------------
const                                         -3.2657      0.598     -5.461      0.000
-4.444      -2.087
Urban population (% of total population)      0.1557      0.011     13.704      0.000
0.133       0.178
==============================================================================
Omnibus:                      151.798   Durbin-Watson:                   2.053
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1764.319
Skew:                           2.378   Prob(JB):                         0.00
Kurtosis:                      15.645   Cond. No.                         127.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specifi
ed.
```

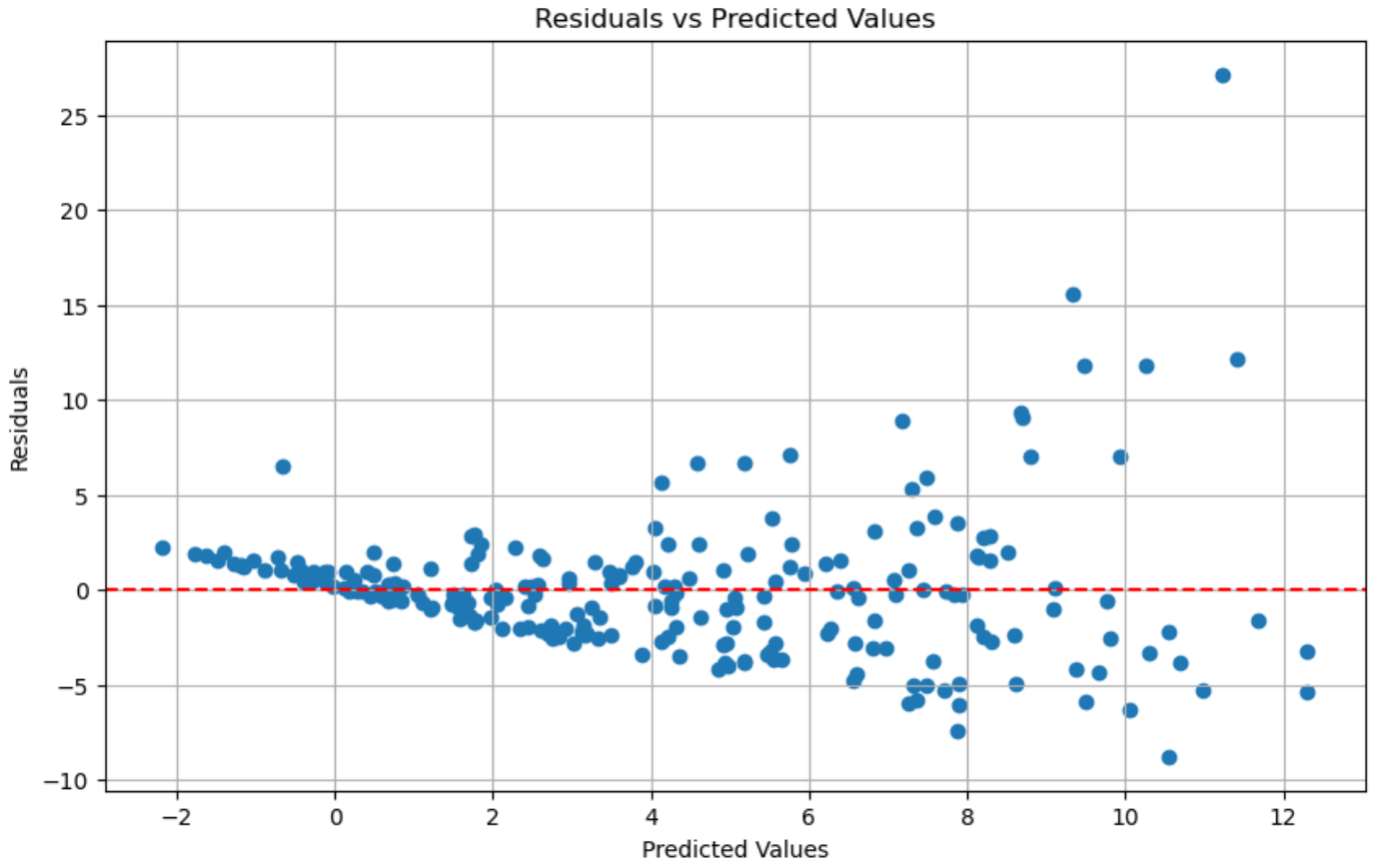
In [17]:

```
slope = model.params['Urban population (% of total population)']
intercept = model.params['const']
p_value = model.pvalues['Urban population (% of total population)']

#slope = model.coef_[0]
#intercept = model.intercept_

# Calculating the residuals
residuals = y - model.predict(x)

# Getting the degrees of freedom
n = len(y)
p = x.shape[1]
df = n - p - 1

# Calculating the residual standard error
residual_standard_error = np.sqrt(np.sum(residuals**2) / df)

# Calculating the t-statistic for the slope
t_statistic = slope / (residual_standard_error / np.sqrt(np.sum((x - x.mean())**2) / (n

# Getting the p-value for the slope
#p_value = 2 * (1 - t.cdf(abs(t_statistic), df))

print(f"Coefficient of Urban population (% of total population) : {slope}")
print(f"Intercept: {intercept}")
print(f"Test Statistic Value: {t_statistic}")
print(f"P-Value: {p_value}")
```

```
Coefficient of Urban population (% of total population) : 0.15572262152338173
Intercept: -3.265685956871571
Test Statistic Value: const                                         0.000000
```

```
Urban population (% of total population)    0.901619
dtype: float64
P-Value: 1.204620716820937e-31
```

In [18]:
```python
y_pred = model.predict(x)

# Residuals
residuals = y - y_pred

# Plotting residuals against predicted values
plt.figure(figsize=(10, 6))
plt.scatter(y_pred, residuals)
plt.xlabel('Predicted Values')
plt.ylabel('Residuals')
plt.title('Residuals vs Predicted Values')
plt.axhline(y=0, color='red', linestyle='--')  # Adding a horizontal line at y=0
plt.grid(True)
plt.show()
```



In [19]:
```python
x2 = sm.add_constant(x)

# Fitting the linear regression model with robust standard errors
model2 = sm.OLS(y, x2).fit(cov_type='HC3')
print(model2.summary())
```

```
                                OLS Regression Results
==============================================================================
Dep. Variable:     CO2 emissions (metric tons per capita)   R-squared:
0.449
Model:                                               OLS   Adj. R-squared:
0.447
Method:                                    Least Squares   F-statistic:
78.46
Date:                                   Fri, 08 Mar 2024   Prob (F-statistic):
2.26e-16
Time:                                           22:37:25   Log-Likelihood:
```

```
-636.42
No. Observations:                                       232   AIC:
1277.
Df Residuals:                                           230   BIC:
1284.
Df Model:                                                 1
Covariance Type:                                        HC3
==============================================================================================
====================
                                                   coef    std err          z      P>|z|
[0.025      0.975]
----------------------------------------------------------------------------------------------
--------------------
const                                          -3.2657      0.652     -5.005      0.000
-4.545      -1.987
Urban population (% of total population)        0.1557      0.018      8.858      0.000
0.121       0.190
==============================================================================
Omnibus:                      151.798   Durbin-Watson:                   2.053
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1764.319
Skew:                           2.378   Prob(JB):                         0.00
Kurtosis:                      15.645   Cond. No.                         127.
==============================================================================

Notes:
[1] Standard Errors are heteroscedasticity robust (HC3)
```

In [20]:

```
slope2 = model2.params['Urban population (% of total population)']
intercept2 = model2.params['const']
p_value2 = model2.pvalues['Urban population (% of total population)']

#slope2 = model2.coef_[0]#['Urban population (% of total population) ']
#intercept2 = model2.intercept_
#p_value = model.pvalues['Urban population (% of total population) ']

# Calculating the residuals
residuals2 = y - model2.predict(x2)

# Getting the degrees of freedom
n = len(y)
p = x2.shape[1]
df = n - p - 1

# Calculating the residual standard error
residual_standard_error2 = np.sqrt(np.sum(residuals2**2) / df)

# Calculating the t-statistic for the slope
t_statistic2 = slope2 / (residual_standard_error2 / np.sqrt(np.sum((x2 - x2.mean())**2)

# Getting the p-value for the slope
#p_value2 = 2 * (1 - t.cdf(abs(t_statistic2), df))

print(f"Coefficient of Urban population (% of total population) : {slope2}")
print(f"Intercept: {intercept2}")
print(f"Test Statistic Value: {t_statistic2}")
print(f"P-Value: {p_value2}")
```

```
Coefficient of Urban population (% of total population) : 0.15572262152338173
Intercept: -3.265685956871571
Test Statistic Value: const                                          0.000000
Urban population (% of total population)    0.901619
dtype: float64
P-Value: 8.169642682111703e-19
```

Stats Project

Diane Chiang

2024-02-18

```
library(zoo)
library(dplyr)
library(tidyverse)
library(ggplot2)
```

```
country = read.csv("/Users/dianechiang/Desktop/data557/Group Project/World Bank/ESGCountry.csv")
country = country[country$Region != "",]
country = country %>% select(Country.Code)
head(country)
```

```
##   Country.Code
## 1          AFG
## 2          AGO
## 3          ALB
## 4          AND
## 6          ARE
## 7          ARG
```

```
co2_data = read.csv("/Users/dianechiang/Desktop/data557/Group Project/World Bank/ESGData.csv")
co2_data = co2_data %>% inner_join(country, by=c("Country.Code")) %>%
  select(-one_of("Country.Code", "Indicator.Code", "X"))
head(co2_data)
```

```
##   Country.Name
## 1  Afghanistan
## 2  Afghanistan
## 3  Afghanistan
## 4  Afghanistan
## 5  Afghanistan
## 6  Afghanistan
##                                                    Indicator.Name X1960
## 1 Access to clean fuels and technologies for cooking (% of population)    NA
## 2                             Access to electricity (% of population)    NA
## 3            Adjusted savings: natural resources depletion (% of GNI)    NA
## 4                   Adjusted savings: net forest depletion (% of GNI)    NA
## 5                                Agricultural land (% of land area)    NA
## 6          Agriculture, forestry, and fishing, value added (% of GDP)    NA
##      X1961    X1962    X1963  X1964    X1965    X1966    X1967    X1968
## 1       NA       NA       NA     NA       NA       NA       NA       NA
## 2       NA       NA       NA     NA       NA       NA       NA       NA
## 3       NA       NA       NA     NA       NA       NA       NA       NA
## 4       NA       NA       NA     NA       NA       NA       NA       NA
## 5 57.87836 57.95502 58.03168 58.116 58.12367 58.19266 58.22946 58.23099
## 6       NA       NA       NA     NA       NA       NA       NA       NA
##      X1969      X1970      X1971      X1972      X1973      X1974      X1975
## 1       NA         NA         NA         NA         NA         NA         NA
## 2       NA         NA         NA         NA         NA         NA         NA
## 3       NA  0.5038543  0.6445909  0.7867444  1.2427550  1.4930569  1.8497008
## 4       NA  0.2794117  0.3375673  0.3892899  0.7517331  0.7831858  0.7956867
## 5 58.25552 58.2708554 58.3168514 58.3352498 58.3370897 58.3387762 58.3387762
## 6       NA         NA         NA         NA         NA         NA         NA
##        X1976    X1977      X1978      X1979      X1980      X1981    X1982
## 1         NA       NA         NA         NA         NA         NA       NA
## 2         NA       NA         NA         NA         NA         NA       NA
## 3  1.9629641  1.84075  1.6182660  1.6929395  1.6110747  1.2269143       NA
## 4  0.7569508  0.58974  0.7349711  0.5599368  0.5816848  0.4342628       NA
## 5 58.3383162 58.33832 58.3383162 58.3367830 58.3367830 58.3429158 58.34445
## 6         NA       NA         NA         NA         NA         NA       NA
##      X1983    X1984    X1985    X1986    X1987    X1988    X1989    X1990
## 1       NA       NA       NA       NA       NA       NA       NA       NA
## 2       NA       NA       NA       NA       NA       NA       NA       NA
## 3       NA       NA       NA       NA       NA       NA       NA       NA
## 4       NA       NA       NA       NA       NA       NA       NA       NA
## 5 58.34445 58.34445 58.34445 58.34445 58.33065 58.32298 58.32298 58.32298
## 6       NA       NA       NA       NA       NA       NA       NA       NA
##      X1991    X1992    X1993    X1994    X1995    X1996    X1997    X1998
## 1       NA       NA       NA       NA       NA       NA       NA       NA
## 2       NA       NA       NA       NA       NA       NA       NA       NA
## 3       NA       NA       NA       NA       NA       NA       NA       NA
## 4       NA       NA       NA       NA       NA       NA       NA       NA
## 5 58.30765 58.30765 58.16046 57.97495 57.89829 57.88909 57.94735 58.05927
## 6       NA       NA       NA       NA       NA       NA       NA       NA
##      X1999     X2000     X2001    X2002    X2003    X2004    X2005    X2006
## 1       NA  6.700000  7.700000  8.80000 10.00000 11.10000 12.50000 13.90000
## 2       NA  4.446891  9.294527 14.13362 18.97116 23.81418 28.66967 33.54442
## 3       NA        NA        NA       NA       NA       NA       NA       NA
## 4       NA        NA        NA       NA       NA       NA       NA       NA
## 5 57.89982 57.945817 57.947350 57.93968 58.08380 58.15127 58.13440 58.12367
## 6       NA        NA        NA 38.62789 37.41886 29.72107 31.11485 28.63597
```

