SPEC REU R Resources: Data Management for Visualization

Alix Ziff, Miriam Barnum, Abigail Longstreth, Claudia Salas Gimenez, Ben Graham

February 2025

Welcome to the final module on data management! In previous modules, we focused on essential functions for handling and cleaning data, emphasizing effective dataset manipulation. This module takes a step further—not only will we refine our ability to aggregate and summarize data, but we will also explore the research-driven motivations behind these operations.

The purpose of research extends beyond running functions or generating statistics—it is about uncovering patterns, testing hypotheses, and effectively communicating findings. Data management plays a crucial role in shaping raw data into structured insights that allow us to address key research questions. Researchers rarely analyze raw data in its original form; instead, they transform it into formats that facilitate meaningful visualizations and statistical analyses.

In this walkthrough, we'll work with the IDC Powersharing dataset, which examines political powersharing and its effects on democratic stability. Our objective is to answer two research questions:

1. How do global trends in natural resource rents and trade volume evolve over time?
2. Which regions rely most heavily on natural resources, and how has this reliance changed over time?

To address these questions, we will produce two plots: one comparing trends in global natural resource rents and trade volume over time, and another highlighting regional variations in natural resource dependence. These visualizations will help us analyze long-term shifts and regional differences, providing insights into broader economic and political trends. Let's get started!

Initial Setup

Before we begin, ensure that your R environment is set up properly. Load the necessary libraries and import the `IDC_training_2022.rds` dataset.

For reference, the IDC Powersharing Dataset was introduced in the research paper *"De Jure Powersharing 1975–2019: Updating the Inclusion, Dispersion, and Constraints Dataset"* by Ziff, Barnum, Abadeer, Chu, Jao, Zaragoza, and Graham (2024).

You can find the research paper and the online appendix here.

```r
# Set working directory
#setwd("YourFolderPath")

# Load required libraries
library(dplyr)
library(ggplot2)
library(tidyr)

# Load the data
idc_controls <- readRDS("IDC_training_2022.rds")
```

Why Do We Aggregate Data?

Unlike previous modules, this assignment is designed to help you think critically about the research process behind data management. When analyzing data, raw numbers alone rarely provide clear insights. The goal of research is to uncover patterns and trends, often through visualizations that make complex data easier to interpret. Aggregating data allows us to summarize information in a meaningful way, helping us compare trends across different categories.

In this walkthrough, we aim to create two visualizations to answer key research questions:

1. How do global trends in natural resource rents and trade volume evolve over time?
2. Which regions rely most heavily on natural resources, and how has this reliance changed over time?

Before we jump into creating our visualizations, let's take a step back and think about why we need to aggregate data in the first place.

1. If we want to track how global natural resource rents and trade volume have changed over time, what kind of information do we need? What type of visualization would best display this trend?
    - To analyze global patterns, we need to summarize the data across all countries for each year. Instead of displaying data for every individual country, we can calculate global averages to help reduce noise from country-specific fluctuations and reveal broader trends.
    - Looking at a single variable in isolation provides only part of the picture. Researchers often compare multiple economic or political indicators side by side to understand how they evolve together, and for this assignment we are looking at natural resource rents and trade volume.
    - A line plot is the most effective way to visualize these trends over time, allowing us to see whether reliance on natural resources and trade volume have increased or decreased globally.
2. If we want to compare regional variations in natural resource dependence? Where should we start?
    - Sometimes, global averages alone aren't enough. Instead of aggregating data at the global level, we may want to compare regions to see which parts of the world depend most on natural resources. In this case, we need to group countries by region and compute regional averages to understand how different parts of the world vary on natural resource dependence.

Through this process, you'll gain a deeper understanding of how data management supports research by transforming raw data into meaningful insights.

How do global trends in natural resource rents and trade volume evolve over time?

Global Averages (Univariate Analysis)

To identify large-scale trends, we first need to calculate global averages. By aggregating data across all countries for each year, we can examine how natural resource rents have changed over time at a global level. This step is important because individual country data may fluctuate significantly due to country-specific factors, making it difficult to identify broader patterns. By summarizing the data, we smooth out these variations and focus on the overall trend.

