STAT_535_Project

Sayak Chatterjee

2025-12-06

Data scraping for the Asthama rates for different years across the different states in the US

```
urls <- c(
  "2000" = "https://www.cdc.gov/asthma/brfss/00/current/tableC1.htm",
  "2001" = "https://www.cdc.gov/asthma/brfss/01/current/tableC1.htm",
  "2002" = "https://www.cdc.gov/asthma/brfss/02/current/tableC1.htm",
  "2003" = "https://www.cdc.gov/asthma/brfss/03/current/tableC1.htm",
  "2004" = "https://www.cdc.gov/asthma/brfss/04/current/tableC1.htm",
  "2005" = "https://www.cdc.gov/asthma/brfss/05/current/tableC1.htm",
  "2006" = "https://www.cdc.gov/asthma/brfss/06/current/tableC1.htm",
  "2007" = "https://www.cdc.gov/asthma/brfss/07/current/tableC1.htm",
  "2008" = "https://www.cdc.gov/asthma/brfss/08/current/tableC1.htm",
  "2009" = "https://www.cdc.gov/asthma/brfss/09/current/tableC1.htm",
  "2010" = "https://www.cdc.gov/asthma/brfss/2010/current/tableC1.htm",
  "2011" = "https://www.cdc.gov/asthma/brfss/2011/tableC1.htm",
  "2012" = "https://www.cdc.gov/asthma/brfss/2012/tableC1.htm",
  "2013" = "https://www.cdc.gov/asthma/brfss/2013/tableC1.htm",
  "2014" = "https://www.cdc.gov/asthma/brfss/2014/tableC1.htm",
  "2015" = "https://www.cdc.gov/asthma/brfss/2015/tableC1.htm",
  "2016" = "https://www.cdc.gov/asthma/brfss/2016/tableC1.htm",
  "2017" = "https://www.cdc.gov/asthma/brfss/2017/tableC1.htm",
  "2018" = "https://www.cdc.gov/asthma/brfss/2018/tableC1.html",
  "2019" = "https://www.cdc.gov/asthma/brfss/2019/tableC1.html",
  "2020" = "https://www.cdc.gov/asthma/brfss/2020/tableC1.html"
)
```

```
library(rvest)
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
```

```r
library(stringr)
library(purrr)

scrape_c1 <- function(url, year) {
  message("Scraping ", year, " from ", url)

  page <- try(read_html(url), silent = TRUE)
  if (inherits(page, "try-error")) {
    warning("Failed to load ", url)
    return(NULL)
  }

  # Extract all HTML tables
  tables <- page %>% html_table(fill = TRUE)
  if (length(tables) == 0) {
    warning("No tables found for ", year)
    return(NULL)
  }

  # Pick the table that contains "State" in headers
  idx <- which.max(
    sapply(tables, function(x) any(grepl("State", names(x), ignore.case = TRUE)))
  )

  df <- tables[[idx]]
  names(df) <- tolower(names(df))

  # Identify relevant columns
  state_col <- grep("state", names(df), value = TRUE)[1]
  prev_col  <- grep("prev|percent", names(df), value = TRUE)[1]
  se_col    <- grep("se|error", names(df), value = TRUE)[1]

  if (is.na(state_col) || is.na(prev_col) || is.na(se_col)) {
    warning("Missing columns in ", year)
    return(NULL)
  }

  # Clean the table
  df_clean <- df %>%
    select(state = !!state_col, prevalence = !!prev_col, se = !!se_col) %>%
    mutate(
      year = as.numeric(year),
      state = trimws(state),
      prevalence = as.numeric(gsub("[^0-9.]", "", prevalence)),
      se = as.numeric(gsub("[^0-9.]", "", se))
    ) %>%
    filter(!is.na(prevalence), !grepl("total", state, ignore.case = TRUE))

  return(df_clean)
}

all_data <- map2_df(urls, names(urls), scrape_c1)
```

```
## Scraping 2000 from https://www.cdc.gov/asthma/brfss/00/current/tableC1.htm

## Scraping 2001 from https://www.cdc.gov/asthma/brfss/01/current/tableC1.htm

## Scraping 2002 from https://www.cdc.gov/asthma/brfss/02/current/tableC1.htm

## Scraping 2003 from https://www.cdc.gov/asthma/brfss/03/current/tableC1.htm

## Scraping 2004 from https://www.cdc.gov/asthma/brfss/04/current/tableC1.htm

## Scraping 2005 from https://www.cdc.gov/asthma/brfss/05/current/tableC1.htm

## Scraping 2006 from https://www.cdc.gov/asthma/brfss/06/current/tableC1.htm

## Scraping 2007 from https://www.cdc.gov/asthma/brfss/07/current/tableC1.htm

## Scraping 2008 from https://www.cdc.gov/asthma/brfss/08/current/tableC1.htm

## Scraping 2009 from https://www.cdc.gov/asthma/brfss/09/current/tableC1.htm

## Scraping 2010 from https://www.cdc.gov/asthma/brfss/2010/current/tableC1.htm

## Scraping 2011 from https://www.cdc.gov/asthma/brfss/2011/tableC1.htm

## Scraping 2012 from https://www.cdc.gov/asthma/brfss/2012/tableC1.htm

## Scraping 2013 from https://www.cdc.gov/asthma/brfss/2013/tableC1.htm

## Scraping 2014 from https://www.cdc.gov/asthma/brfss/2014/tableC1.htm

## Scraping 2015 from https://www.cdc.gov/asthma/brfss/2015/tableC1.htm

## Scraping 2016 from https://www.cdc.gov/asthma/brfss/2016/tableC1.htm

## Scraping 2017 from https://www.cdc.gov/asthma/brfss/2017/tableC1.htm

## Scraping 2018 from https://www.cdc.gov/asthma/brfss/2018/tableC1.html

## Scraping 2019 from https://www.cdc.gov/asthma/brfss/2019/tableC1.html

## Scraping 2020 from https://www.cdc.gov/asthma/brfss/2020/tableC1.html
```

```
unique(all_data$year)
```