```
##      X2007    X2008      X2009      X2010      X2011      X2012      X2013
## 1 15.30000 16.80000 18.2000000 19.7000000 21.3000000 22.7000000 24.3000000
## 2 38.44000 42.40000 48.2790070 42.7000000 43.2220189 69.1000000 68.0408783
## 3       NA       NA  0.2703654  0.3594540  0.3866440  0.3809881  0.3350905
## 4       NA       NA  0.2298643  0.2927884  0.2442393  0.2113761  0.2114126
## 5 58.12980 58.13287 58.1328672 58.1344004 58.1313340 58.1298008 58.1236680
## 6 30.10501 24.89227 29.2975011 26.2100685 23.7436640 24.3908736 22.8106627
##        X2014     X2015      X2016      X2017      X2018      X2019      X2020
## 1 25.7000000 27.250000 28.5000000 30.0000000 31.1000000 32.4500000 33.8000000
## 2 89.5000000 71.500000 97.7000000 97.7000000 93.4308777 97.7000000 97.7000000
## 3  0.3155709  0.290261  0.3632821  0.3508792  0.4010530  0.3701308  0.2436684
## 4  0.2166090  0.232762  0.2847814  0.2298215  0.2376147  0.2693532  0.2379582
## 5 58.1236680 58.123668 58.1236680 58.1236680 58.2769882 58.2769882 58.7415482
## 6 22.1370414 20.634323 25.7403140 26.4201991 22.0428968 25.7739707 29.9755825
##        X2021 X2022 X2023
## 1 35.4000000    NA    NA
## 2 97.7000000    NA    NA
## 3  0.3359349    NA    NA
## 4  0.3177321    NA    NA
## 5 58.7415482    NA    NA
## 6 33.5976189    NA    NA
```

```
## first_half -> 1990 - 2004, second_half -> 2005 - 2020
co2_data = co2_data[co2_data$Indicator.Name == "CO2 emissions (metric tons per capita)",]
co2_data = co2_data %>% pivot_longer(cols = starts_with("X"), names_to = "Year", values_to = "CO2_
emissions_metric_tons_per_capita") %>% na.omit()
co2_data$Year<-gsub("X","",as.character(co2_data$Year))
co2_data = co2_data %>% select("Country.Name", "Year", "CO2_emissions_metric_tons_per_capita")
co2_data$Year = as.numeric(co2_data$Year)
co2_data$Year_Cat = "second_half"
co2_data$Year_Cat[co2_data$Year >= 1990 & co2_data$Year <= 2004] <- "first_half"
co2_data_mean = co2_data %>% group_by(Country.Name, Year_Cat) %>% summarise(mean_CO2_emissions = m
ean(CO2_emissions_metric_tons_per_capita), .groups = 'drop')
head(co2_data_mean)
```

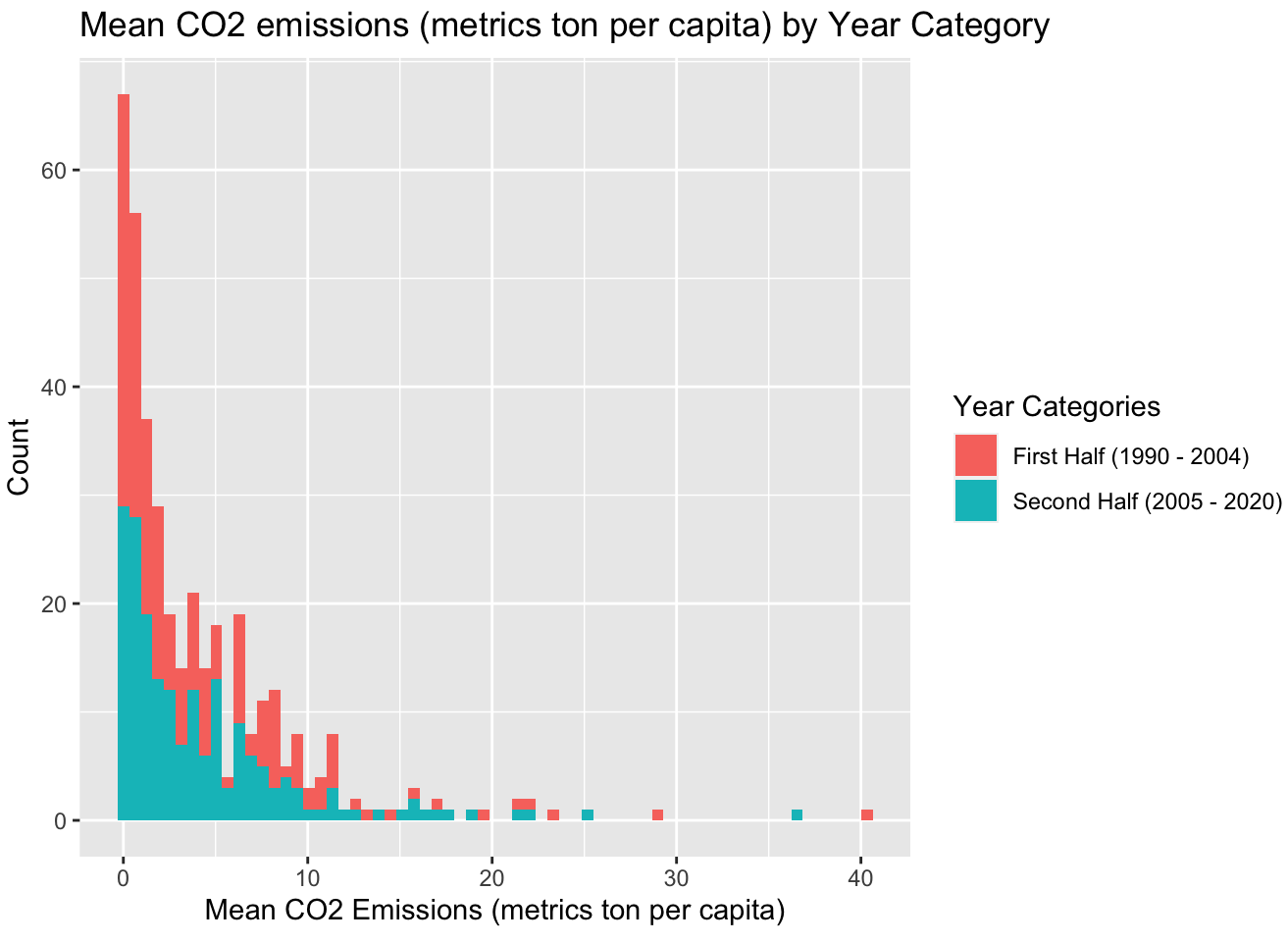
```
## # A tibble: 6 × 3
##   Country.Name Year_Cat    mean_CO2_emissions
##   <chr>        <chr>                    <dbl>
## 1 Afghanistan  first_half              0.0924
## 2 Afghanistan  second_half             0.249
## 3 Albania      first_half              0.963
## 4 Albania      second_half             1.61
## 5 Algeria      first_half              2.59
## 6 Algeria      second_half             3.50
```

Global t test of difference in top 15 years and bottom 15 years of CO2 emission

```
co2_data_top = co2_data_mean[co2_data_mean$Year_Cat == "first_half",]
co2_data_bottom = co2_data_mean[co2_data_mean$Year_Cat == "second_half",]

co2_mean_top = co2_data_mean[co2_data_mean$Year_Cat == "first_half",]$mean_CO2_emissions
co2_mean_bottom = co2_data_mean[co2_data_mean$Year_Cat == "second_half",]$mean_CO2_emissions

t.test(co2_mean_bottom, co2_mean_top, paired = TRUE, var.equal=FALSE, corr = FALSE)
```

```
##
##  Paired t-test
##
## data:  co2_mean_bottom and co2_mean_top
## t = 0.57062, df = 190, p-value = 0.5689
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -0.1507267  0.2734286
## sample estimates:
## mean difference
##      0.06135097
```

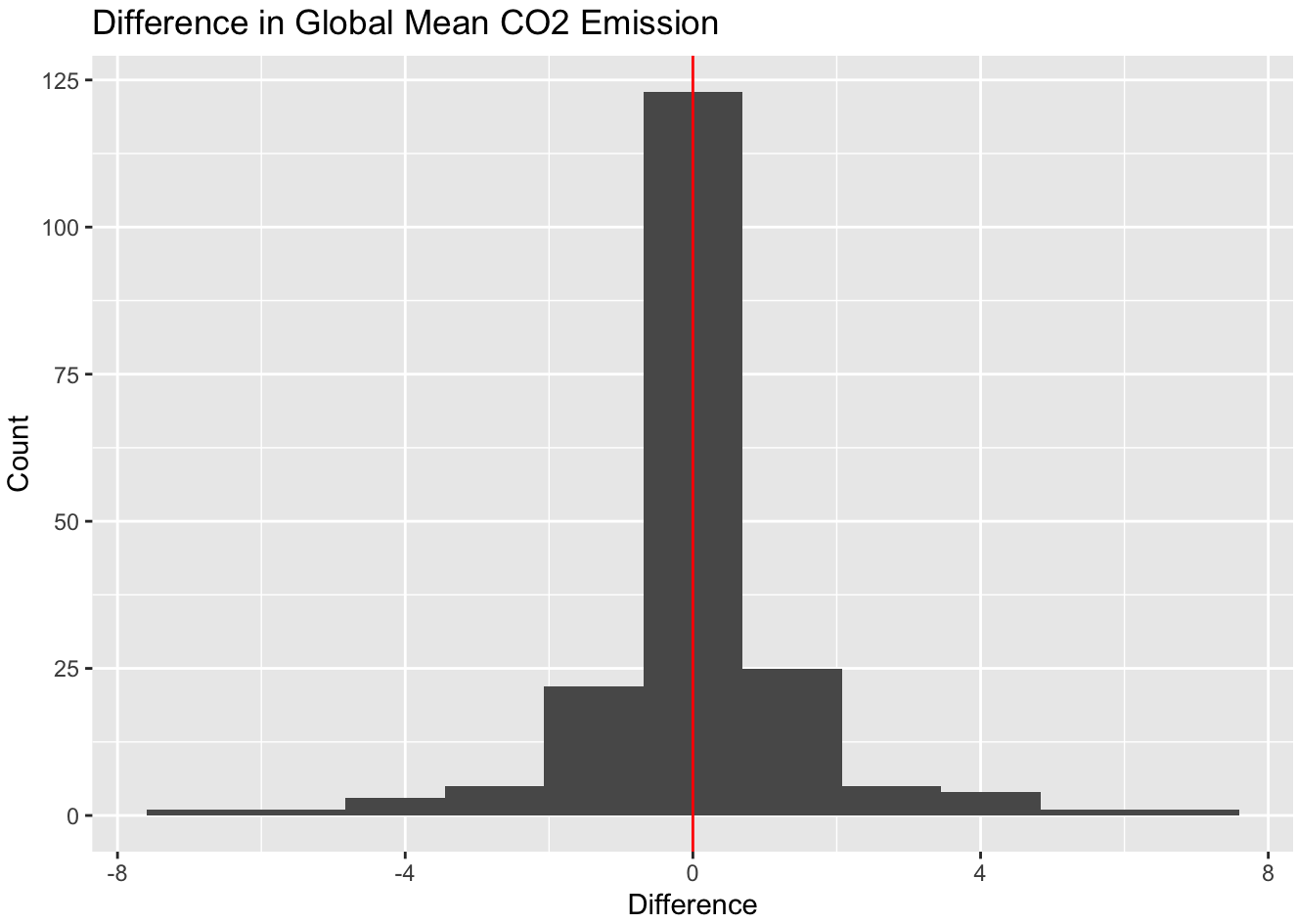
```
co2_data_mean %>%
  ggplot(aes(x = mean_CO2_emissions, fill = Year_Cat)) +
  geom_histogram(bins = 65) +
  ggtitle("Mean CO2 emissions (metrics ton per capita) by Year Category") +
  xlab("Mean CO2 Emissions (metrics ton per capita)") + ylab("Count") +
  scale_fill_discrete(name = "Year Categories", labels=c('First Half (1990 - 2004)', 'Second Half
(2005 - 2020)'))
```



```
first = co2_data_mean[co2_data_mean$Year_Cat == "first_half",]
second = co2_data_mean[co2_data_mean$Year_Cat == "second_half",]
fs_combine = first %>% inner_join(second, by=c("Country.Name")) %>% mutate(diff=mean_CO2_emissions.y - mean_CO2_emissions.x)
head(fs_combine)
```

```
## # A tibble: 6 × 6
##   Country.Name   Year_Cat.x mean_CO2_emissions.x Year_Cat.y mean_CO2_emissions.y
##   <chr>          <chr>                    <dbl> <chr>                    <dbl>
## 1 Afghanistan    first_half              0.0924 second_ha…               0.249
## 2 Albania        first_half              0.963  second_ha…               1.61
## 3 Algeria        first_half              2.59   second_ha…               3.50
## 4 Andorra        first_half              7.31   second_ha…               6.69
## 5 Angola         first_half              0.877  second_ha…               0.898
## 6 Antigua and B… first_half              3.86   second_ha…               5.29
## # ℹ 1 more variable: diff <dbl>
```

```
fs_combine %>%
  ggplot(aes(x = diff)) +
  geom_histogram(bins = 11) +
  ggtitle("Difference in Global Mean CO2 Emission") +
  geom_vline(xintercept = 0, color="red") +
  xlab("Difference") + ylab("Count")
```



30 most CO2 emission (according to bottom) [mean diff]

```
most_CO2_emission = co2_data_top[order(co2_data_top$mean_CO2_emissions, decreasing = TRUE),] %>% t
op_n(30)
```