We use the `group_by()` function to organize the data into subgroups based on a specific variable (e.g., group data by year or region) and `summarise()` to compute summary statistics (e.g., mean, median) for each group. This gives us a single data point per year, making it easier to compare trends over time.

```
# Calculating global averages for 'natresource_rents_WDI'
rents_global <- idc_controls %>%
  group_by(year) %>%
  summarise(natresource_rents_mean = mean(natresource_rents_WDI, na.rm = T))
```

```r
# View the resulting dataset
head(rents_global)
```

```
## # A tibble: 6 x 2
##    year natresource_rents_mean
##   <dbl>                  <dbl>
## 1  1975                   9.07
## 2  1976                   8.59
## 3  1977                   8.98
## 4  1978                   8.18
## 5  1979                   9.89
## 6  1980                  10.6
```

Note that `na.rm = TRUE` instructs R to exclude missing values from calculations, preventing them from skewing the averages. Without this, missing data from certain countries could distort the results. However, even after removing missing values, the trends we observe may still be influenced by regions with lower data availability, as their absence can affect the overall patterns.

As social scientists, we can usually anticipate significant regional variations, particularly when analyzing factors such as economic indicators. Thus, to examine regional averages, we can also include more variables in the `group_by()` function, to subgroup not only by time, but also by region.

Comparing Multiple Variables (Multivariate Analysis)

While analyzing global trends is useful, researchers often need to compare trends across multiple variables to understand how they interact. Looking at a single variable in isolation provides only part of the picture—by examining multiple variables, we can gain a deeper understanding of how different economic factors evolve together.

For this assignment, let's say that we want to see how natural resource rents compare to trade volumes over time. To do this, instead of summarizing each variable separately, the `across()` function within `summarise()` allows us to apply the summary statistic in `summarise()` to multiple variables at once, making data manipulation more efficient.

Note: We use the `c()` function to combine the different variables into a vector, allowing `across()` to apply the same function to all selected columns at once.

```r
# Calculating global averages for 'natresource_rents_WDI' and 'trade_WDI'
trade_rents_global <- idc_controls %>%
  group_by(year) %>%
  summarise(across(c(natresource_rents_WDI, trade_WDI),
                   ~mean(., na.rm = TRUE),
                   .names = "{.col}_mean"))

# View the resulting dataset
head(trade_rents_global)
```

```
## # A tibble: 6 x 3
##    year natresource_rents_WDI_mean trade_WDI_mean
##   <dbl>                      <dbl>          <dbl>
## 1  1975                       9.07           62.6
## 2  1976                       8.59           63.8
## 3  1977                       8.98           66.4
## 4  1978                       8.18           66.1
## 5  1979                       9.89           70.0
## 6  1980                      10.6            76.5
```