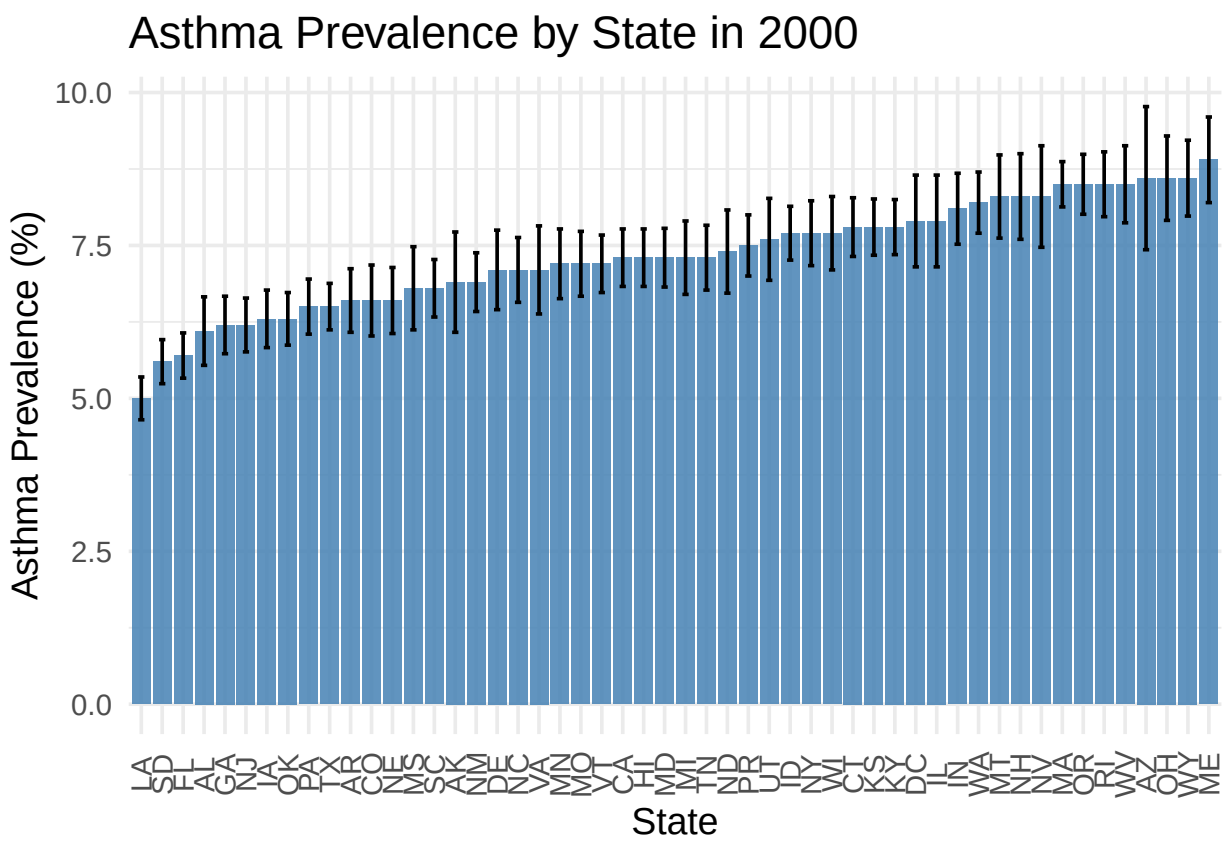
```
##  [1] 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014
## [16] 2015 2016 2017 2018 2019 2020
```

```r
library(dplyr)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.5.2
```

```r
plot_asthma_year <- function(data, year_to_plot) {

  df_year <- data %>%
    filter(year == year_to_plot,
           !grepl("total", state, ignore.case = TRUE)) %>%
    mutate(state = trimws(state)) %>%
    arrange(prevalence)

  ggplot(df_year, aes(x = reorder(state, prevalence), y = prevalence)) +
    geom_col(fill = "steelblue", alpha = 0.85) +
    geom_errorbar(aes(ymin = prevalence - se,
                      ymax = prevalence + se),
                  width = 0.3, color = "black") +
    labs(
      title = paste("Asthma Prevalence by State in", year_to_plot),
      x = "State",
      y = "Asthma Prevalence (%)"
    ) +
    theme_minimal(base_size = 14) +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
}
```

```r
plot_asthma_year(all_data, 2000)
```




```
#plot_asthma_year(all_data, 2001)
#plot_asthma_year(all_data, 2002)
#plot_asthma_year(all_data, 2003)
#plot_asthma_year(all_data, 2004)
#plot_asthma_year(all_data, 2005)
#plot_asthma_year(all_data, 2006)
#plot_asthma_year(all_data, 2007)
#plot_asthma_year(all_data, 2008)
#plot_asthma_year(all_data, 2009)
#plot_asthma_year(all_data, 2010)
#plot_asthma_year(all_data, 2011)
#plot_asthma_year(all_data, 2012)
#plot_asthma_year(all_data, 2013)
#plot_asthma_year(all_data, 2014)
#plot_asthma_year(all_data, 2015)
#plot_asthma_year(all_data, 2016)
#plot_asthma_year(all_data, 2017)
#plot_asthma_year(all_data, 2018)
#plot_asthma_year(all_data, 2019)
#plot_asthma_year(all_data, 2020)
```

Data scraping for the Air pollutants (average) 2000-2024 (AQI)

```r
library(readr)
```

```
##
## Attaching package: 'readr'

## The following object is masked from 'package:rvest':
##
##     guess_encoding
```

```r
library(dplyr)
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
library(stringr)

# Load CSV file
aqi <- read_csv("pollution_2000_2023.csv") %>%
  clean_names()
```

```
## New names:
## * `` -> `...1`

## Rows: 665414 Columns: 22
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (4): Address, State, County, City
## dbl  (17): ...1, O3 Mean, O3 1st Max Value, O3 1st Max Hour, O3 AQI, CO Mean...
## date  (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Convert date column
aqi <- aqi %>%
  mutate(date = suppressWarnings(as.Date(date, format = "%d/%m/%y")),
         year = year(date))

# Identify AQI columns that exist
aqi_columns <- c("o3_aqi", "co_aqi", "so2_aqi", "no2_aqi")
aqi_columns <- aqi_columns[aqi_columns %in% names(aqi)]

# Compute average AQI by state and year
aqi_state_year <- aqi %>%
  group_by(state, year) %>%
  summarise(
    across(
      all_of(aqi_columns),
      ~ mean(.x, na.rm = TRUE)
    )
  ) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.
```

```r
aqi_state_year
```

```
## # A tibble: 816 x 6
##    state    year o3_aqi co_aqi so2_aqi no2_aqi
##    <chr>   <dbl>  <dbl>  <dbl>   <dbl>   <dbl>
##  1 Alabama  2013   18.5   4.13    6.58    21.4
##  2 Alabama  2014   37.2   3.71    8.02    21.5
##  3 Alabama  2015   37.3   3.94    6.97    20.1
##  4 Alabama  2016   41.4   3.90    7.43    22.3
##  5 Alabama  2017   37.8   5.11    5.73    20.1
##  6 Alabama  2018   43.1   3.13    5.65    18.8
##  7 Alabama  2019   39.8   6.34    5.11    20.9
##  8 Alabama  2020   35.0   4.01    2.26    17.9
##  9 Alabama  2021   35.2   4.03    2.40    18.0
## 10 Alabama  2022   38.7   3.43    1.31    18.5
## # i 806 more rows
```