```
## Selecting by mean_CO2_emissions
```

```
least_CO2_emission = co2_data_top[order(co2_data_top$mean_CO2_emissions, decreasing = TRUE),] %>%
top_n(-30)
```

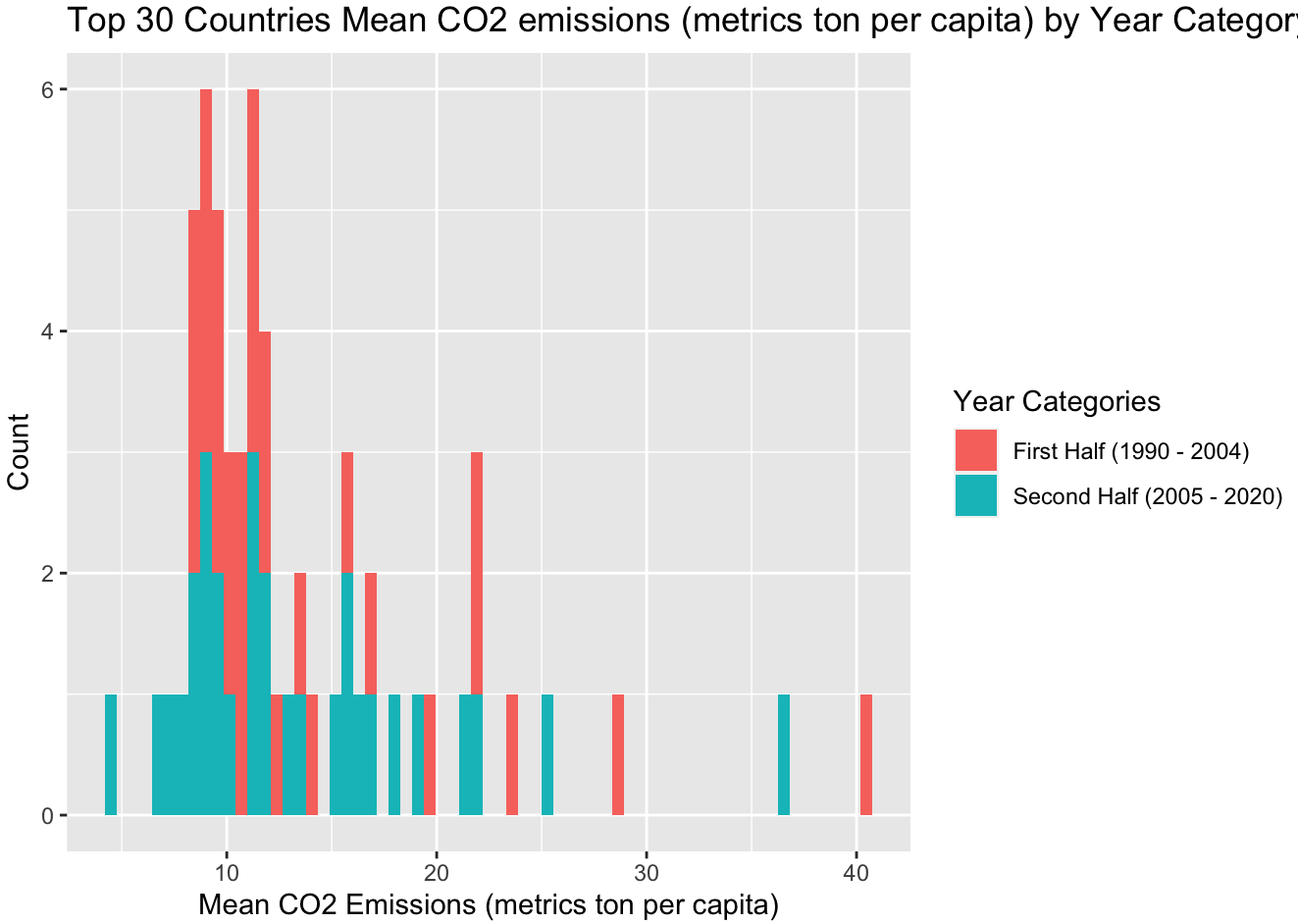
```
## Selecting by mean_CO2_emissions
```

```
most = co2_data_bottom %>% inner_join(most_CO2_emission, by=c("Country.Name")) %>% mutate(diff=mea
n_CO2_emissions.x - mean_CO2_emissions.y)
t.test(most$diff, var.equal = FALSE, corr = FALSE)
```

```
##
##  One Sample t-test
##
## data:  most$diff
## t = -0.75826, df = 29, p-value = 0.4544
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -1.628907  0.747765
## sample estimates:
## mean of x
## -0.440571
```

```
most = most %>% select(Country.Name, Year_Cat.x, mean_CO2_emissions.x, Year_Cat.y, mean_CO2_emissi
ons.y)
colnames(most) <- c('Country.Name','Year_Cat.x','second_half', 'Year_Cat.y', 'first_half')
most = most %>% select(Country.Name, second_half, first_half)
most = most %>% pivot_longer(cols=c('first_half', 'second_half'),
                             names_to='Year_Category',
                             values_to='mean_CO2_emissions')
```

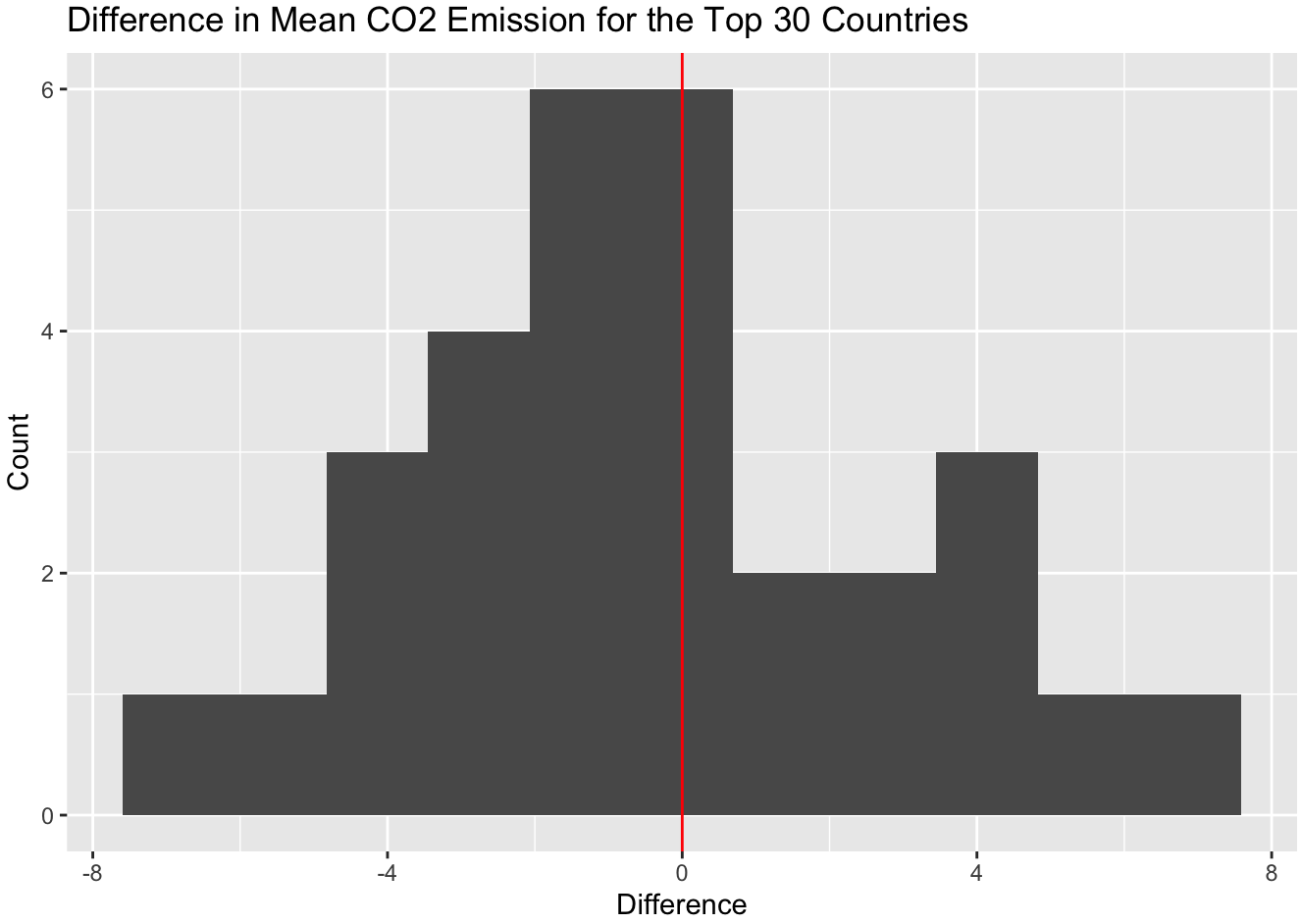
```
most %>%
  ggplot(aes(x = mean_CO2_emissions, fill = Year_Category)) +
  geom_histogram(bins = 65) +
  ggtitle("Top 30 Countries Mean CO2 emissions (metrics ton per capita) by Year Category") +
  xlab("Mean CO2 Emissions (metrics ton per capita)") + ylab("Count") +
  scale_fill_discrete(name = "Year Categories", labels=c('First Half (1990 - 2004)', 'Second Half
(2005 - 2020)'))
```




```
most = co2_data_bottom %>% inner_join(most_CO2_emission, by=c("Country.Name")) %>% mutate(diff=mea
n_CO2_emissions.x - mean_CO2_emissions.y)
head(most)
```

```
## # A tibble: 6 × 6
##   Country.Name   Year_Cat.x mean_CO2_emissions.x Year_Cat.y mean_CO2_emissions.y
##   <chr>          <chr>                    <dbl> <chr>                    <dbl>
## 1 Australia      second_ha…               17.0  first_half                16.8
## 2 Bahrain        second_ha…               22.0  first_half                22.1
## 3 Belgium        second_ha…                8.89 first_half                11.3
## 4 Brunei Daruss… second_ha…               17.9  first_half                14.2
## 5 Canada         second_ha…               15.8  first_half                15.8
## 6 Czechia        second_ha…               10.3  first_half                12.4
## # ℹ 1 more variable: diff <dbl>
```

```
most %>%
  ggplot(aes(x = diff)) +
  geom_histogram(bins = 11) +
  ggtitle("Difference in Mean CO2 Emission for the Top 30 Countries") +
  geom_vline(xintercept = 0, color="red") +
  xlab("Difference") + ylab("Count")
```



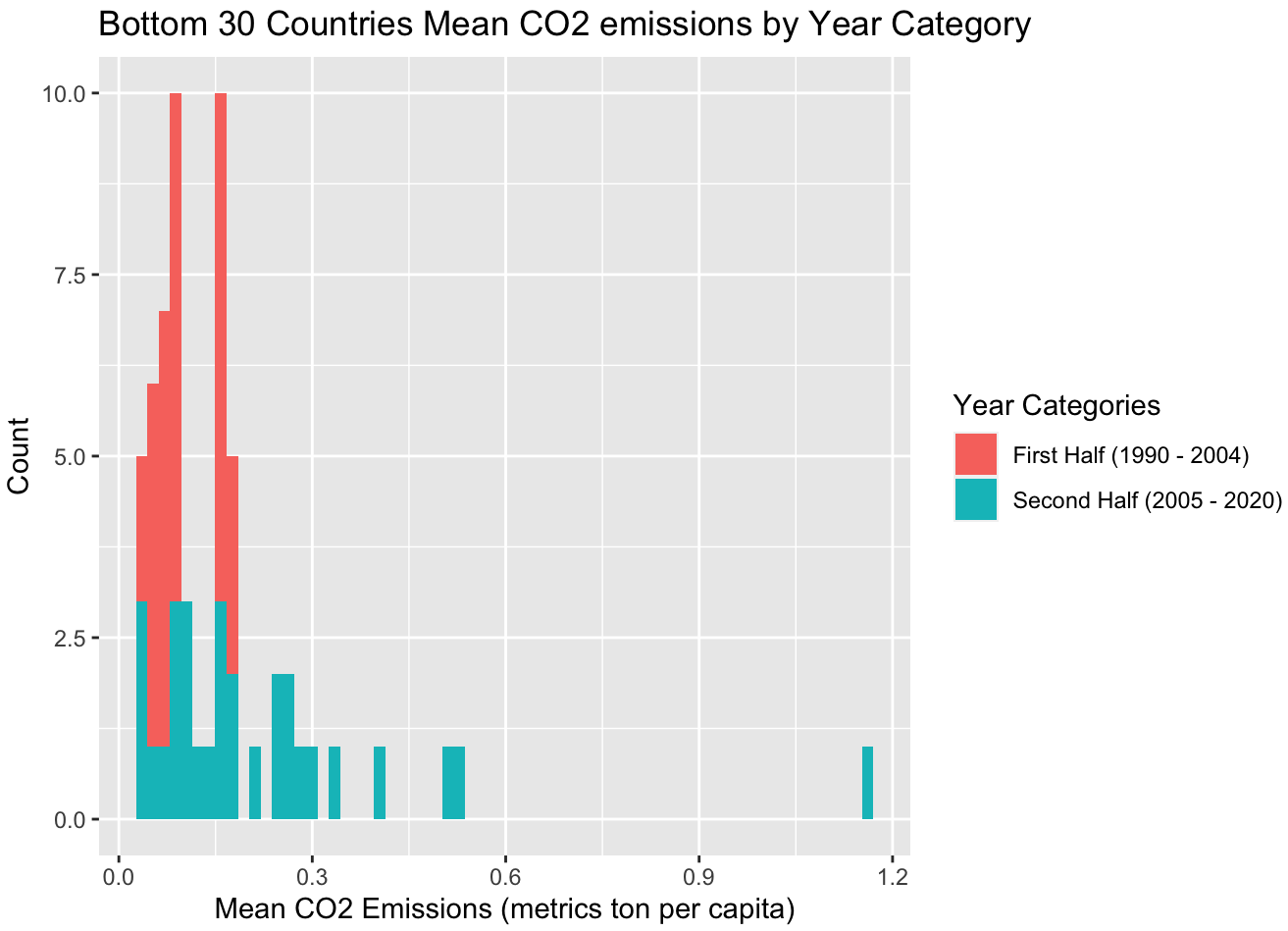
30 least CO2 emission (according to top) -> [mean diff]