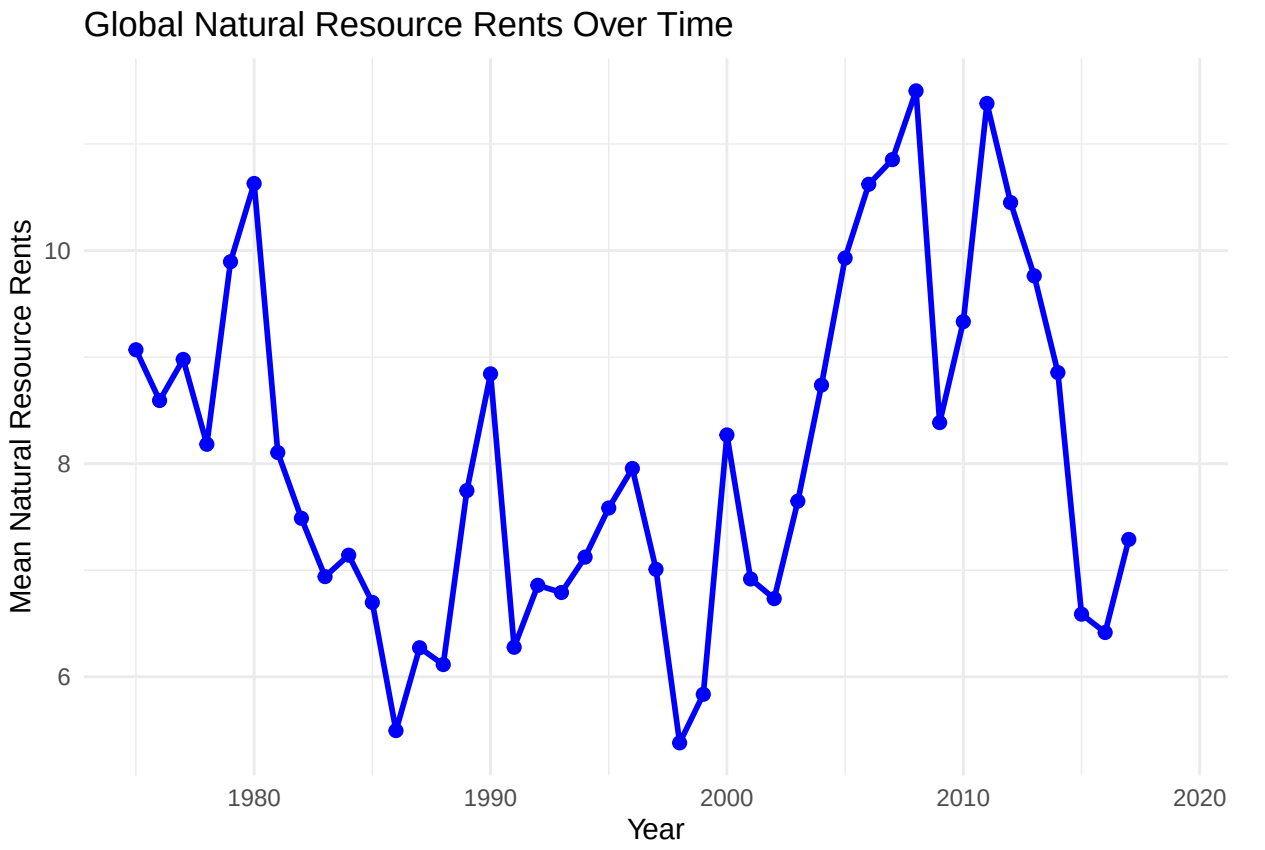
The `.names = "{.col}_mean"` function ensures the new column names clearly indicate the applied statistic. In this case, `natresource_rents_WDI` and `trade_WDI` are summarized, so the new column names will be `natresource_rents_WDI_mean` and `trade_WDI_mean`, making it clear that these represent the mean values.

Create Line Plots

Now, let's create two separate line plots to illustrate how trade volume and natural resource rents have evolved over time globally.