Data scraping for the Air pollutants (average) 2000-2024 (Mean)

```r
library(readr)
library(dplyr)
library(janitor)
library(lubridate)
library(stringr)

# Load CSV file
poll <- read_csv("pollution_2000_2023.csv") %>%
  clean_names()    # converts "O3 Mean" → "o3_mean"
```

```
## New names:
## Rows: 665414 Columns: 22
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (4): Address, State, County, City dbl (17): ...1, O3 Mean, O3 1st Max Value, O3
## 1st Max Hour, O3 AQI, CO Mean... date (1): Date
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

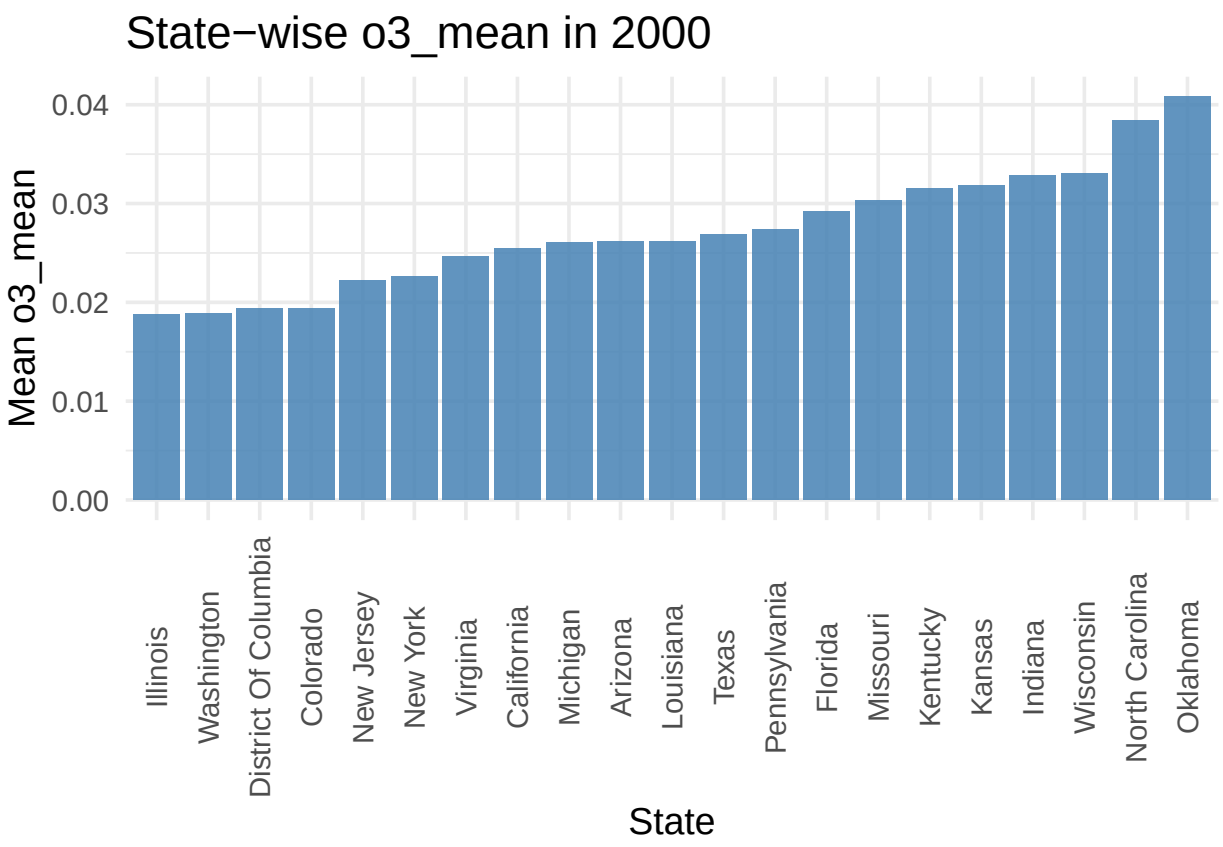
```
# Convert date column (your format is like "01/01/00")
poll <- poll %>%
  mutate(
    date = suppressWarnings(as.Date(date, format = "%m/%d/%y")),
    year = year(date)
  )

# Identify pollutant MEAN columns (NOT AQI)
pollutant_mean_cols <- c(
  "o3_mean",
  "co_mean",
  "so2_mean",
  "no2_mean",
  "pm25_mean",
  "pm10_mean"
)

# Only keep columns that actually exist in the file
pollutant_mean_cols <- pollutant_mean_cols[pollutant_mean_cols %in% names(poll)]

# Compute state-year mean pollutant concentrations
pollution_state_year <- poll %>%
  group_by(state, year) %>%
  summarise(
    across(
      all_of(pollutant_mean_cols),
      ~ mean(.x, na.rm = TRUE)
    ),
    .groups = "drop"
  )