```
least = co2_data_bottom %>%
  inner_join(least_CO2_emission, by=c("Country.Name")) %>%
  mutate(diff=mean_CO2_emissions.x - mean_CO2_emissions.y)

t.test(least$diff, var.equal = FALSE, corr = FALSE)
```

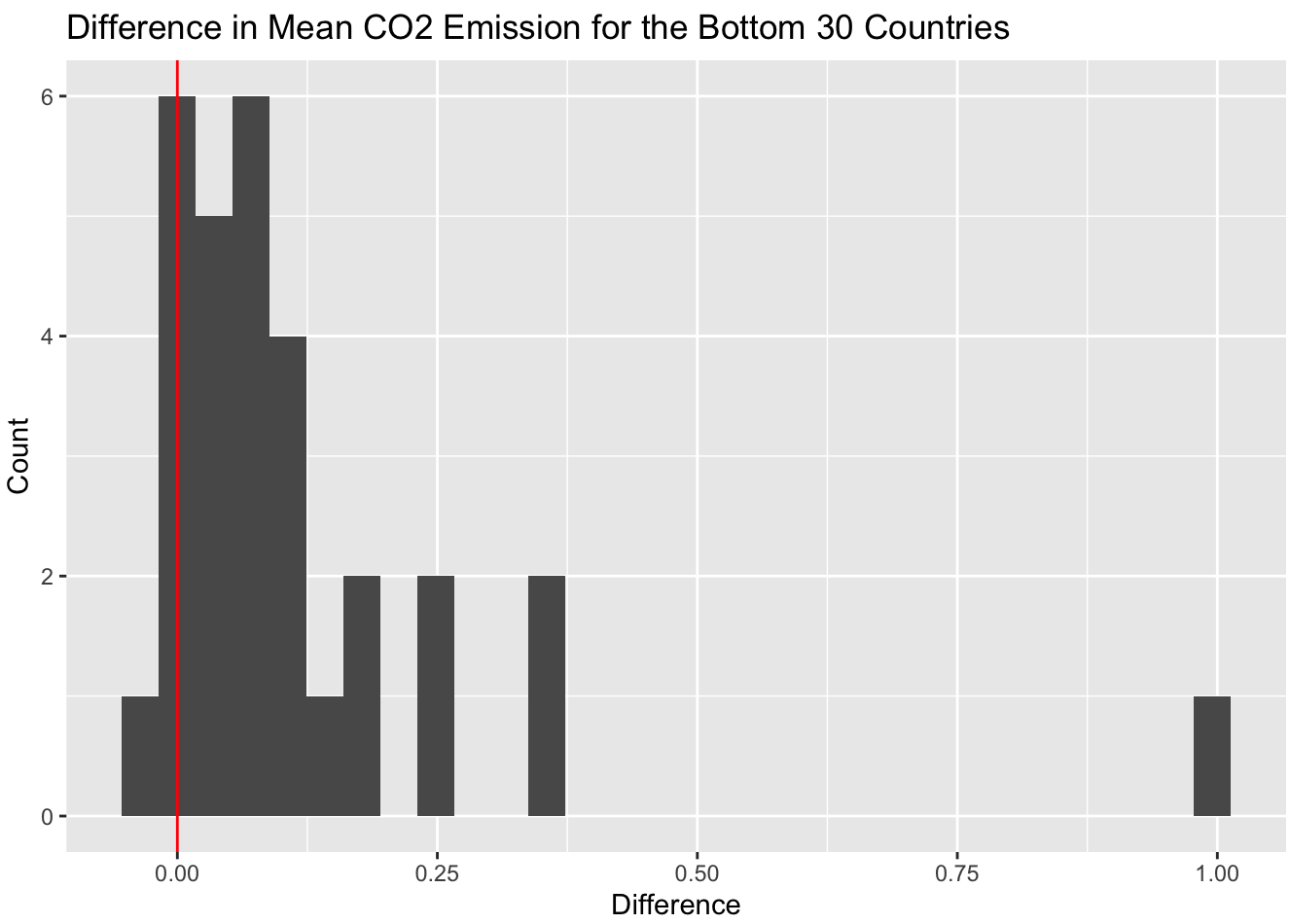
```
##
##  One Sample t-test
##
## data:  least$diff
## t = 3.4364, df = 29, p-value = 0.001801
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.04967302 0.19572834
## sample estimates:
## mean of x
## 0.1227007
```

```
least = least %>% select(Country.Name, Year_Cat.x, mean_CO2_emissions.x, Year_Cat.y, mean_CO2_emis
sions.y)
colnames(least) <- c('Country.Name','Year_Cat.x','second_half', 'Year_Cat.y', 'first_half')
least = least %>% select(Country.Name, second_half, first_half)
least = least %>% pivot_longer(cols=c('first_half', 'second_half'),
                               names_to='Year_Category',
                               values_to='mean_CO2_emissions')
```

```
least %>%
  ggplot(aes(x = mean_CO2_emissions, fill = Year_Category)) +
  geom_histogram(bins = 65) +
  ggtitle("Bottom 30 Countries Mean CO2 emissions by Year Category") +
  xlab("Mean CO2 Emissions (metrics ton per capita)") + ylab("Count") +
  scale_fill_discrete(name = "Year Categories", labels=c('First Half (1990 - 2004)', 'Second Half
(2005 - 2020)'))
```




```
least = co2_data_bottom %>%
  inner_join(least_CO2_emission, by=c("Country.Name")) %>%
  mutate(diff=mean_CO2_emissions.x - mean_CO2_emissions.y)
least %>%
  ggplot(aes(x = diff)) +
  geom_histogram(bins = 30) +
  ggtitle("Difference in Mean CO2 Emission for the Bottom 30 Countries") +
  geom_vline(xintercept = 0, color="red") +
  xlab("Difference") + ylab("Count")
```



Select most 3 to do individual mean diff

```
colnames(most) = c("Country Name", "Year_Cat_second_half", "mean_CO2_emissions_sh", "Year_Cat_firs
t_half", "mean_CO2_emissions_fh", "Diff")
head(most[order(most$mean_CO2_emissions_sh, decreasing = TRUE),])
```

```
## # A tibble: 6 × 6
##   `Country Name`  Year_Cat_second_half mean_CO2_emissions_sh Year_Cat_first_half
##   <chr>           <chr>                                <dbl> <chr>
## 1 Qatar           second_half                           36.5 first_half
## 2 Kuwait          second_half                           25.2 first_half
## 3 Bahrain         second_half                           22.0 first_half
## 4 United Arab Em… second_half                           21.4 first_half
## 5 Luxembourg      second_half                           19.1 first_half
## 6 Brunei Darussa… second_half                           17.9 first_half
## # ℹ 2 more variables: mean_CO2_emissions_fh <dbl>, Diff <dbl>
```

```
qatar_co2 = co2_data[co2_data$Country.Name == "Qatar",]
qatar_co2_fh = qatar_co2[qatar_co2$Year_Cat == "first_half",]$CO2_emissions_metric_tons_per_capita
qatar_co2_sh = qatar_co2[qatar_co2$Year_Cat == "second_half",]$CO2_emissions_metric_tons_per_capit
a
qatar_co2_sh = head(qatar_co2_sh, -1)

qatar_co2_fh
```

```
##  [1] 28.39962 32.82686 31.14326 34.18354 36.93155 36.97628 39.56827 46.11881
##  [9] 45.61590 47.28894 44.37925 42.20579 45.56469 46.41689 47.65696
```

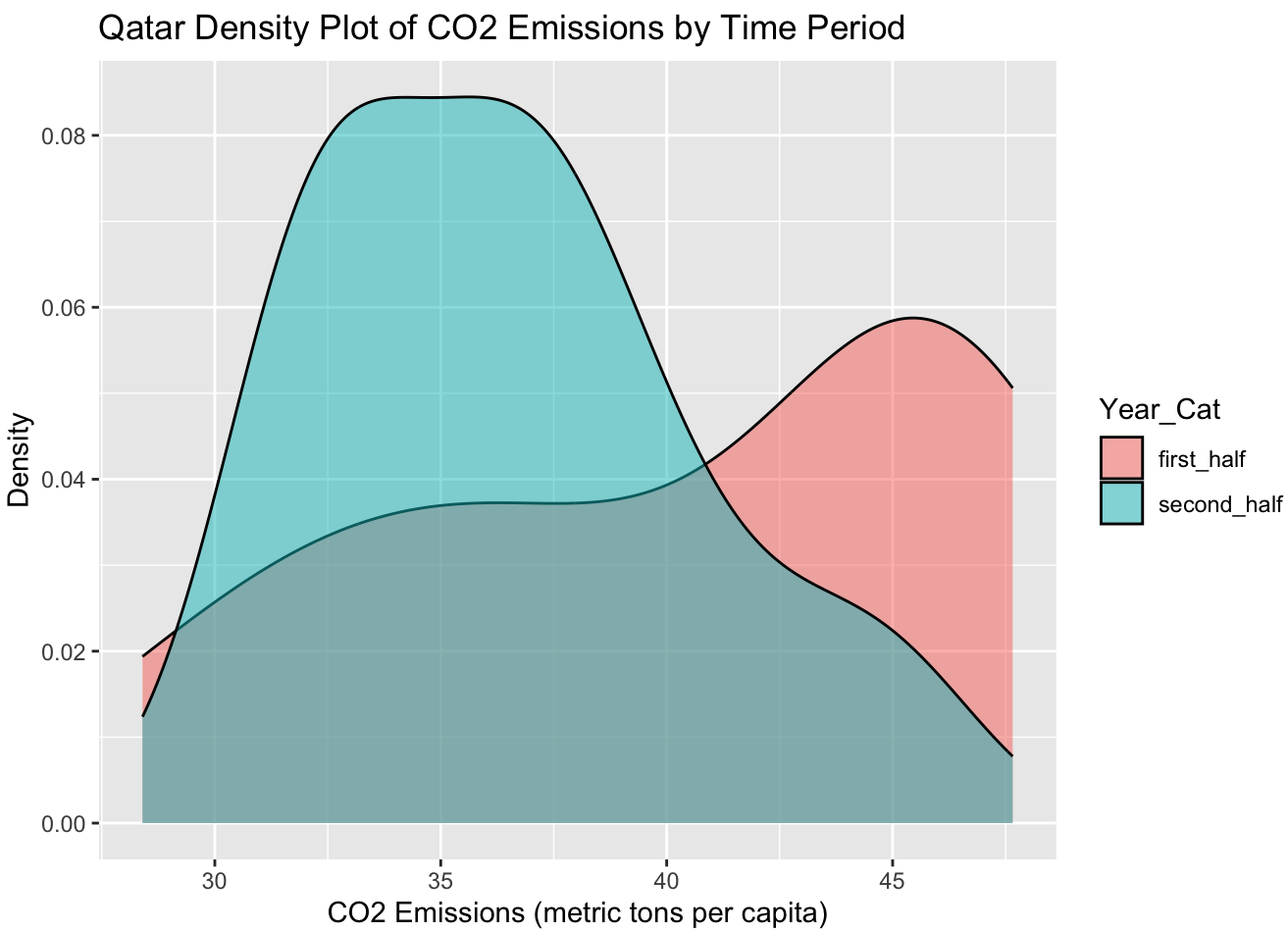
```
qatar_co2_sh
```

```
##  [1] 45.40609 43.28857 40.60945 36.88995 33.72730 35.54827 37.97949 39.58214
##  [9] 37.60288 37.10503 35.29042 33.54957 32.25664 31.48097 31.87720
```

```
t.test(qatar_co2_sh, qatar_co2_fh, paired = TRUE, alternative = "two.sided", corr=FALSE)
```

```
##
##  Paired t-test
##
## data:  qatar_co2_sh and qatar_co2_fh
## t = -1.401, df = 14, p-value = 0.183
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -8.956594  1.878913
## sample estimates:
## mean difference
##       -3.538841
```

```
# Create a density plot for each time period
ggplot(qatar_co2, aes(x = CO2_emissions_metric_tons_per_capita, fill = Year_Cat)) +
  geom_density(alpha = 0.5) +
  labs(title = "Qatar Density Plot of CO2 Emissions by Time Period",
       x = "CO2 Emissions (metric tons per capita)",
       y = "Density")
```