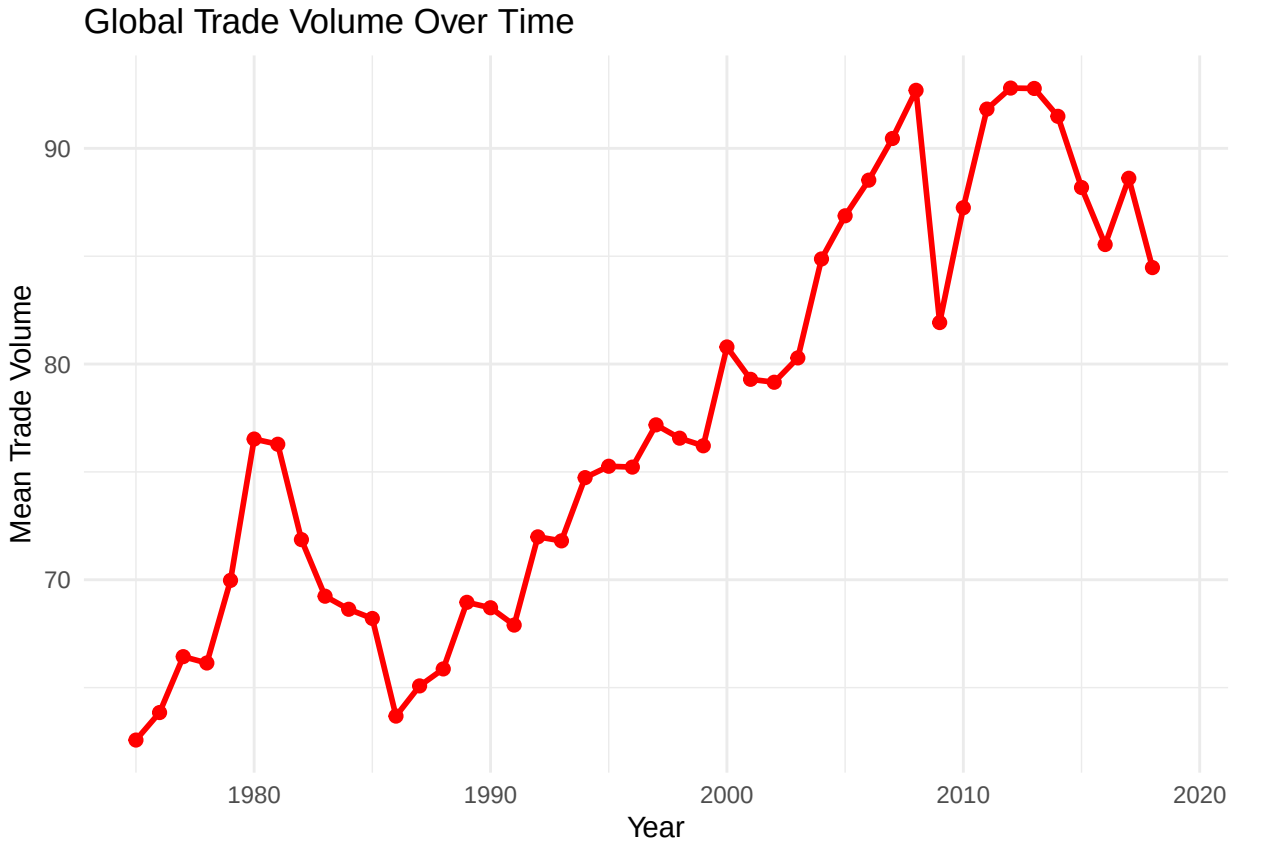
Note: As a rule of thumb, if you're plotting two different units using the same X and Y variables, it can be a single plot. If you're plotting with two different dependent variables or two different independent variables, it should (usually) be plotted in two separate plots.

```
# Line plot for natural resource rents over time
ggplot(trade_rents_global,
       aes(x = year, y = natresource_rents_WDI_mean)) +
  geom_line(color = "blue", size = 1) +
  geom_point(color = "blue", size = 2) +
  labs(title = "Global Natural Resource Rents Over Time",
       x = "Year",
       y = "Mean Natural Resource Rents") +
  theme_minimal()
```




```
# Line plot for trade volume over time
ggplot(trade_rents_global,
       aes(x = year, y = trade_WDI_mean)) +
  geom_line(color = "red", size = 1) +
  geom_point(color = "red", size = 2) +
  labs(title = "Global Trade Volume Over Time",
```

```
       x = "Year",
       y = "Mean Trade Volume") +
  theme_minimal()
```



Simplifying Data to Decade-Level Aggregation

Sometimes, reliable yearly data is patchy or incomplete. Aggregating by decade can make sense if some countries only have data every few years. It can also help mitigate the effect of sparse data in certain periods by averaging across a broader window.

To group data into decades, we create a new variable using the modulo operator (`%%`). The modulo operator calculates the remainder of a division. For instance, `1977 %% 10 = 7` 1977 divided by 10 leaves a remainder of 7). Subtracting this remainder from the year yields the start of the decade: `1977 - 1977 %% 10 = 1970`.