pollution_state_year
```

```
## # A tibble: 816 x 6
##    state    year o3_mean co_mean so2_mean no2_mean
##    <chr>   <dbl>   <dbl>   <dbl>    <dbl>    <dbl>
##  1 Alabama  2013  0.0127   0.212    0.772    12.1
##  2 Alabama  2014  0.0264   0.206    1.20      9.41
##  3 Alabama  2015  0.0253   0.222    1.03      9.09
##  4 Alabama  2016  0.0290   0.216    1.07      9.55
##  5 Alabama  2017  0.0272   0.312    1.13      9.24
##  6 Alabama  2018  0.0295   0.203    1.09      7.97
##  7 Alabama  2019  0.0286   0.422    1.12      9.49
##  8 Alabama  2020  0.0259   0.239    0.759     8.24
```

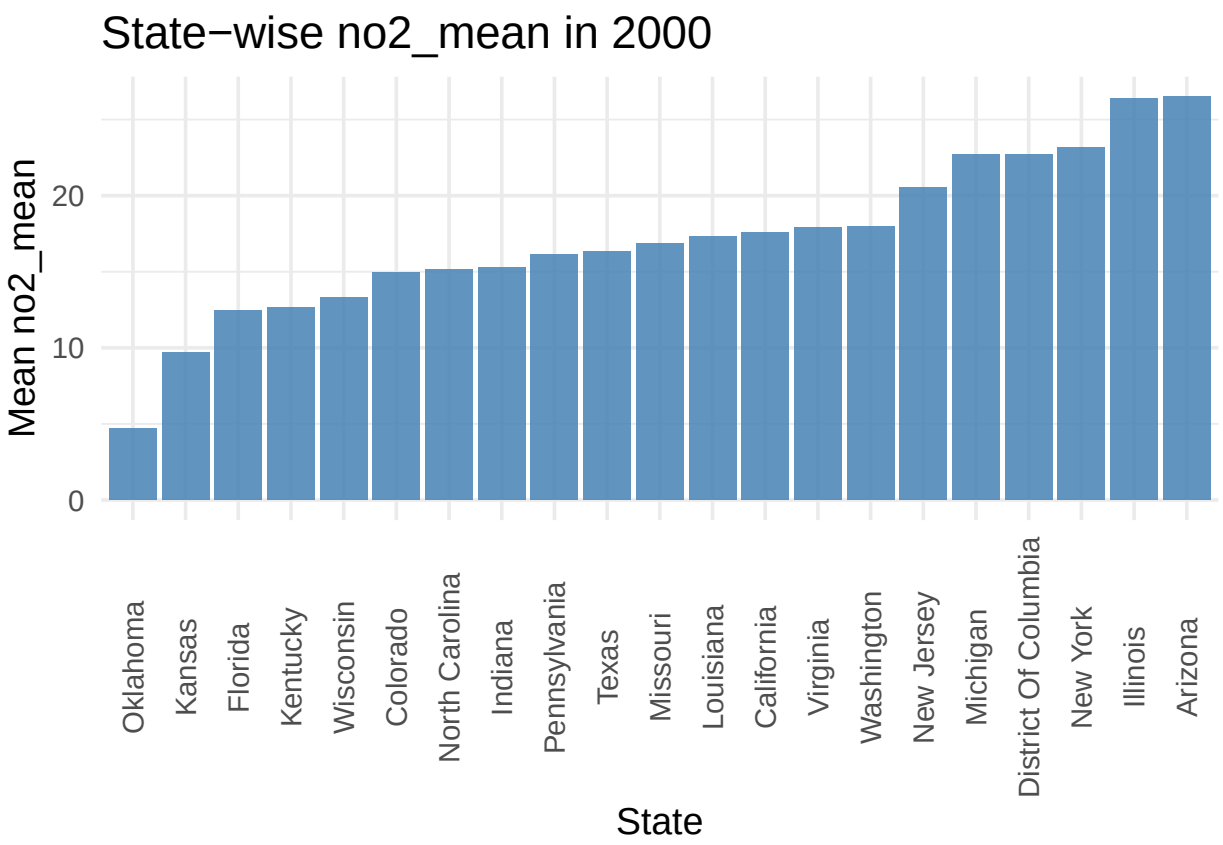
```
##  9 Alabama  2021  0.0248   0.255    0.642     8.53
## 10 Alabama  2022  0.0278   0.207    0.192     8.71
## # i 806 more rows
```

Plotting for the pollutant means

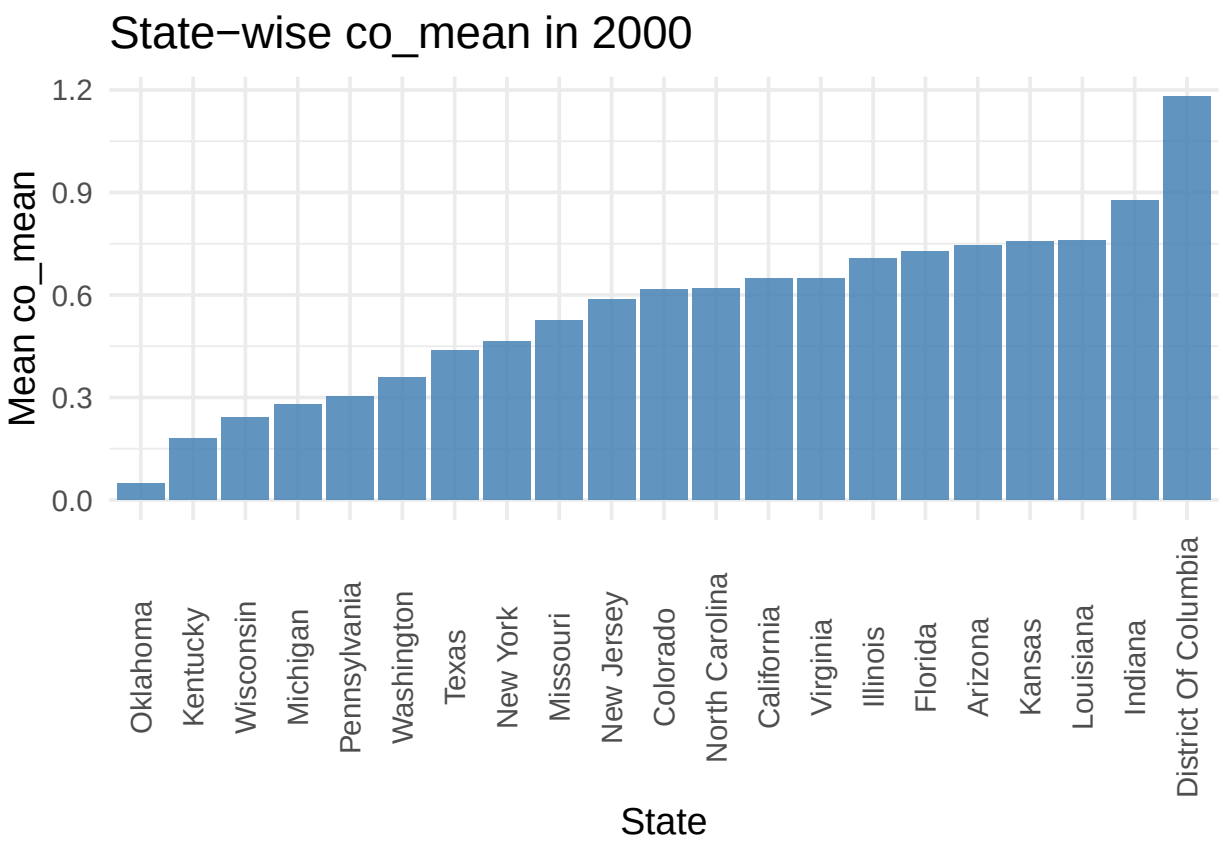
```
library(ggplot2)
library(dplyr)

plot_pollutant_by_state <- function(data, year_to_plot, pollutant_col) {

  df_year <- data %>%
    filter(year == year_to_plot) %>%
    select(state, year, all_of(pollutant_col)) %>%
    arrange(.data[[pollutant_col]])

  ggplot(df_year, aes(x = reorder(state, .data[[pollutant_col]]),
                      y = .data[[pollutant_col]])) +
    geom_col(fill = "steelblue", alpha = 0.85) +
    labs(
      title = paste("State-wise", pollutant_col, "in", year_to_plot),
      x = "State",
      y = paste("Mean", pollutant_col)
    ) +
    theme_minimal(base_size = 14) +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
}