```
us_co2 = co2_data[co2_data$Country.Name == "United States",]
us_co2_fh = us_co2[us_co2$Year_Cat == "first_half",]$CO2_emissions_metric_tons_per_capita
us_co2_sh = us_co2[us_co2$Year_Cat == "second_half",]$CO2_emissions_metric_tons_per_capita
us_co2_sh = head(us_co2_sh, -1)

us_co2_fh
```

```
##  [1] 19.40734 19.00339 19.02285 19.21833 19.25619 19.21690 19.57537 20.33085
##  [9] 20.26629 20.10112 20.46980 20.17154 19.44553 19.50651 19.59762
```

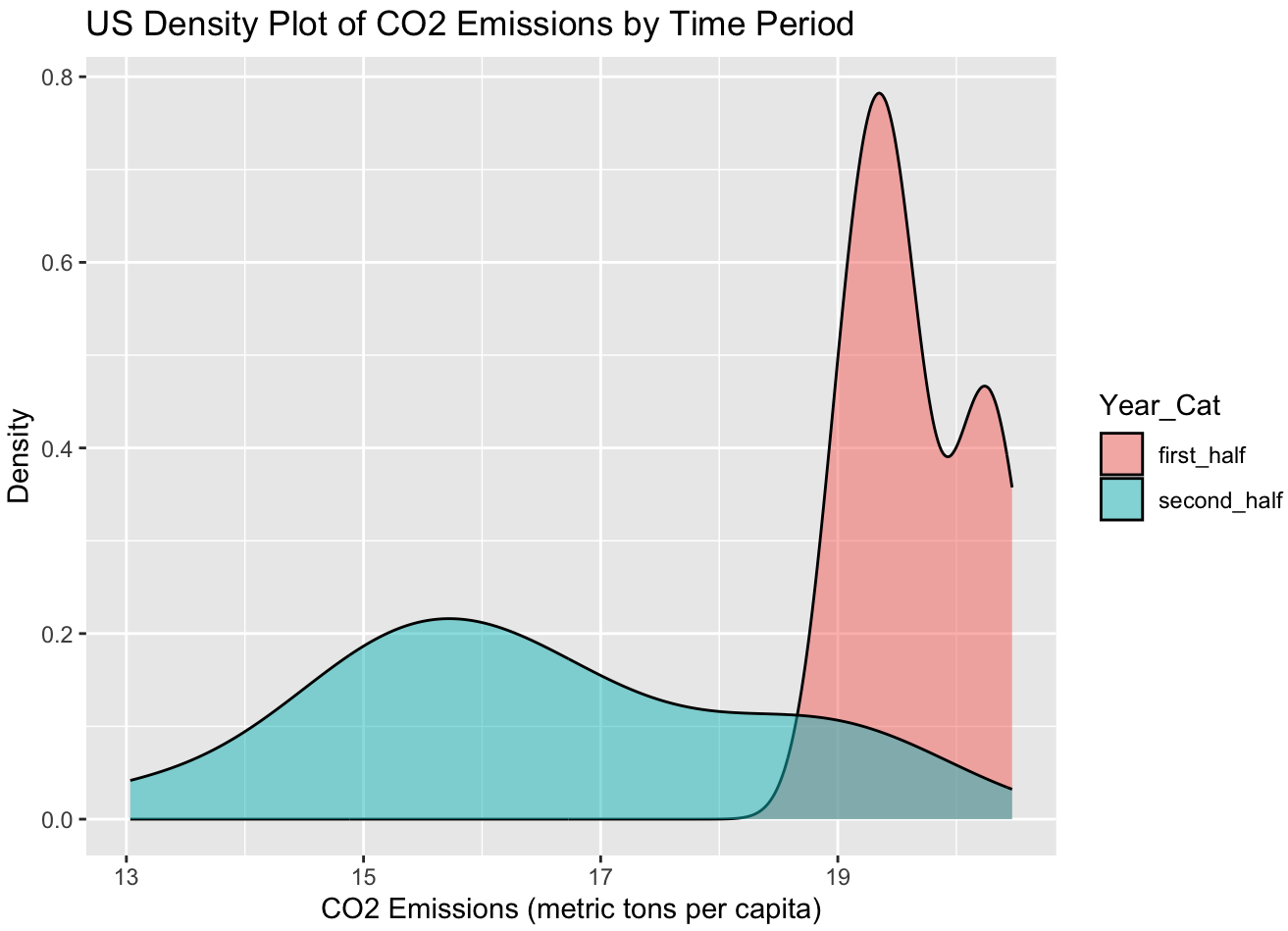
```
us_co2_sh
```

```
##  [1] 19.46927 18.94592 19.04291 18.27849 16.80868 17.43174 16.60419 15.78976
##  [9] 16.11118 16.04092 15.56002 15.14988 14.82325 15.22252 14.67338
```

```
t.test(us_co2_sh, us_co2_fh, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  us_co2_sh and us_co2_fh
## t = -5.8835, df = 14, p-value = 3.98e-05
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -4.060645 -1.891022
## sample estimates:
## mean difference
##       -2.975834
```

```
# Create a density plot for each time period
ggplot(us_co2, aes(x = CO2_emissions_metric_tons_per_capita, fill = Year_Cat)) +
  geom_density(alpha = 0.5) +
  labs(title = "US Density Plot of CO2 Emissions by Time Period",
       x = "CO2 Emissions (metric tons per capita)",
       y = "Density")
```



Select least 3 to do individual mean diff

```
colnames(least) = c("Country Name", "Year_Cat_sh", "mean_CO2_emissions_sh", "Year_Cat_fh", "mean_C
O2_emissions_fh", "Diff")
least = least[order(least$mean_CO2_emissions_sh, decreasing = FALSE),]
head(least)
```

```
## # A tibble: 6 × 6
##   `Country Name`           Year_Cat_sh mean_CO2_emissions_sh Year_Cat_fh
##   <chr>                    <chr>                       <dbl> <chr>
## 1 Burundi                  second_half                0.0373 first_half
## 2 Congo, Dem. Rep.         second_half                0.0405 first_half
## 3 Central African Republic second_half                0.0418 first_half
## 4 Somalia                  second_half                0.0491 first_half
## 5 Malawi                   second_half                0.0726 first_half
## 6 Niger                    second_half                0.0830 first_half
## # ℹ 2 more variables: mean_CO2_emissions_fh <dbl>, Diff <dbl>
```

```
bur_co2 = co2_data[co2_data$Country.Name == "Burundi",]
bur_co2_fh = bur_co2[bur_co2$Year_Cat == "first_half",]$CO2_emissions_metric_tons_per_capita
bur_co2_sh = bur_co2[bur_co2$Year_Cat == "second_half",]$CO2_emissions_metric_tons_per_capita
bur_co2_sh = head(bur_co2_sh, -1)

bur_co2_fh
```

```
##  [1] 0.03420990 0.04103790 0.03503344 0.03791029 0.03769864 0.03655957
##  [7] 0.03765277 0.03818455 0.03852310 0.03783061 0.04163193 0.03250987
## [13] 0.03280223 0.02495319 0.02241417
```

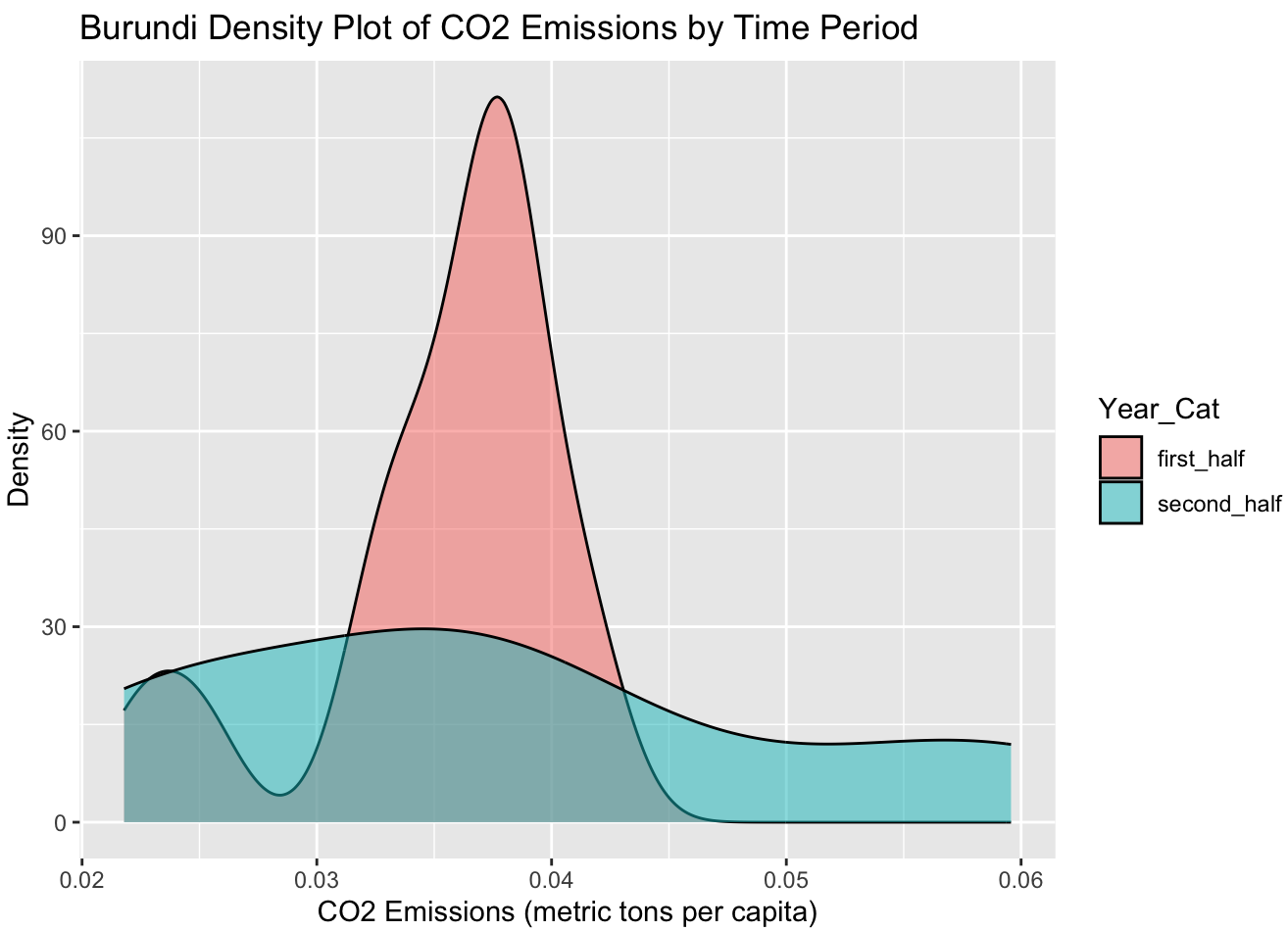
```
bur_co2_sh
```

```
##  [1] 0.02178952 0.02501897 0.02417992 0.02400307 0.02274563 0.03542391
##  [7] 0.03799811 0.03718042 0.03610003 0.03475970 0.03419362 0.04053809
## [13] 0.04742912 0.05785023 0.05957134
```

```
t.test(bur_co2_sh, bur_co2_fh, paired = TRUE, var.equal=FALSE, corr = FALSE,  alternative = "two.s
ided")
```

```
##
##  Paired t-test
##
## data:  bur_co2_sh and bur_co2_fh
## t = 0.155, df = 14, p-value = 0.879
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -0.008412366  0.009722966
## sample estimates:
## mean difference
##    0.0006552999
```

```
# Create a density plot for each time period
ggplot(bur_co2, aes(x = CO2_emissions_metric_tons_per_capita, fill = Year_Cat)) +
  geom_density(alpha = 0.5) +
  labs(title = "Burundi Density Plot of CO2 Emissions by Time Period",
       x = "CO2 Emissions (metric tons per capita)",
       y = "Density")
```




```
ethiopia_co2 = co2_data[co2_data$Country.Name == "Ethiopia",]
ethiopia_co2_fh = ethiopia_co2[ethiopia_co2$Year_Cat == "first_half",]$CO2_emissions_metric_tons_p
er_capita
ethiopia_co2_sh = ethiopia_co2[ethiopia_co2$Year_Cat == "second_half",]$CO2_emissions_metric_tons_
per_capita
ethiopia_co2_sh = head(ethiopia_co2_sh, -1)

ethiopia_co2_fh
```

```
##  [1] 0.04826218 0.04672047 0.02917044 0.03608923 0.04065788 0.04479567
##  [7] 0.04836754 0.04994653 0.05112099 0.04881251 0.05306431 0.06376946
## [13] 0.06432100 0.06864124 0.07083834
```

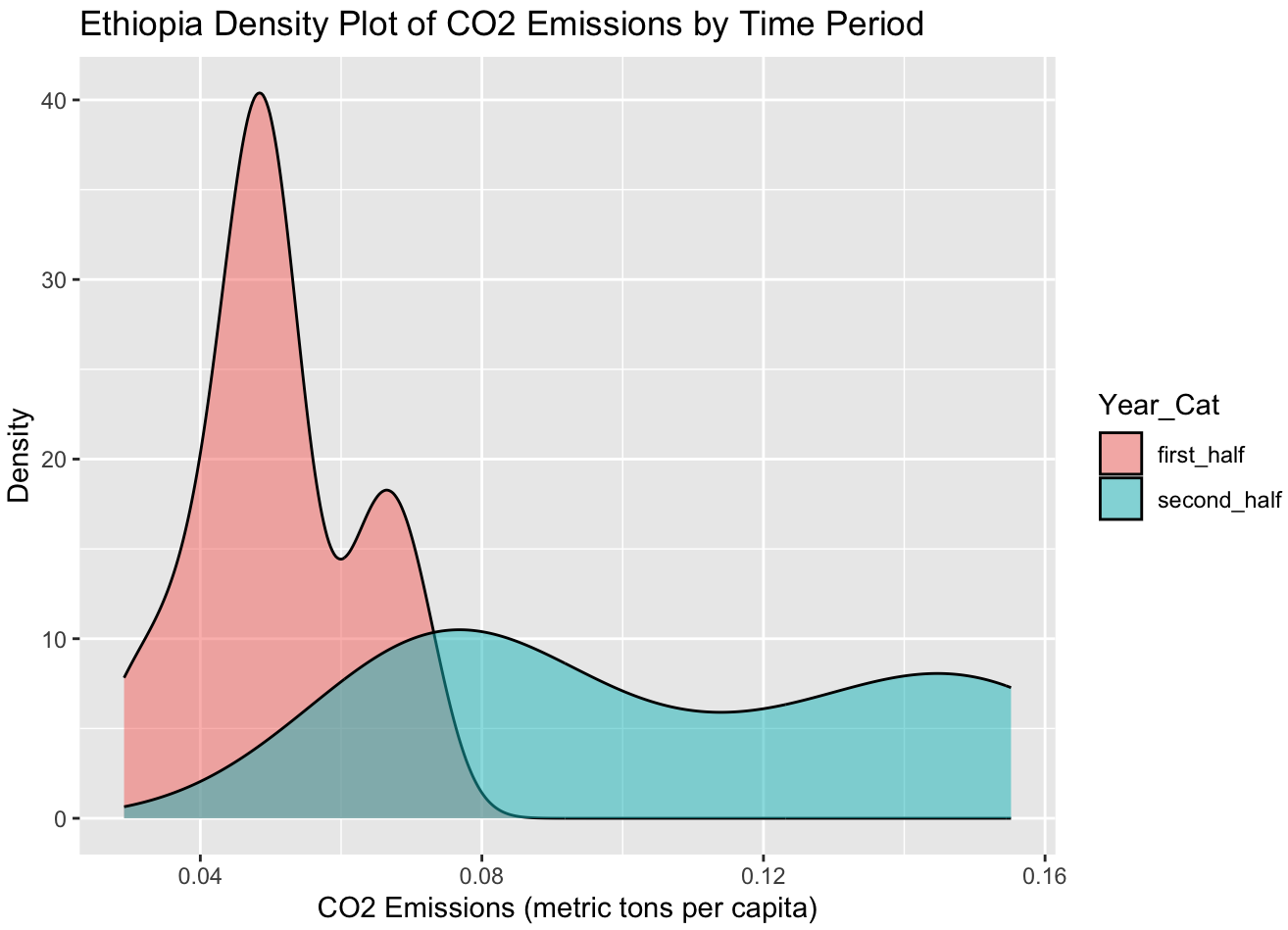
```
ethiopia_co2_sh
```