Using this approach, let's create a decade variable and summarize data by decades as an exercise:

```
# Calculating global averages for 'natresource_rents_WDI' and 'trade_WDI'
trade_rents_decade <- idc_controls %>%
  mutate(decade = year - year %% 10) %>%
  group_by(decade) %>%
  summarise(across(c(natresource_rents_WDI, trade_WDI),
                   ~mean(., na.rm = TRUE),
                   .names = "{.col}_mean"))

# View the resulting dataset
head(trade_rents_decade)
```

```
## # A tibble: 5 x 3
##   decade natresource_rents_WDI_mean trade_WDI_mean
```

```
##    <dbl>                    <dbl>          <dbl>
## 1   1970                     8.94           65.8
## 2   1980                     7.24           69.4
## 3   1990                     6.95           73.8
## 4   2000                     8.97           84.5
## 5   2010                     8.77           89.6
```

This produces a single data point per decade, which you can then plot or analyze as a smooth series. We provide this code as a reference in case you want to use it. However, for our analysis, because we have complete year-level data, it does not make sense to plot or conduct any further analysis at the decade level.

Which regions rely most heavily on natural resources?

Regional averages for natural resource rents

Understanding global trends is valuable, but significant regional differences often exist. Let's say that for our research we're interesting in looking at which regions rely most heavily on natural resources and how do these trends change over time. To do so, we need to calculate regional averages for natural resource rents using the original data and grouping by both region and year.

```
# Compute regional averages for resource rents by year
rents_regional <- idc_controls %>%
  group_by(region, year) %>%
  summarise(natresource_rents_mean = mean(natresource_rents_WDI, na.rm = TRUE)) %>%
  filter(!is.na(region))
  ## Remove rows where region is NA

# Preview the results
head(rents_regional)
```

```
## # A tibble: 6 x 3
## # Groups:   region [1]
##   region               year natresource_rents_mean
##   <chr>               <dbl>                  <dbl>
## 1 East Asia & Pacific  1975                   7.33
## 2 East Asia & Pacific  1976                   7.39
## 3 East Asia & Pacific  1977                   7.63
## 4 East Asia & Pacific  1978                   6.78
## 5 East Asia & Pacific  1979                  10.3
## 6 East Asia & Pacific  1980                  10.4
```