plot_pollutant_by_state(pollution_state_year, 2000, "o3_mean")
```



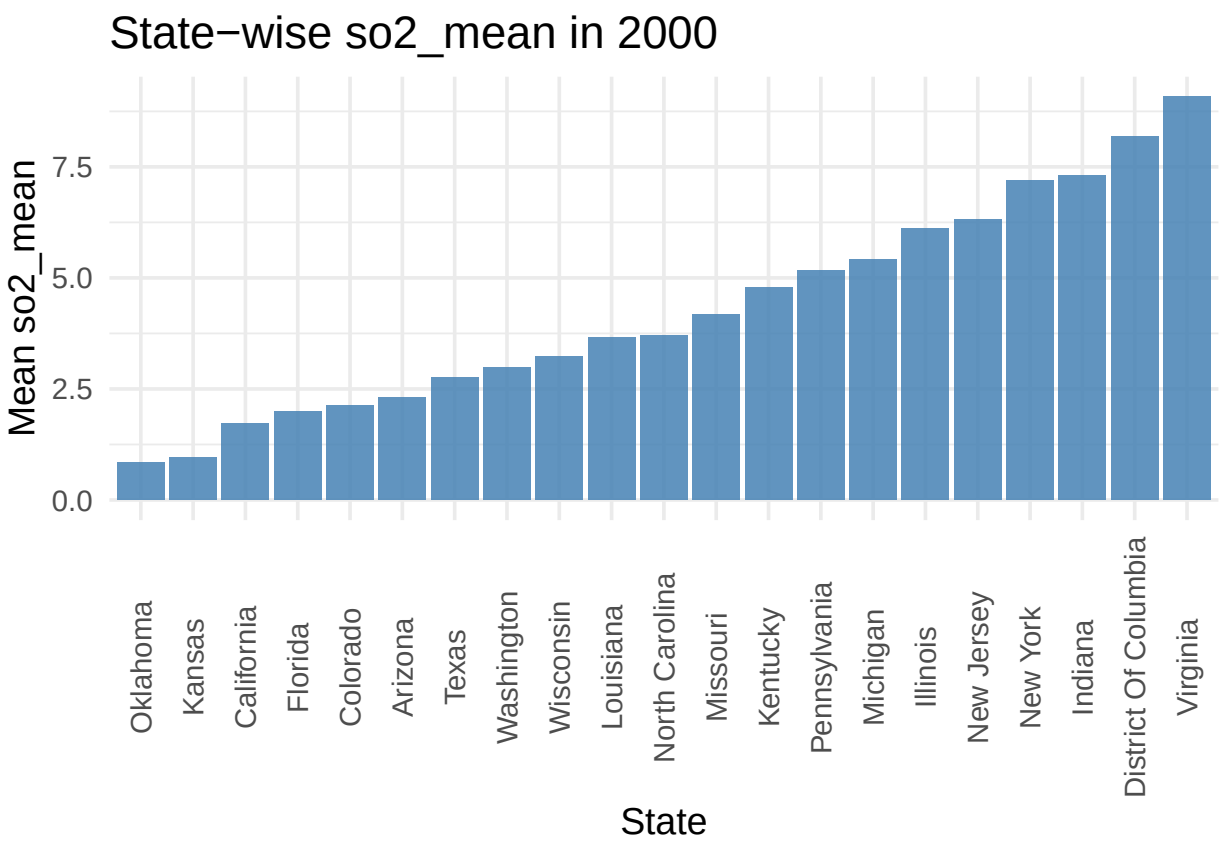

```
plot_pollutant_by_state(pollution_state_year, 2000, "no2_mean")
```




```
plot_pollutant_by_state(pollution_state_year, 2000, "co_mean")
```

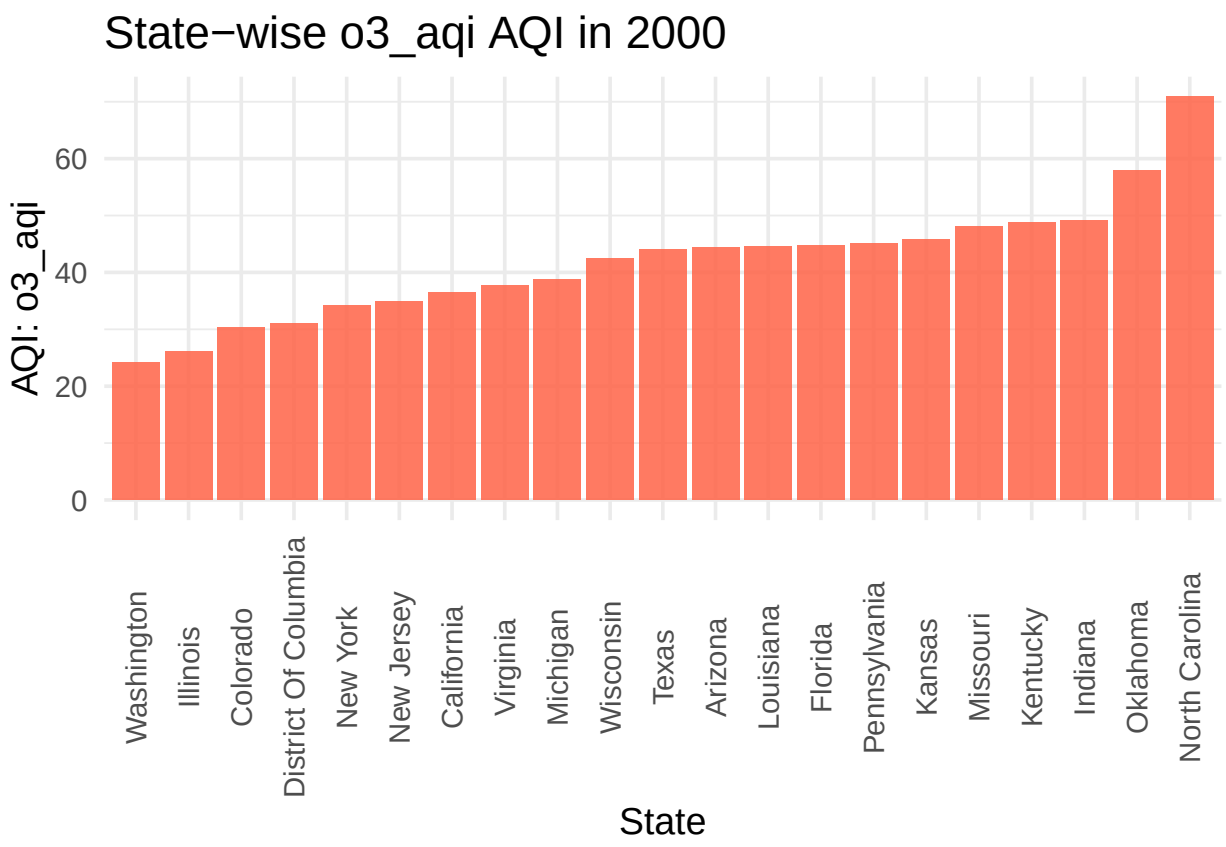



