Module 1: Tabular Data

Working with larger-than-RAM data using duckdbfs

ESPM 288

Table of contents

Introduction

In this module, we will explore high-performance workflows for tabular data. We will use `duckdbfs` to work with datasets that are larger than available RAM by leveraging DuckDB's streaming and remote file access capabilities.

Case Study: Global Supply Chains

We will be working with EXIOBASE 3.8.1, a global Multi-Regional Input-Output (MRIO) database. This dataset tracks economic transactions between sectors and regions, along with their environmental impacts (emissions, resource use, etc.).

Data description: - **Coverage**: 44 countries + 5 rest-of-world regions. - **Timeframe**: 1995–2022. - **Content**: Economic transactions (Z matrix), final demand (Y matrix), and environmental stressors (F matrix). - **Format**: Cloud-optimized Parquet, partitioned by year and matrix type.

Setup

```
library(duckdbfs)
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
```

Exercise 1: connecting to remote data

We can open the entire dataset without downloading it using `open_dataset()`. The data is hosted on Source Cooperative. The `**` pattern allows recursive scanning of the partitioned parquet files.

```r
# Remote S3 path to EXIOBASE 3 (Source Cooperative)
duckdbfs::duckdb_secrets(
    key = "",
    secret = "",
    endpoint = "s3.amazonaws.com",
    region = "us-west-2"
)
```

```
[1] 1
```

```r
s3_url <- "s3://us-west-2.opendata.source.coop/youssef-harby/exiobase-3/4588235/parquet/**"

# Open the dataset lazily
exio <- open_dataset(s3_url)

# View the schema (column names and types) without reading data
glimpse(exio)
```

```
Rows: ??
Columns: 8
Database: DuckDB 1.4.3 [ltsta@Windows 10 x64:R 4.5.2/:memory:]
$ stressor <chr> "Value Added", "Value Added", "Value Added", "Value Added", "~
$ region   <chr> "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "~
$ sector   <chr> "Cultivation of wheat", "Cultivation of cereal grains nec", "~
$ value    <dbl> 183.1118891, 402.2305799, 830.2127384, 101.9705426, 31.763189~
$ unit     <chr> "M.EUR", "M.EUR", "M.EUR", "M.EUR", "M.EUR", "M.EUR", "M.EUR"~
$ year     <dbl> 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1~
$ format   <chr> "ixi", "ixi", "ixi", "ixi", "ixi", "ixi", "ixi", "ixi", "ixi"~
$ matrix   <chr> "F_impacts", "F_impacts", "F_impacts", "F_impacts", "F_impact~
```

Exercise 2: Efficient Filtering

The dataset is large. We should filter *before* collecting any data into R.

```r
exio |>
    filter(year == 2022, region == "US") |>
    head() |> # view the first 6 rows
    collect()
```

```
# A tibble: 6 x 8
  stressor    region sector                       value unit   year format matrix
  <chr>       <chr>  <chr>                        <dbl> <chr> <dbl> <chr>  <chr>
1 Value Added US     Cultivation of paddy rice     750. M.EUR  2022 ixi    F_imp~
2 Value Added US     Cultivation of wheat         2019. M.EUR  2022 ixi    F_imp~
3 Value Added US     Cultivation of cereal gra~   7355. M.EUR  2022 ixi    F_imp~
4 Value Added US     Cultivation of vegetables~ 26878. M.EUR  2022 ixi    F_imp~
5 Value Added US     Cultivation of oil seeds     5003. M.EUR  2022 ixi    F_imp~
6 Value Added US     Cultivation of sugar cane~    290. M.EUR  2022 ixi    F_imp~
```