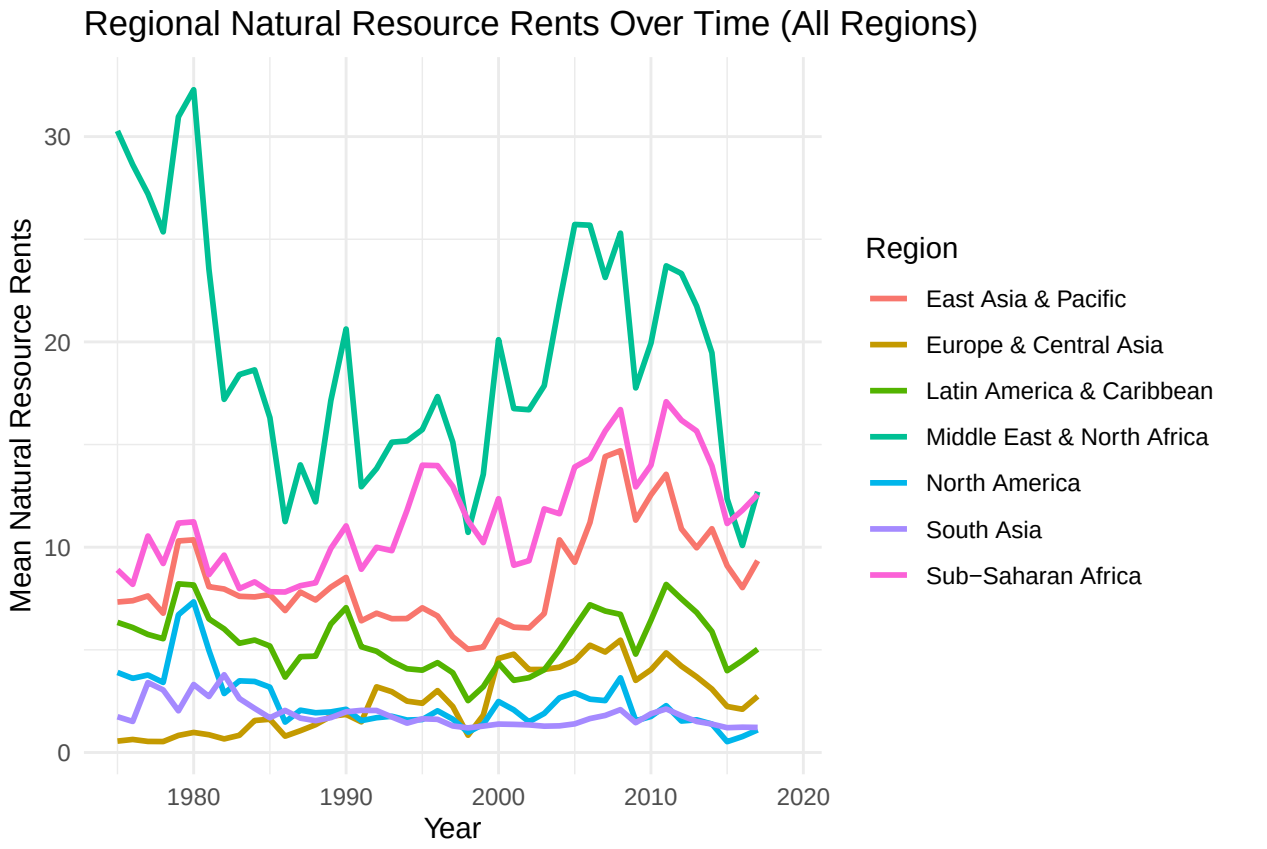
Each row now corresponds to a region-year combination with mean natural resource rents values.

Create Line Plot of All Regions

Now that we have restructured the data, it's time to create our final visualization. The goal is to illustrate which regions rely most heavily on natural resources and how this reliance has changed over time using a line plot. Remember to challenge yourself to customize your visualization to produce a publication-quality figure.

```
# Create line plot of all regions
ggplot(rents_regional,
       aes(x = year, y = natresource_rents_mean, color = region)) +
  geom_line(size = 1) +
  labs(title = "Regional Natural Resource Rents Over Time (All Regions)",
       x = "Year",
       y = "Mean Natural Resource Rents",
```

```
       color = "Region") +
  theme_minimal()
```