```
plot_pollutant_by_state(pollution_state_year, 2000, "so2_mean")
```

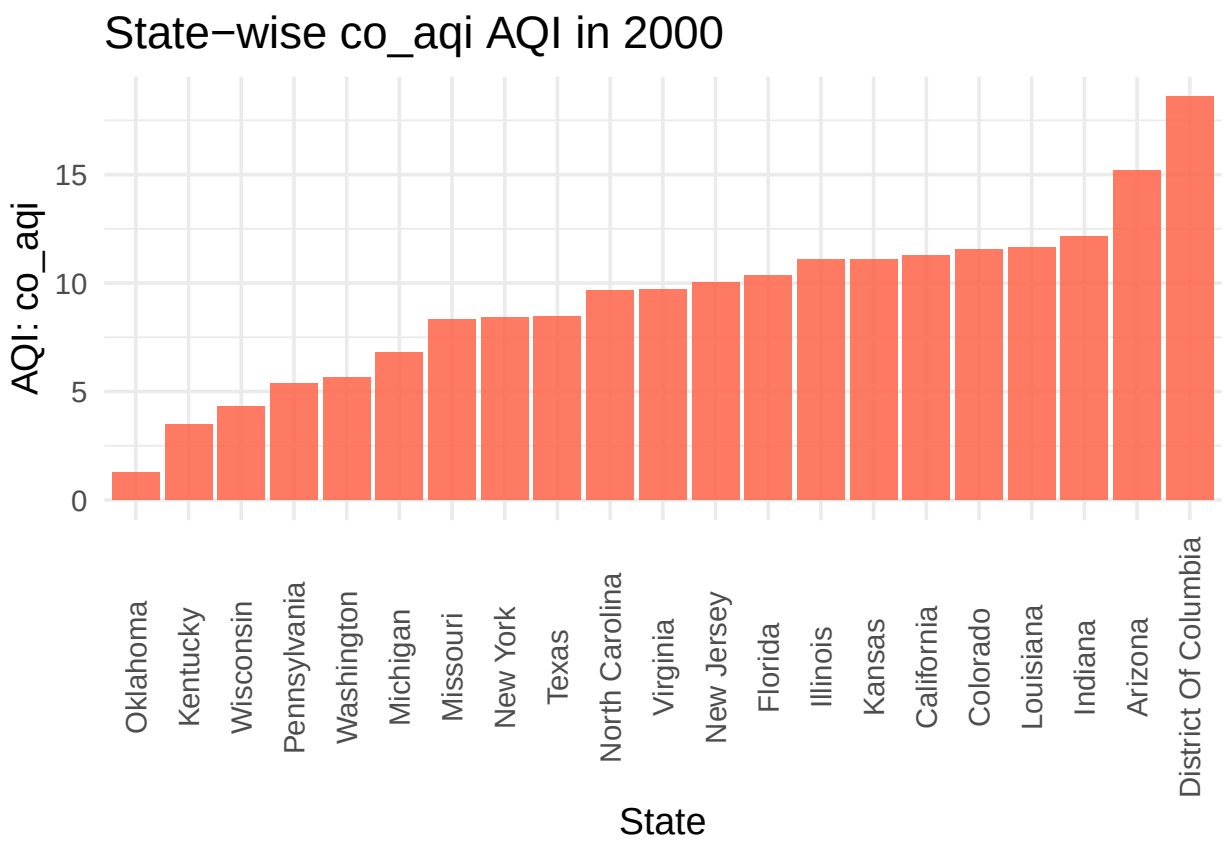


Plotting for the pollutant AQIs

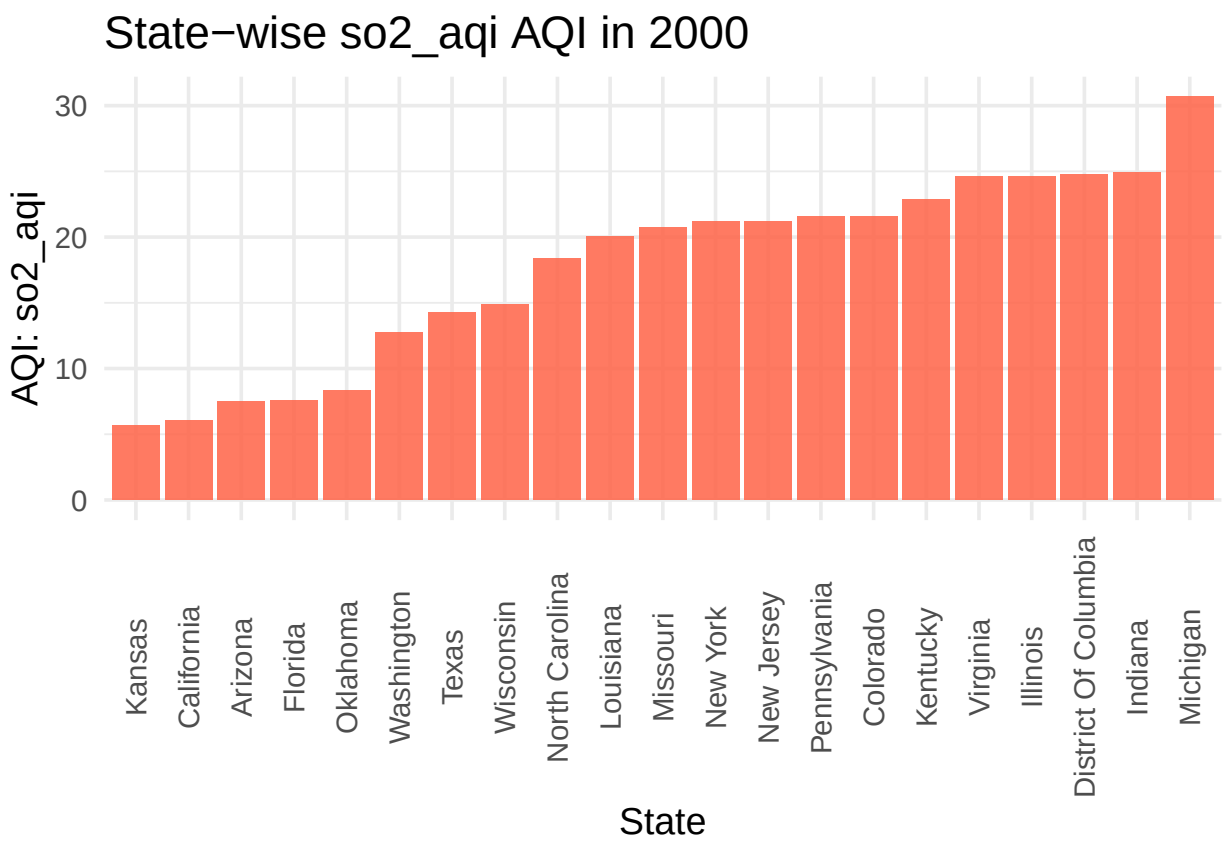
```
plot_aqi_by_state <- function(data, year_to_plot, aqi_col) {

  df_year <- data %>%
    filter(year == year_to_plot) %>%
    select(state, year, all_of(aqi_col)) %>%
    arrange(.data[[aqi_col]])

  ggplot(df_year, aes(x = reorder(state, .data[[aqi_col]]),
                      y = .data[[aqi_col]])) +
    geom_col(fill = "tomato", alpha = 0.85) +
    labs(
      title = paste("State-wise", aqi_col, "AQI in", year_to_plot),
      x = "State",
      y = paste("AQI:", aqi_col)
    ) +
    theme_minimal(base_size = 14) +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
}