Exercise 3: CO2 Emissions Analysis

Read CO2 emissions data from the F_satellite matrix and analyze top sectors.

```r
# Read CO2 production data from F_satellite matrix
# Filter for CO2-related stressors in 2022
co2_data <- exio |>
    filter(
        year == 2022,
        matrix == "F_satellite",
        stressor %like% "%CO2%"
    ) |>
    collect()

# View the CO2 data
co2_data
```

```
# A tibble: 35,334 x 8
   stressor               region sector           value unit   year format matrix
   <chr>                  <chr>  <chr>            <dbl> <chr> <dbl> <chr>  <chr>
 1 CO2 - combustion - air AT     Cultivation o~ 2.40e8 kg     2022 ixi    F_sat~
 2 CO2 - combustion - air AT     Cultivation o~ 2.27e8 kg     2022 ixi    F_sat~
 3 CO2 - combustion - air AT     Cultivation o~ 9.84e7 kg     2022 ixi    F_sat~
 4 CO2 - combustion - air AT     Cultivation o~ 4.81e7 kg     2022 ixi    F_sat~
 5 CO2 - combustion - air AT     Cultivation o~ 1.29e7 kg     2022 ixi    F_sat~
 6 CO2 - combustion - air AT     Cultivation o~ 1.12e4 kg     2022 ixi    F_sat~
 7 CO2 - combustion - air AT     Cultivation o~ 1.97e6 kg     2022 ixi    F_sat~
 8 CO2 - combustion - air AT     Cattle farming 9.26e7 kg     2022 ixi    F_sat~
 9 CO2 - combustion - air AT     Pigs farming   5.57e7 kg     2022 ixi    F_sat~
10 CO2 - combustion - air AT     Poultry farmi~ 5.07e7 kg     2022 ixi    F_sat~
# i 35,324 more rows
```

```r
glimpse(co2_data)
```

```
Rows: 35,334
Columns: 8
$ stressor <chr> "CO2 - combustion - air", "CO2 - combustion - air", "CO2 - co~
$ region   <chr> "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "~
$ sector   <chr> "Cultivation of wheat", "Cultivation of cereal grains nec", "~
$ value    <dbl> 2.401531e+08, 2.271071e+08, 9.838481e+07, 4.809213e+07, 1.291~
```

```
$ unit     <chr> "kg", "kg", "kg", "kg", "kg", "kg", "kg", "kg", "kg", "kg", "~
$ year     <dbl> 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2~
$ format   <chr> "ixi", "ixi", "ixi", "ixi", "ixi", "ixi", "ixi", "ixi", "ixi"~
$ matrix   <chr> "F_satellite", "F_satellite", "F_satellite", "F_satellite", "~
```

```r
# Find unique CO2 stressors
unique_stressors <- co2_data |>
    distinct(stressor) |>
    arrange(stressor)

unique_stressors
```

```
# A tibble: 6 x 1
  stressor
  <chr>
1 CO2 - agriculture - peat decay - air
2 CO2 - combustion - air
3 CO2 - non combustion - Cement production - air
4 CO2 - non combustion - Lime production - air
5 CO2 - waste - biogenic - air
6 CO2 - waste - fossil - air
```

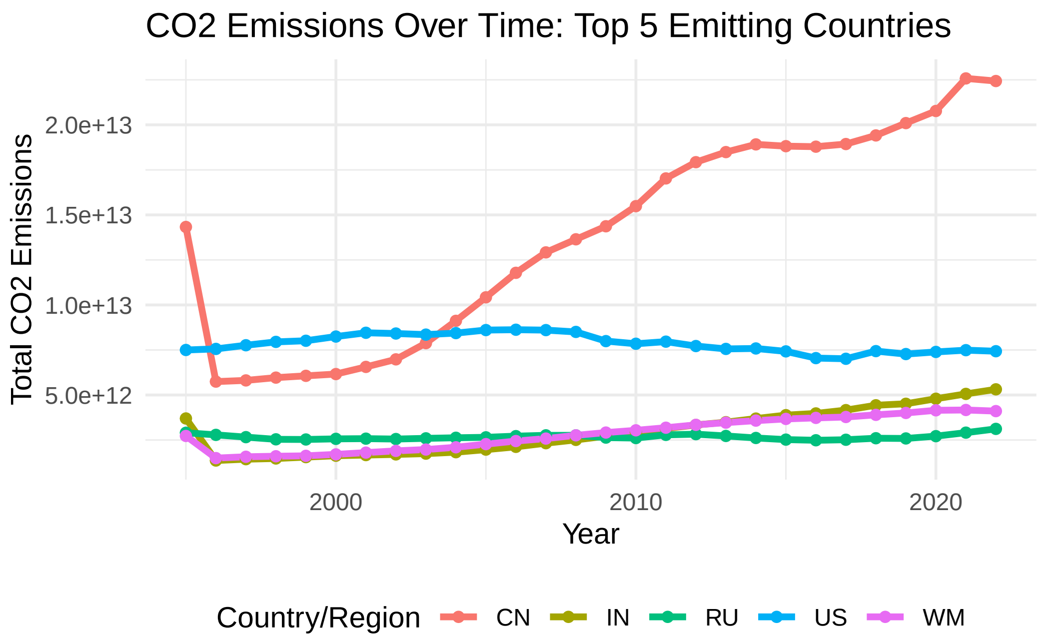
Exercise 3: Time Series Analysis of Top CO2 Emitters