```
##  [1] 0.06518012 0.06782192 0.07272899 0.07690757 0.07532080 0.07253877
##  [7] 0.08224429 0.09061815 0.10496850 0.12522010 0.12731589 0.14480703
## [13] 0.14679021 0.15323033 0.15516919
```

```
t.test(ethiopia_co2_sh, ethiopia_co2_fh, paired = TRUE, var.equal=FALSE, corr = FALSE,  alternativ
e = "two.sided")
```

```
##
##  Paired t-test
##
## data:  ethiopia_co2_sh and ethiopia_co2_fh
## t = 8.2557, df = 14, p-value = 9.478e-07
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  0.03929417 0.06687703
## sample estimates:
## mean difference
##       0.0530856
```

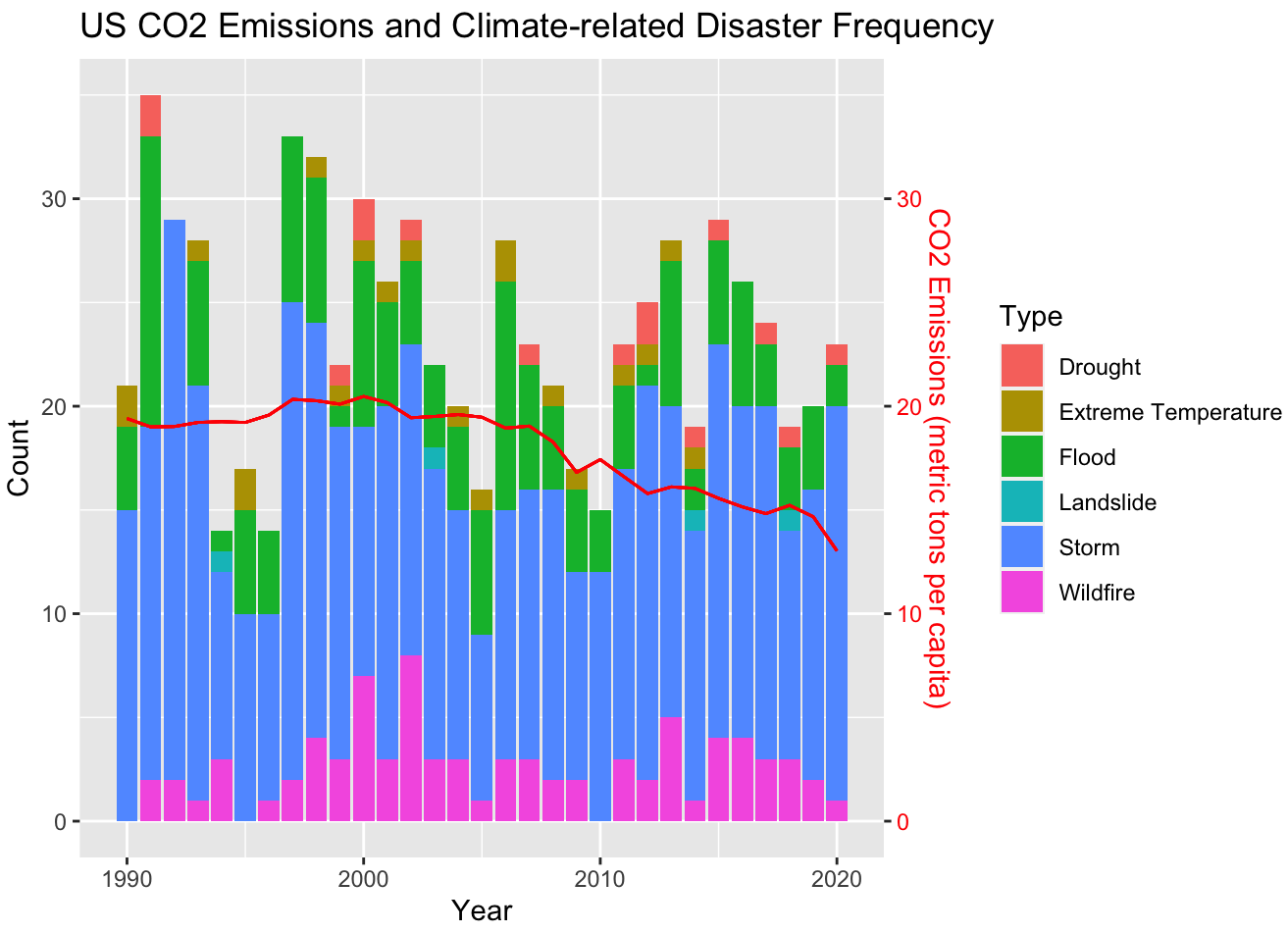
```
# Create a density plot for each time period
ggplot(ethiopia_co2, aes(x = CO2_emissions_metric_tons_per_capita, fill = Year_Cat)) +
  geom_density(alpha = 0.5) +
  labs(title = "Ethiopia Density Plot of CO2 Emissions by Time Period",
       x = "CO2 Emissions (metric tons per capita)",
       y = "Density")
```



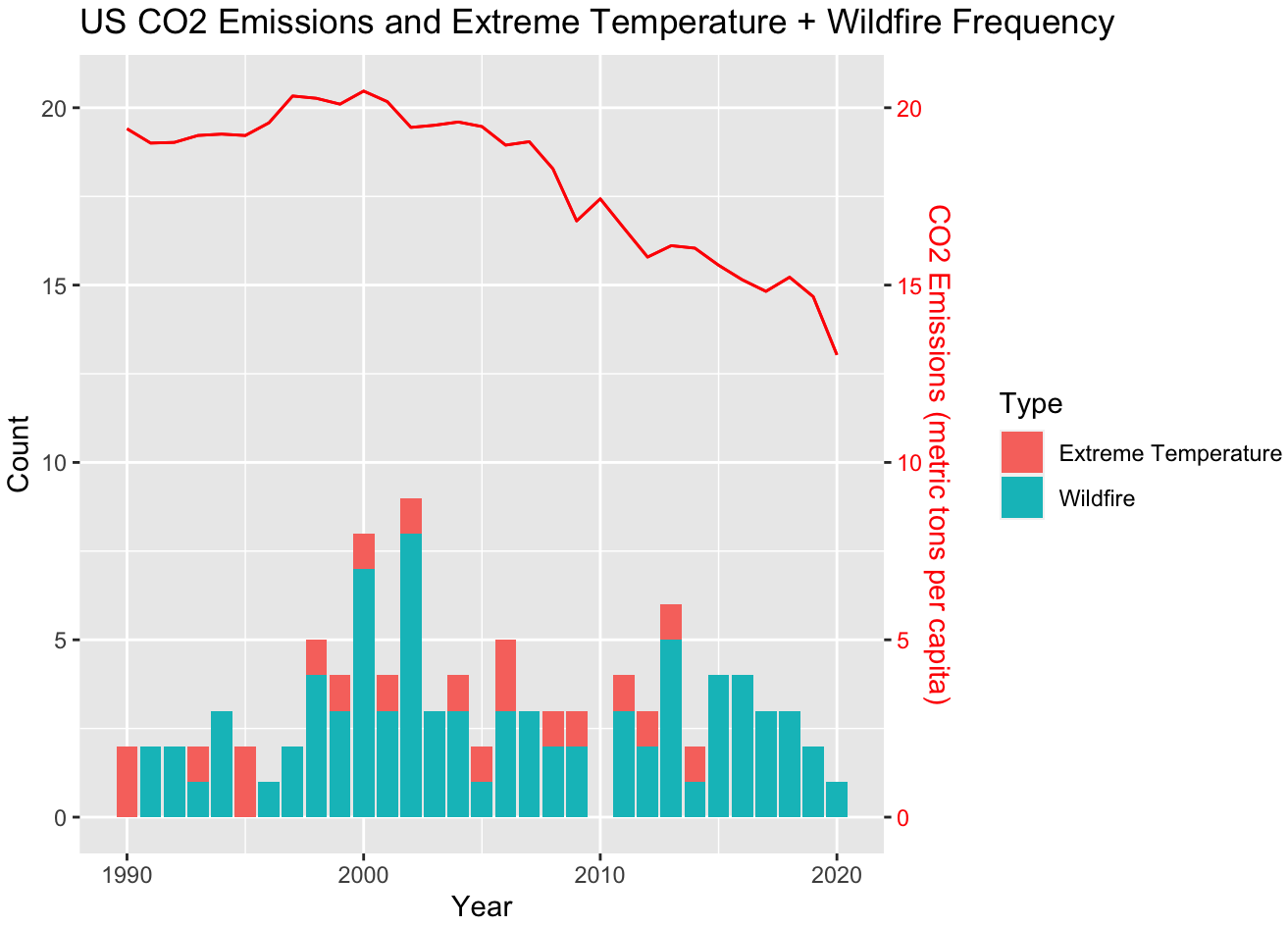
```
climate_data = read.csv("Climate-related_Disasters_Frequency.csv")
climate_data = climate_data %>% select(-one_of("ObjectId", "ISO2", "ISO3", "Unit", "Source", "CTS_
Code", "CTS_Name", "CTS_Full_Descriptor"))

climate_data = climate_data %>% pivot_longer(cols = starts_with("F"), names_to = "Year", values_to
= "count")
climate_data$Type <- "Drought"
climate_data$Type[climate_data$Indicator == "Climate related disasters frequency, Number of Disast
ers: Extreme temperature"] <- "Extreme Temperature"
climate_data$Type[climate_data$Indicator == "Climate related disasters frequency, Number of Disast
ers: Flood"] <- "Flood"
climate_data$Type[climate_data$Indicator == "Climate related disasters frequency, Number of Disast
ers: Landslide"] <- "Landslide"
climate_data$Type[climate_data$Indicator == "Climate related disasters frequency, Number of Disast
ers: Storm"] <- "Storm"
climate_data$Type[climate_data$Indicator == "Climate related disasters frequency, Number of Disast
ers: TOTAL"] <- "TOTAL"
climate_data$Type[climate_data$Indicator == "Climate related disasters frequency, Number of Disast
ers: Wildfire"] <- "Wildfire"
climate_data = climate_data %>% select("Country", "Type", "Year", "count")
climate_data[is.na(climate_data)] = 0

head(climate_data)
```

```
## # A tibble: 6 × 4
##   Country                     Type    Year  count
##   <chr>                       <chr>   <chr> <int>
## 1 Afghanistan, Islamic Rep. of Drought F1980     0
## 2 Afghanistan, Islamic Rep. of Drought F1981     0
## 3 Afghanistan, Islamic Rep. of Drought F1982     0
## 4 Afghanistan, Islamic Rep. of Drought F1983     0
## 5 Afghanistan, Islamic Rep. of Drought F1984     0
## 6 Afghanistan, Islamic Rep. of Drought F1985     0
```

```
us_climate = climate_data[climate_data$Country == "United States",]
us_climate_ex_total = subset(us_climate, !(Type %in% "TOTAL"))
us_climate_ex_total$Year = gsub("F", "", as.character(us_climate_ex_total$Year))
us_climate_ex_total$Year = as.numeric(us_climate_ex_total$Year)
us_data = us_climate_ex_total %>% inner_join(us_co2, by=c("Year")) %>% select("Country", "Type",
"Year", "count", "CO2_emissions_metric_tons_per_capita", "Year_Cat")