plot_aqi_by_state(aqi_state_year, 2000, "o3_aqi")
```



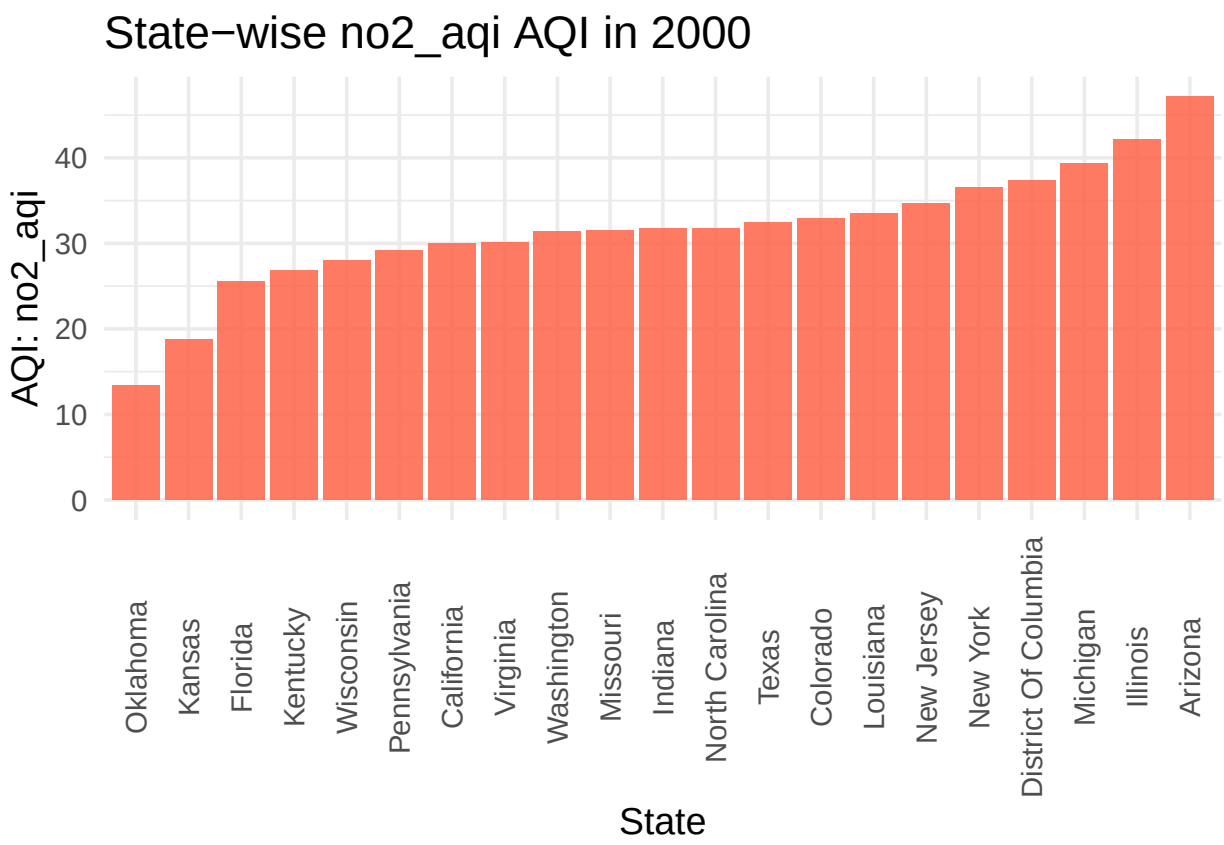

```
plot_aqi_by_state(aqi_state_year, 2000, "co_aqi")
```




```
plot_aqi_by_state(aqi_state_year, 2000, "so2_aqi")
```




```
plot_aqi_by_state(aqi_state_year, 2000, "no2_aqi")
```

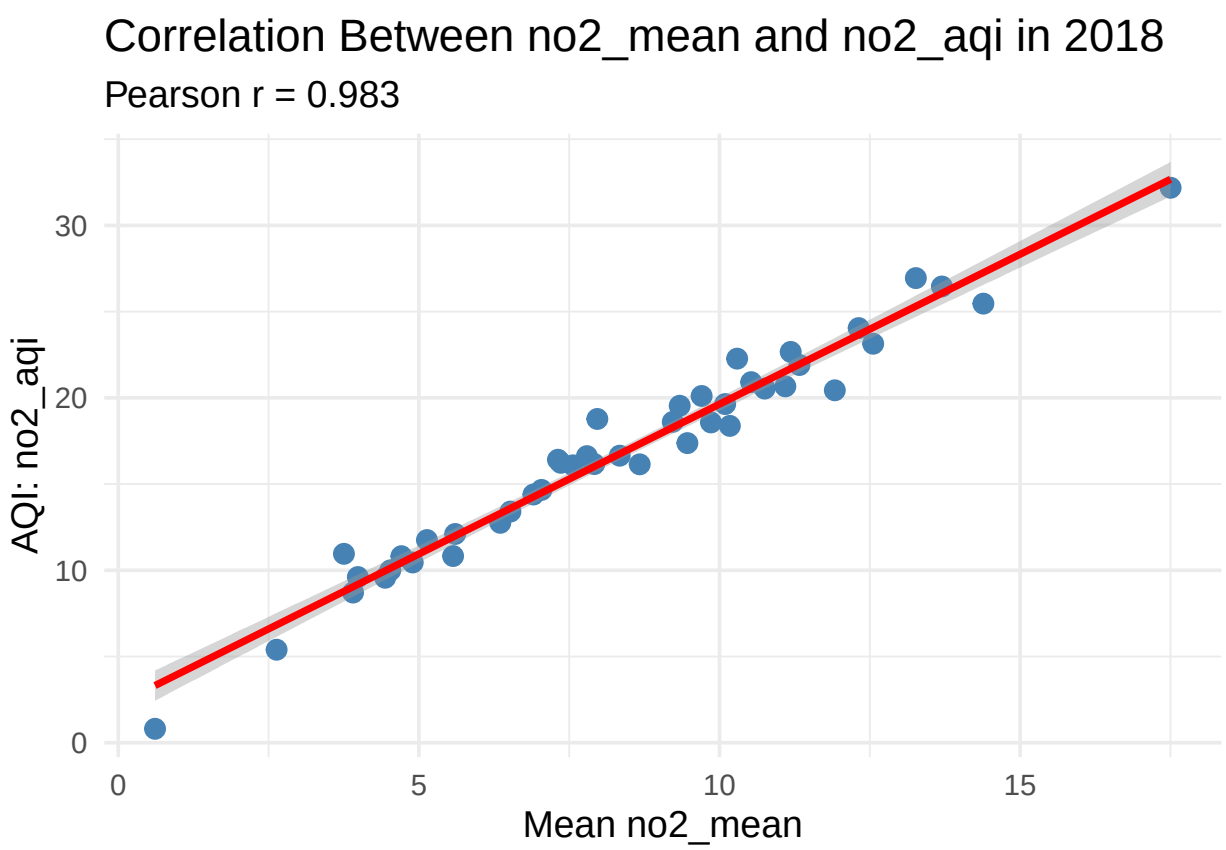


Correlation between AQI and the mean value of a particular pollutant

```
library(ggplot2)
library(dplyr)

plot_correlation_aqi_pollutant <- function(aqi_data, pollutant_data,
                                           year_to_plot,
                                           aqi_col, pollutant_col) {

  # Merge AQI + pollutant dataset by state + year
  df <- aqi_data %>%
    inner_join(pollutant_data, by = c("state", "year")) %>%
    filter(year == year_to_plot) %>%
    select(state, year, all_of(aqi_col), all_of(pollutant_col))