Create Line Plot Focusing on a Few Key Regions

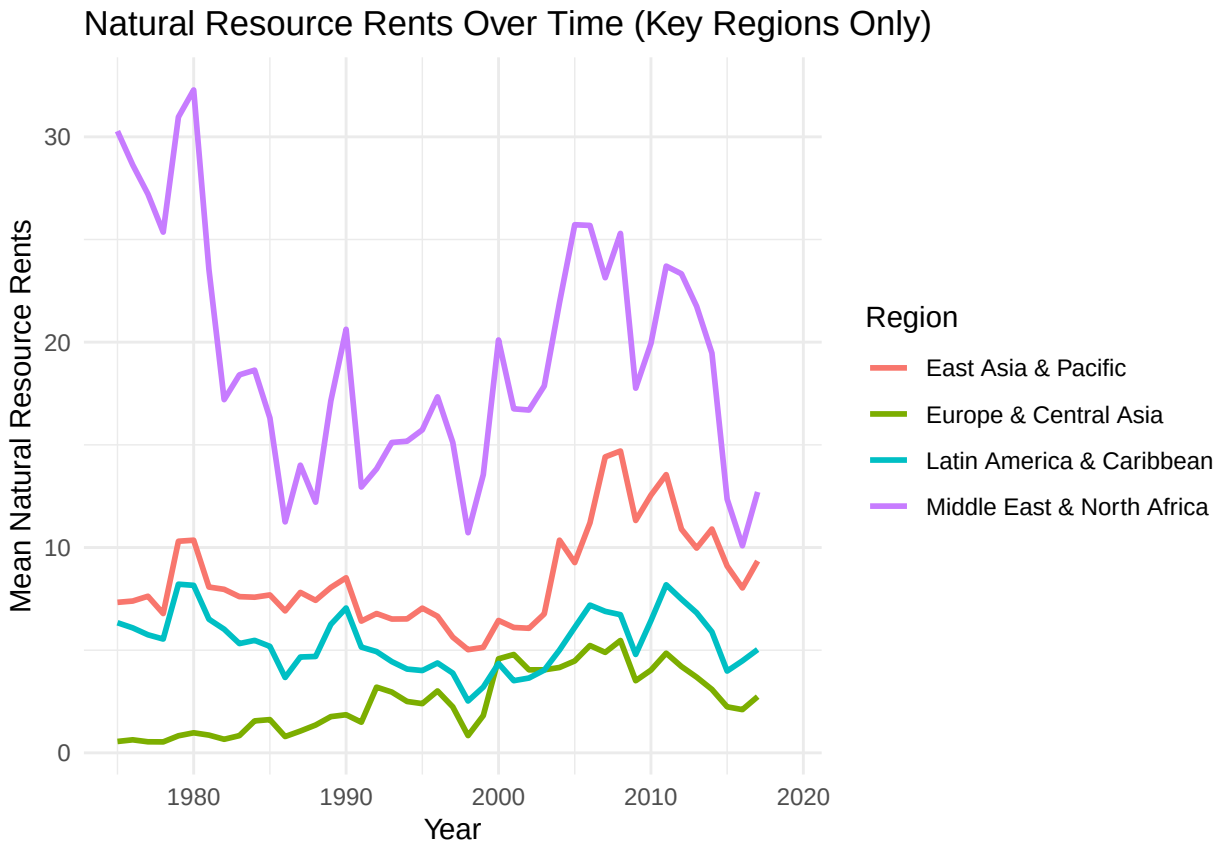
Our dataset has seven different regions, all plotted in the same figure, which can make the plot crowded and difficult to interpret. Therefore, it can be helpful to narrow the analysis to the regions of greatest interest. Suppose, for this research context, we decide to focus on only four key regions (e.g., "East Asia & Pacific," "Middle East & North Africa," "Latin America & Caribbean," and "Europe & Central Asia"). By focusing on a subset of regions, we reduce clutter and draw attention to the geographic areas that are of primary theoretical or substantive interest for our research questions.

```
# Define key regions
key_regions <- c("East Asia & Pacific",
                 "Middle East & North Africa",
                 "Europe & Central Asia",
                 "Latin America & Caribbean")

# Filter dataset to only have the key regions
rents_key_regions <- rents_regional %>%
  filter(region %in% key_regions)

# Create line plot for key regions
ggplot(rents_key_regions,
       aes(x = year, y = natresource_rents_mean, color = region)) +
  geom_line(size = 1) +
  labs(title = "Natural Resource Rents Over Time (Key Regions Only)",
```

```
      x = "Year",
      y = "Mean Natural Resource Rents",
      color = "Region") +
  theme_minimal()
```



Conclusion

Throughout this walkthrough, we demonstrated how data management is a key step in the research process, not the end goal. Researchers often start with raw data that is not immediately suitable for analysis or visualization. By grouping, summarizing, and aggregating data, we transform it into a structured format that reveals meaningful insights.

In this walkthrough, you learned essential techniques for summarizing and simplifying large datasets. You explored how to aggregate data at global, regional, and decadal levels, calculate key summary statistics, and prepare data for visualization. These skills are invaluable for identifying patterns and trends in complex datasets.

Moving forward, you will apply and refine these techniques in groupwork and homework assignments, working with real-world data to strengthen your understanding of data management and its role in research and analysis.