Now let's create a visualization showing CO2 emissions over time for the top 5 emitting countries.

```r
library(ggplot2)

# Get CO2 data, identify top 5 emitters, and create time series
exio |>
    filter(matrix == "F_satellite", stressor %like% "%CO2%") |>
    collect() |>
    group_by(region) |>
    mutate(region_total = sum(value, na.rm = TRUE)) |>
    ungroup() |>
    filter(dense_rank(desc(region_total)) <= 5) |>
    group_by(year, region) |>
    summarize(total_co2 = sum(value, na.rm = TRUE), .groups = "drop") |>
    ggplot(aes(x = year, y = total_co2, color = region)) +
    geom_line(linewidth = 1.2) +
    geom_point(linewidth = 2) +
    labs(
        title = "CO2 Emissions Over Time: Top 5 Emitting Countries",
        x = "Year",
        y = "Total CO2 Emissions",
        color = "Country/Region"
    ) +
    theme_minimal() +
    theme(legend.position = "bottom")
```

```
Warning in geom_point(linewidth = 2): Ignoring unknown parameters: `linewidth`
```

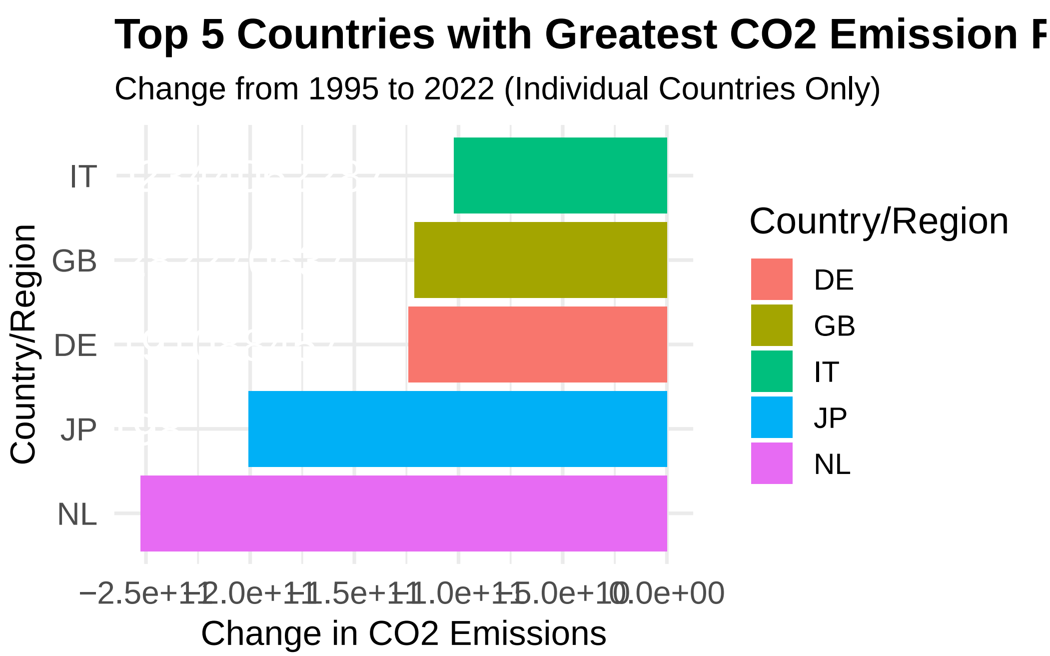


Exercise 4: Top Countries Reducing CO2 Emissions

Let's identify the countries that have achieved the greatest reduction in CO2 emissions from 1995 to 2022.

```r
# Calculate emission changes and identify top 5 reducers (individual countries only)
exio |>
    filter(matrix == "F_satellite", stressor %like% "%CO2%") |>
    collect() |>
    filter(!region %in% c("WE", "WA", "WF", "WL", "WM")) |>  # Exclude aggregated regions
    group_by(year, region) |>
    summarize(total_co2 = sum(value, na.rm = TRUE), .groups = "drop") |>
    group_by(region) |>
    filter(n() >= 2) |>  # Ensure at least 2 years of data
    summarize(
        first_year_emissions = total_co2[which.min(year)],
        last_year_emissions = total_co2[which.max(year)],
        change = last_year_emissions - first_year_emissions,
        .groups = "drop"
    ) |>
    arrange(change) |>
    head(5) |>
    ggplot(aes(x = reorder(region, change), y = change, fill = region)) +
    geom_col() +
    geom_text(aes(label = round(change, 0)), hjust = 1.2, color = "white", size = 6) +
    coord_flip() +
    labs(
```

```r
        title = "Top 5 Countries with Greatest CO2 Emission Reductions",
        subtitle = "Change from 1995 to 2022 (Individual Countries Only)",
        x = "Country/Region",
        y = "Change in CO2 Emissions",