us_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(

    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
  ) +
  theme(axis.title.y = element_text(color = "black"),
    axis.title.y.right = element_text(color = "red"),
    axis.text.y.right = element_text(colour = "red")) +
  labs(title = "US CO2 Emissions and Climate-related Disaster Frequency")
```



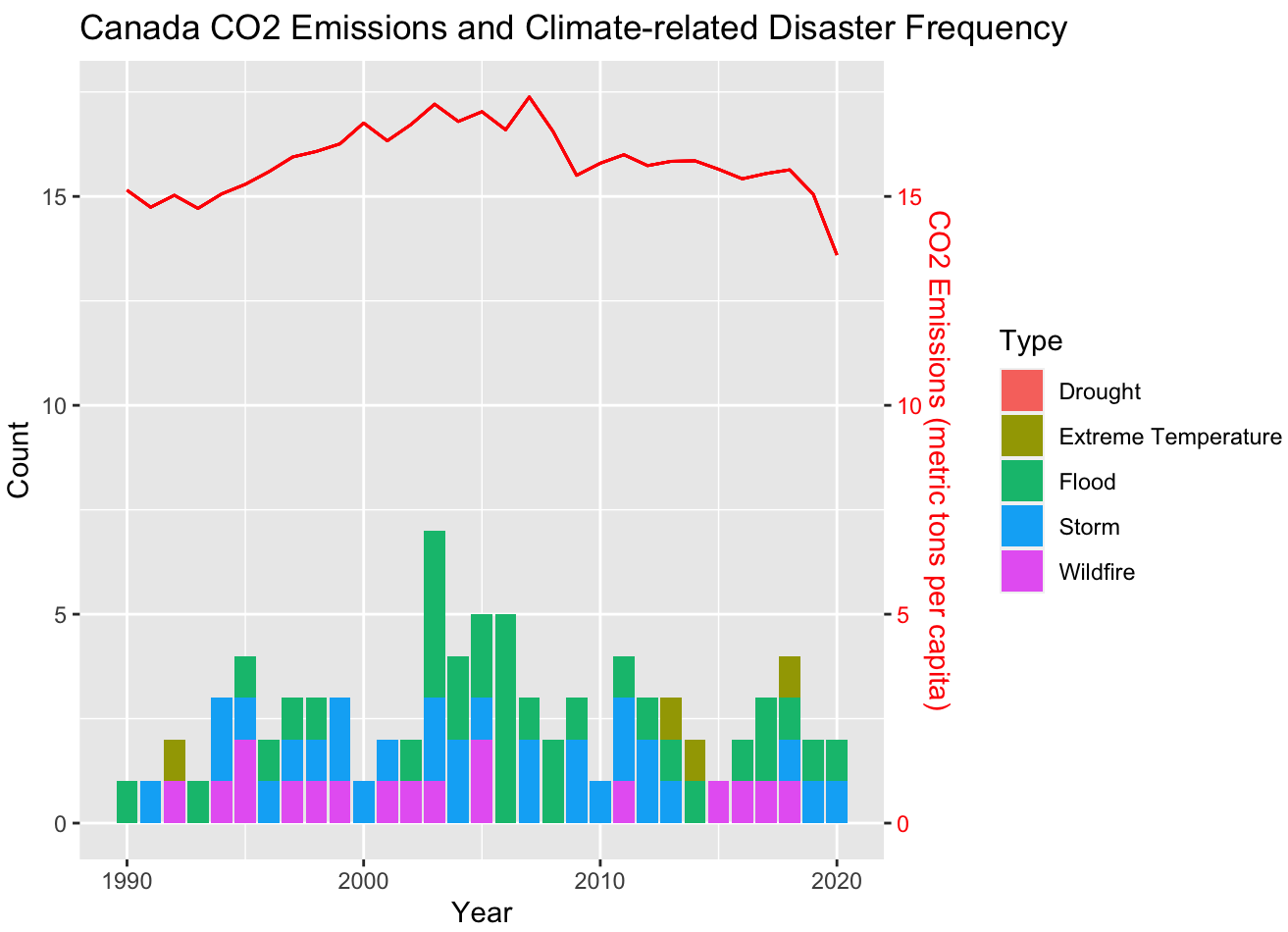
```
us_climate_extreme_temp_wildfire = subset(us_climate, (Type %in% c("Extreme Temperature", "Wildfir
e")))
us_climate_extreme_temp_wildfire$Year = gsub("F", "", as.character(us_climate_extreme_temp_wildfir
e$Year))
us_climate_extreme_temp_wildfire$Year = as.numeric(us_climate_extreme_temp_wildfire$Year)

us_data = us_climate_extreme_temp_wildfire %>%
            inner_join(us_co2, by=c("Year")) %>%
            select("Type", "Year", "count", "CO2_emissions_metric_tons_per_capita")
us_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(

    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
  ) +
  theme(axis.title.y = element_text(color = "black"),
    axis.title.y.right = element_text(color = "red"),
    axis.text.y.right = element_text(colour = "red")) +
  labs(title = "US CO2 Emissions and Extreme Temperature + Wildfire Frequency")
```



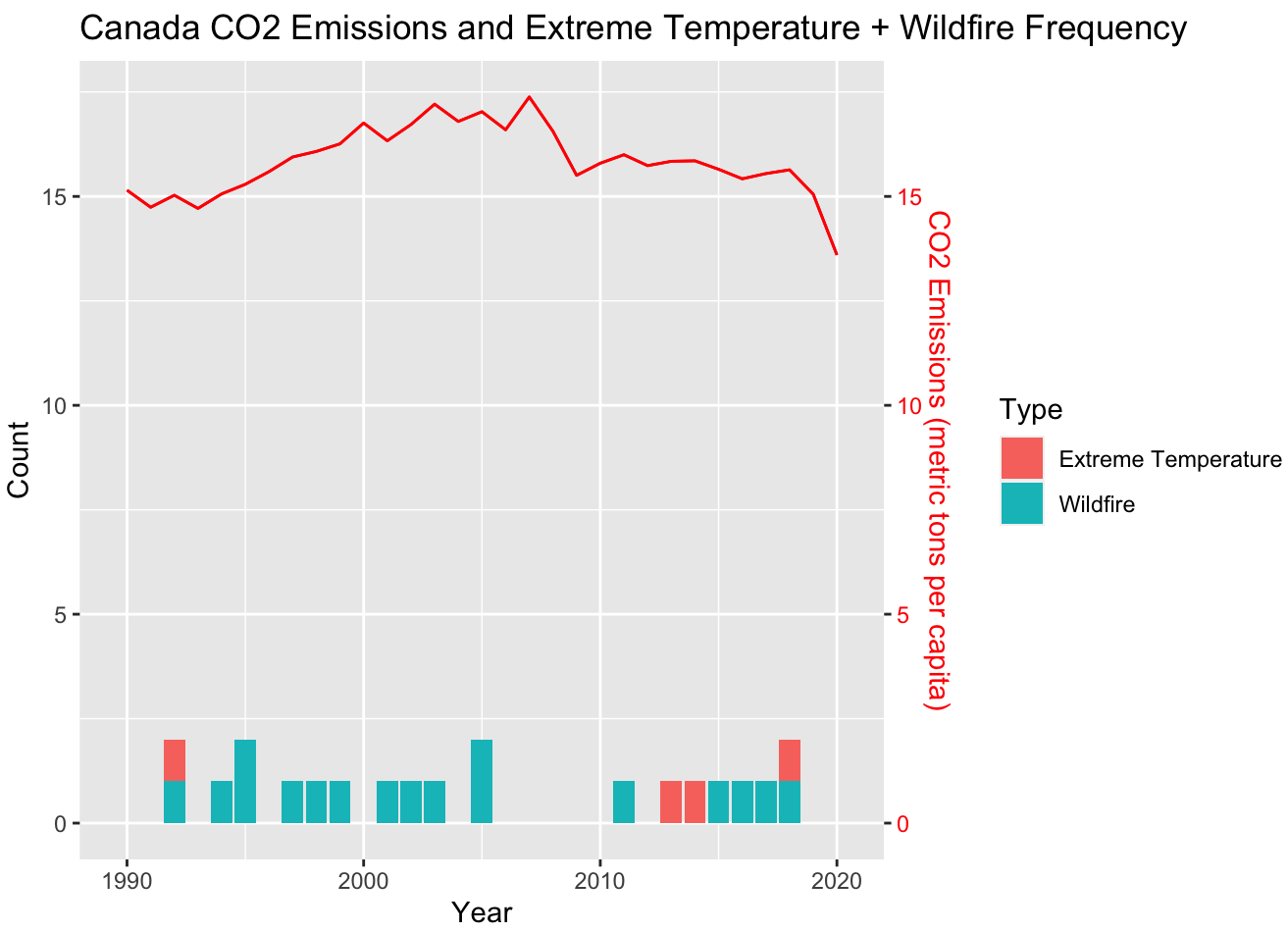
```
canada_co2 = co2_data[co2_data$Country.Name == "Canada",]
canada_climate = climate_data[climate_data$Country == "Canada",]
canada_climate
```

```
## # A tibble: 258 × 4
##    Country Type    Year  count
##    <chr>   <chr>   <chr> <int>
##  1 Canada  Drought F1980     0
##  2 Canada  Drought F1981     0
##  3 Canada  Drought F1982     0
##  4 Canada  Drought F1983     0
##  5 Canada  Drought F1984     1
##  6 Canada  Drought F1985     0
##  7 Canada  Drought F1986     0
##  8 Canada  Drought F1987     0
##  9 Canada  Drought F1988     1
## 10 Canada  Drought F1989     0
## # ℹ 248 more rows
```

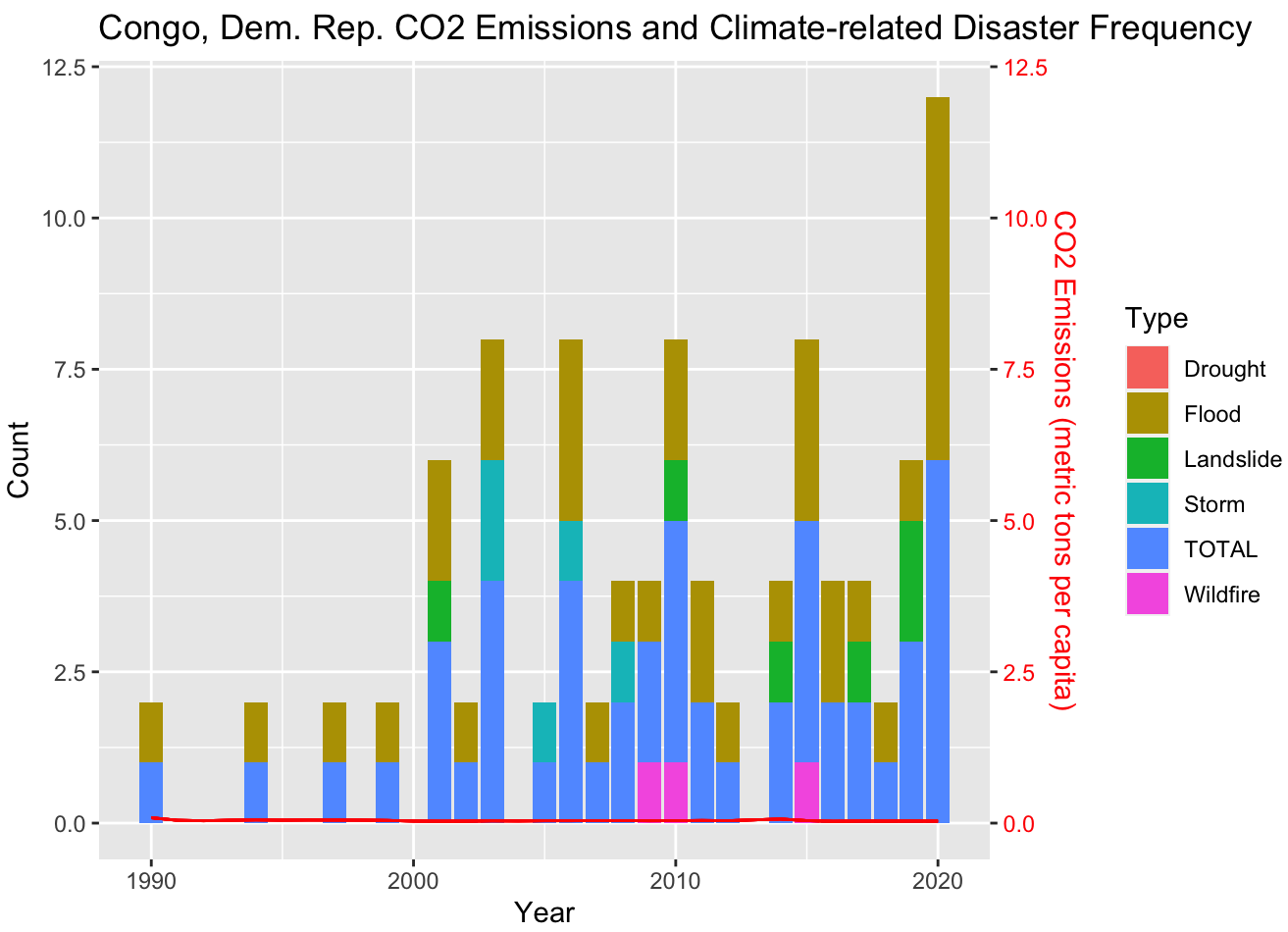
```
can_climate_ex_total = subset(canada_climate, !(Type %in% "TOTAL"))
can_climate_ex_total$Year = gsub("F", "", as.character(can_climate_ex_total$Year))
can_climate_ex_total$Year = as.numeric(can_climate_ex_total$Year)
can_data = can_climate_ex_total %>% inner_join(canada_co2, by=c("Year")) %>% select("Country", "Ty
pe", "Year", "count", "CO2_emissions_metric_tons_per_capita", "Year_Cat")

can_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(
    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
    ) +
  theme(axis.title.y = element_text(color = "black"),
        axis.title.y.right = element_text(color = "red"),
        axis.text.y.right = element_text(colour = "red")) +
  labs(title = "Canada CO2 Emissions and Climate-related Disaster Frequency")
```



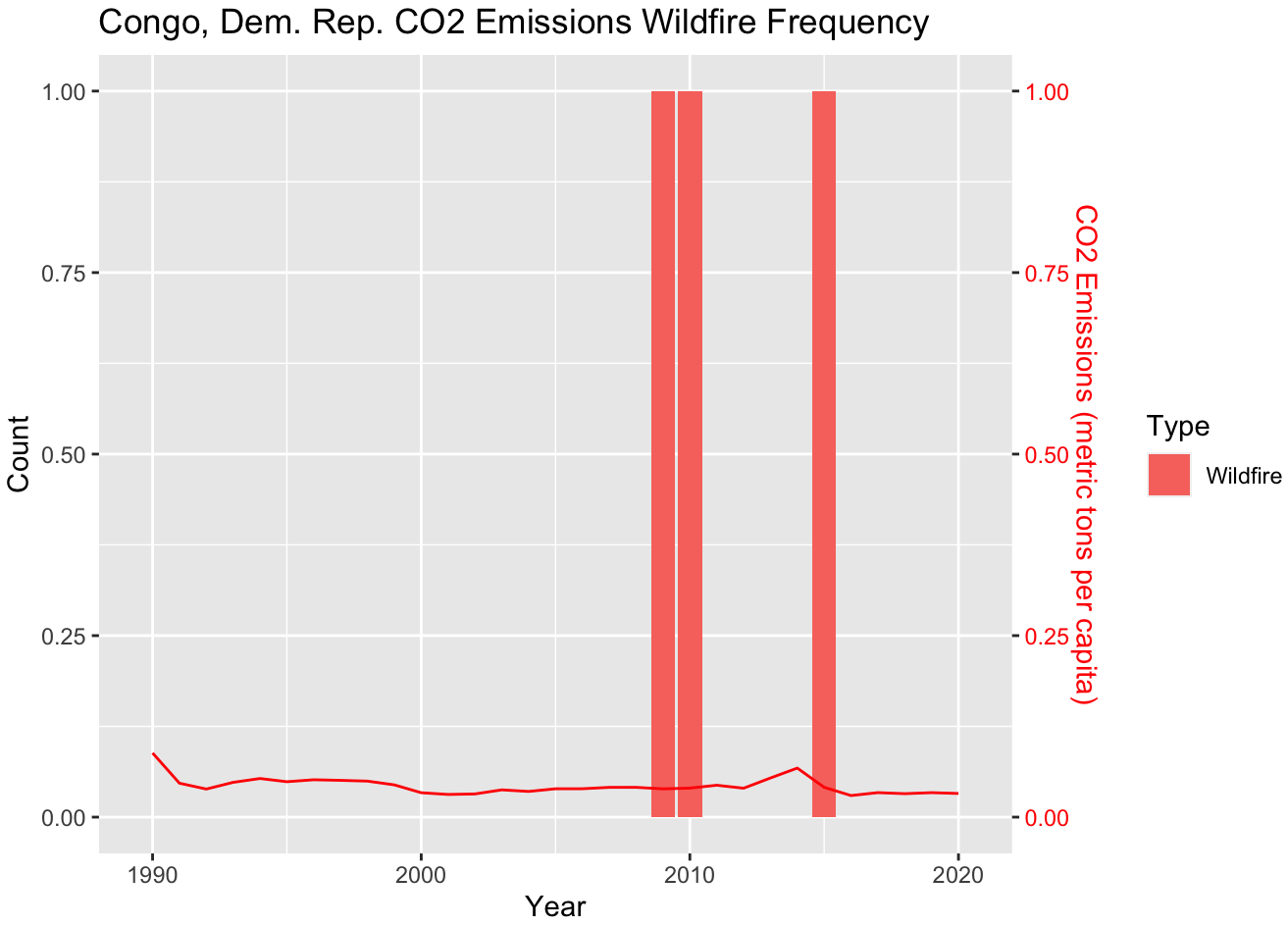

```
can_climate_extreme_temp_wildfire = subset(canada_climate, (Type %in% c("Extreme Temperature", "Wi
ldfire")))
can_climate_extreme_temp_wildfire$Year = gsub("F", "", as.character(can_climate_extreme_temp_wildf
ire$Year))
can_climate_extreme_temp_wildfire$Year = as.numeric(can_climate_extreme_temp_wildfire$Year)

can_data = can_climate_extreme_temp_wildfire %>%
            inner_join(canada_co2, by=c("Year")) %>%
            select("Type", "Year", "count", "CO2_emissions_metric_tons_per_capita")
can_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(

    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
  ) +
  theme(axis.title.y = element_text(color = "black"),
    axis.title.y.right = element_text(color = "red"),
    axis.text.y.right = element_text(colour = "red")) +
  labs(title = "Canada CO2 Emissions and Extreme Temperature + Wildfire Frequency")
```



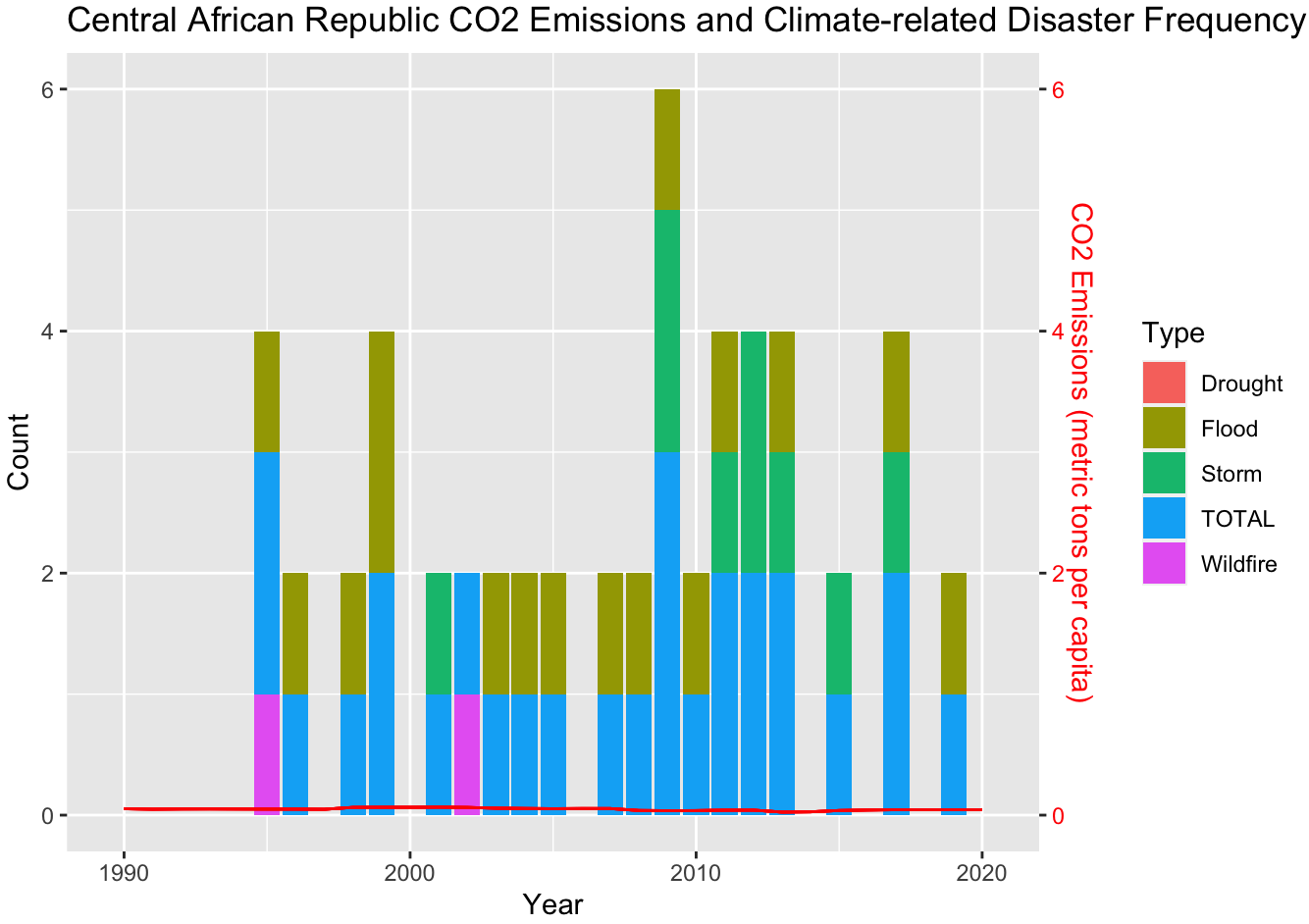

```
congo_co2 = co2_data[co2_data$Country.Name == "Congo, Dem. Rep.",]
congo_climate = climate_data[climate_data$Country == "Congo, Dem. Rep. of the",]
congo_climate$Year = gsub("F", "", as.character(congo_climate$Year))
congo_climate$Year = as.numeric(congo_climate$Year)

congo_data = congo_climate %>% inner_join(congo_co2, by=c("Year")) %>% select("Country", "Type",
"Year", "count", "CO2_emissions_metric_tons_per_capita", "Year_Cat")

congo_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(
    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
    ) +
  theme(axis.title.y = element_text(color = "black"),
        axis.title.y.right = element_text(color = "red"),
        axis.text.y.right = element_text(colour = "red")) +
  labs(title = "Congo, Dem. Rep. CO2 Emissions and Climate-related Disaster Frequency")
```



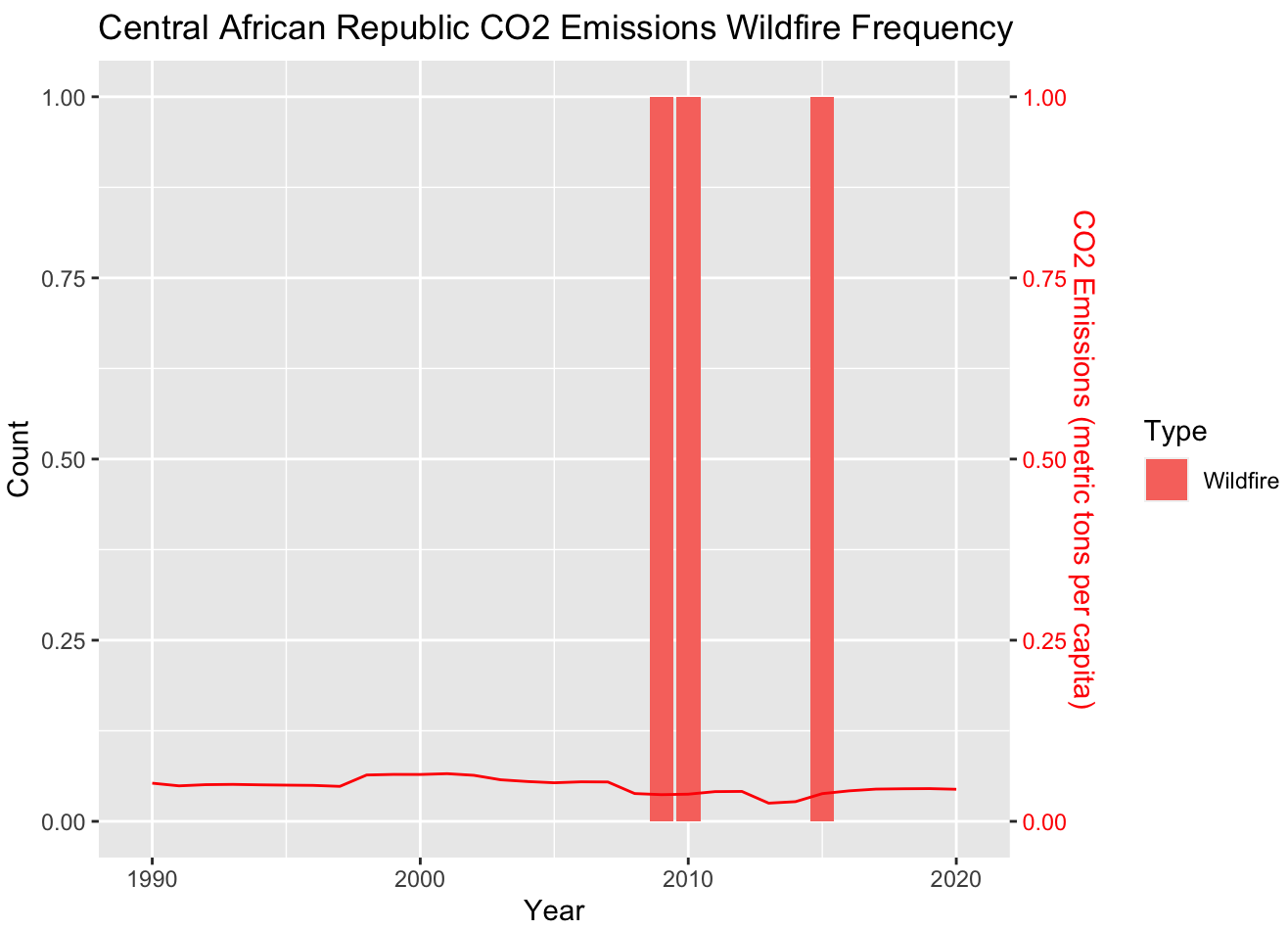

```
congo = subset(congo_climate, (Type %in% c("Extreme Temperature", "Wildfire")))
congo$Year = gsub("F", "", as.character(congo$Year))
congo$Year = as.numeric(congo$Year)

congo_data = congo %>%
             inner_join(congo_co2, by=c("Year")) %>%
             select("Type", "Year", "count", "CO2_emissions_metric_tons_per_capita")
congo_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(

    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
  ) +
  theme(axis.title.y = element_text(color = "black"),
    axis.title.y.right = element_text(color = "red"),
    axis.text.y.right = element_text(colour = "red")) +
  labs(title = "Congo, Dem. Rep. CO2 Emissions Wildfire Frequency")
```




```
car_co2 = co2_data[co2_data$Country.Name == "Central African Republic",]
car_climate = climate_data[climate_data$Country == "Central African Rep.",]
car_climate$Year = gsub("F", "", as.character(car_climate$Year))
car_climate$Year = as.numeric(car_climate$Year)

car_data = car_climate %>% inner_join(car_co2, by=c("Year")) %>% select("Country", "Type", "Year",
"count", "CO2_emissions_metric_tons_per_capita", "Year_Cat")

car_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(
    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
    ) +
  theme(axis.title.y = element_text(color = "black"),
        axis.title.y.right = element_text(color = "red"),
        axis.text.y.right = element_text(colour = "red")) +
  labs(title = "Central African Republic CO2 Emissions and Climate-related Disaster Frequency")
```




```
car = subset(car_climate, (Type %in% c("Extreme Temperature", "Wildfire")))
car$Year = gsub("F", "", as.character(car$Year))
car$Year = as.numeric(car$Year)

car_data = congo %>%
            inner_join(car_co2, by=c("Year")) %>%
            select("Type", "Year", "count", "CO2_emissions_metric_tons_per_capita")
car_data %>%
  ggplot(aes(x=Year, y=count, fill=Type)) +
  geom_bar(stat="identity") +
  geom_line(aes(x=Year, y=CO2_emissions_metric_tons_per_capita), color="red") +
  scale_y_continuous(

    # Features of the first axis
    name = "Count",

    # Add a second axis and specify its features
    sec.axis = sec_axis(~., name="CO2 Emissions (metric tons per capita)")
  ) +
  theme(axis.title.y = element_text(color = "black"),
    axis.title.y.right = element_text(color = "red"),
    axis.text.y.right = element_text(colour = "red")) +
  labs(title = "Central African Republic CO2 Emissions Wildfire Frequency")
```


Central African Republic CO2 Emissions Wildfire Frequency
Count
1.00
0.75
0.50
0.25
0.00
1990
2000
2010
2020
Year
CO2 Emissions (metric tons per capita)
Type
Wildfire