        fill = "Country/Region"
    ) +
    theme_minimal(base_size = 14) +
    theme(
        legend.position = "right",
        plot.title = element_text(size = 16, face = "bold"),
        plot.subtitle = element_text(size = 12),
        axis.text = element_text(size = 12),
        axis.title = element_text(size = 13)
    )
```




```r
# Find top 5 sectors in the US by CO2 emissions in 2022
top_co2_sectors <- co2_data |>
    filter(region == "US") |>
    group_by(sector) |>
    summarise(total_co2 = sum(value, na.rm = TRUE)) |>
    arrange(desc(total_co2)) |>
    slice_head(n = 5)

top_co2_sectors
```

```
# A tibble: 5 x 2
  sector                               total_co2
  <chr>                                    <dbl>
1 Production of electricity by coal      1.38e12
```

```
2 Electricity by coal                  1.18e12
3 Production of electricity by gas     6.97e11
4 Electricity by gas                   6.33e11
5 Chemicals nec                        2.19e11
```

Exercise 4: Time Series of Top CO2 Emitting Countries

Visualize CO2 emissions over time for the top 5 emitting countries.

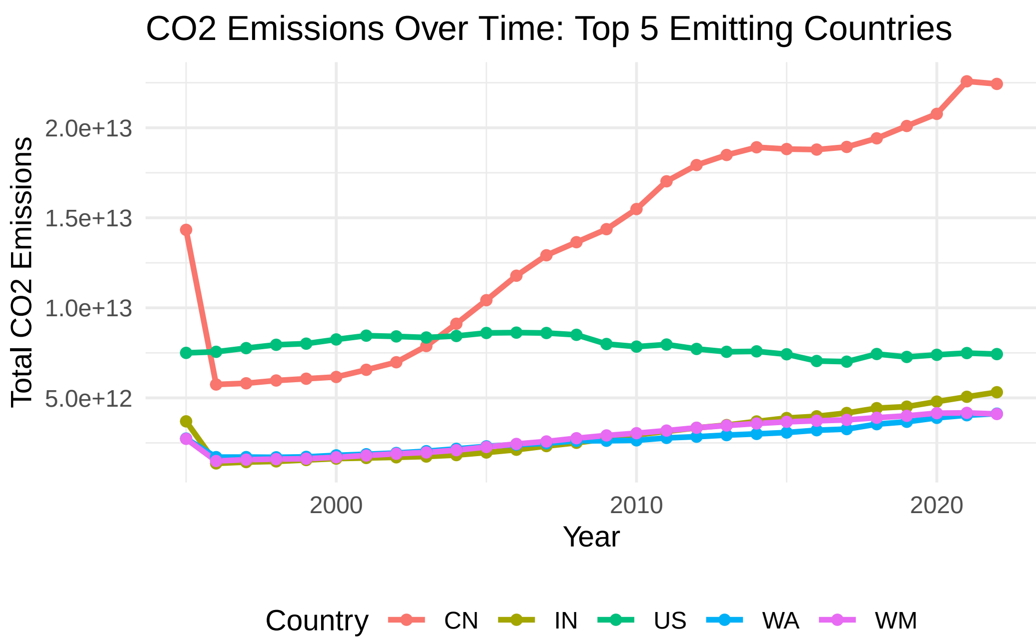
```r
library(ggplot2)

# First, get all CO2 data across all years
co2_all_years <- exio |>
    filter(
        matrix == "F_satellite",
        stressor %like% "%CO2%"
    ) |>
    collect()

# Identify top 5 CO2 emitting countries in 2022
top_5_countries <- co2_all_years |>
    filter(year == 2022) |>
    group_by(region) |>
    summarise(total_co2 = sum(value, na.rm = TRUE)) |>
    arrange(desc(total_co2)) |>
    slice_head(n = 5) |>
    pull(region)

# Filter data for top 5 countries across all years
co2_time_series <- co2_all_years |>
    filter(region %in% top_5_countries) |>
    group_by(year, region) |>
    summarise(total_co2 = sum(value, na.rm = TRUE), .groups = "drop")

# Create the plot
ggplot(co2_time_series, aes(x = year, y = total_co2, color = region)) +
    geom_line(linewidth = 1) +
    geom_point() +
    labs(
        title = "CO2 Emissions Over Time: Top 5 Emitting Countries",
        x = "Year",
        y = "Total CO2 Emissions",
        color = "Country"
    ) +
    theme_minimal() +
    theme(legend.position = "bottom")
```



Exercise 5: Top CO2 Emitting Industries Globally

Identify the top 5 industries/sectors that emit the most CO2 globally in 2022.

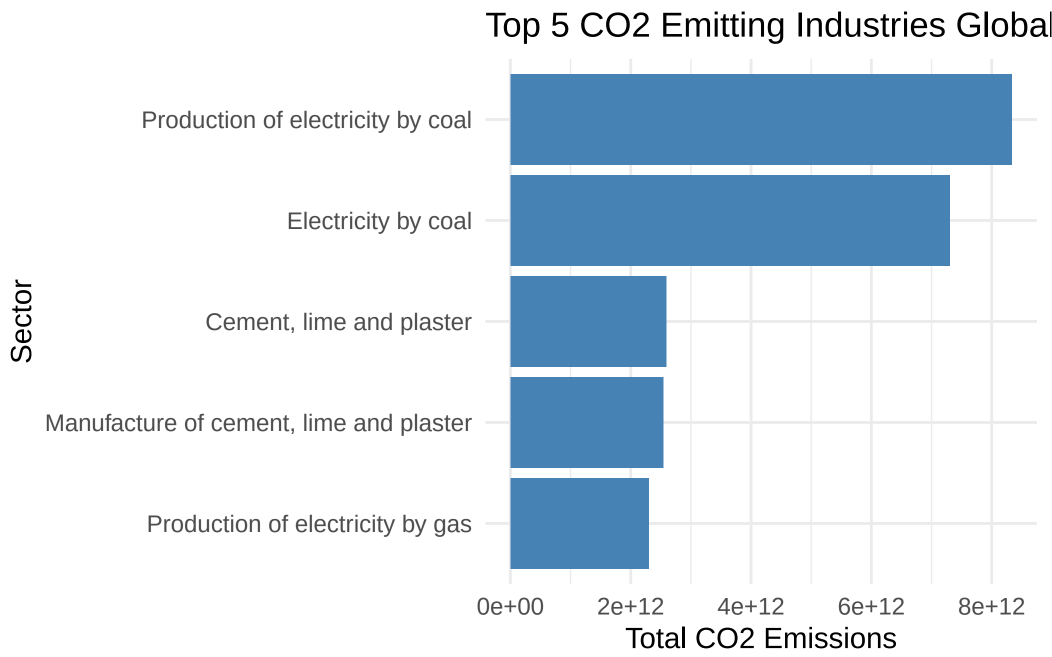
```
# Find top 5 industries globally by CO2 emissions in 2022
top_co2_industries <- co2_data |>
    group_by(sector) |>
    summarise(total_co2 = sum(value, na.rm = TRUE)) |>
    arrange(desc(total_co2)) |>
    slice_head(n = 5)

top_co2_industries
```

```
# A tibble: 5 x 2
  sector                                  total_co2
  <chr>                                       <dbl>
1 Production of electricity by coal         8.33e12
2 Electricity by coal                       7.30e12
3 Cement, lime and plaster                  2.59e12
4 Manufacture of cement, lime and plaster   2.54e12
5 Production of electricity by gas          2.30e12
```

```
# Visualize top 5 emitting industries
ggplot(top_co2_industries, aes(x = reorder(sector, total_co2), y = total_co2)) +
    geom_col(fill = "steelblue") +
    coord_flip() +
    labs(
        title = "Top 5 CO2 Emitting Industries Globally (2022)",
        x = "Sector",
```

```
        y = "Total CO2 Emissions"
    ) +
    theme_minimal()
```



Exercise 6: types of pollutants

In the US, what are the top 5 sectors with highest emissions in 2022?

```
# Find top 5 sectors with highest total emissions in the US in 2022
top_us_sectors <- exio |>
    filter(year == 2022, matrix == "F_satellite", region == "US") |>
    collect() |>
    group_by(sector) |>
    summarise(total_emissions = sum(value, na.rm = TRUE), .groups = "drop") |>
    arrange(desc(total_emissions)) |>
    slice_head(n = 5)

top_us_sectors
```

```
# A tibble: 5 x 2
  sector                             total_emissions
  <chr>                                        <dbl>
1 Production of electricity by coal          1.39e12
2 Electricity by coal                        1.19e12
3 Production of electricity by gas           6.98e11
4 Electricity by gas                         6.34e11
5 Chemicals nec                              2.23e11
```