  # Compute correlation
  corr_value <- round(cor(df[[aqi_col]], df[[pollutant_col]], use = "complete.obs"), 3)

  # Plot
  ggplot(df, aes(x = .data[[pollutant_col]], y = .data[[aqi_col]])) +
    geom_point(color = "steelblue", size = 3) +
    geom_smooth(method = "lm", se = TRUE, color = "red") +
    labs(
```

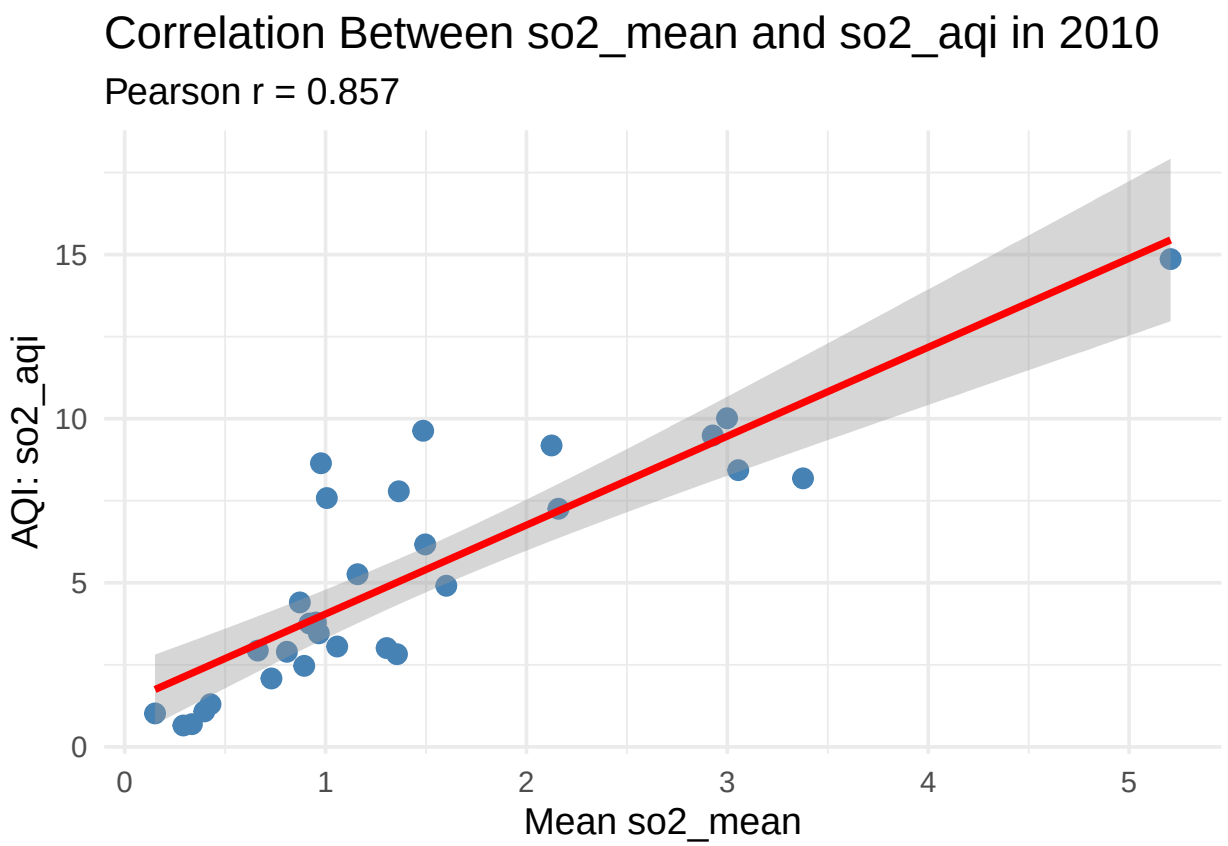
```
      title = paste("Correlation Between", pollutant_col, "and", aqi_col, "in", year_to_plot),
      subtitle = paste("Pearson r =", corr_value),
      x = paste("Mean", pollutant_col),
      y = paste("AQI:", aqi_col)
    ) +
    theme_minimal(base_size = 14)
}

plot_correlation_aqi_pollutant(aqi_state_year, pollution_state_year,
                               2018, "no2_aqi", "no2_mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



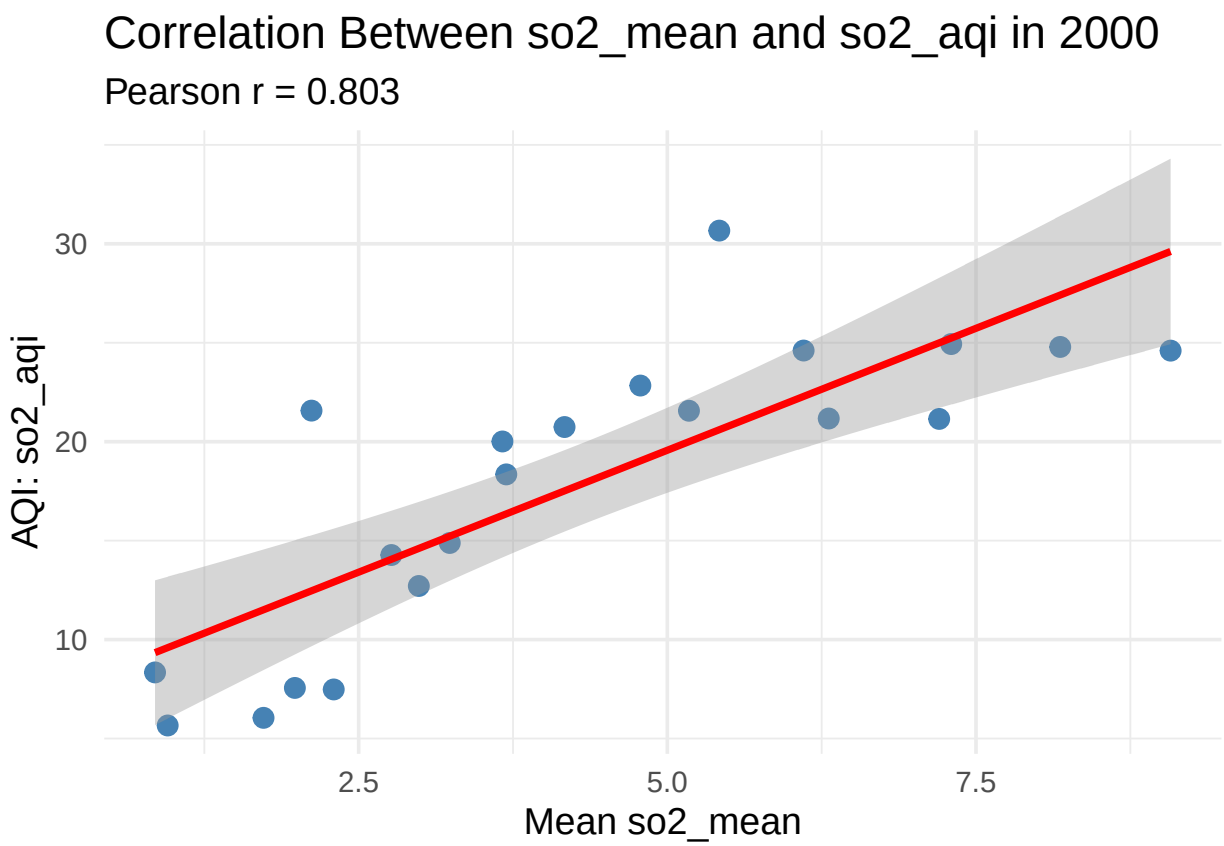

```
plot_correlation_aqi_pollutant(aqi_state_year, pollution_state_year,
                               2010, "so2_aqi", "so2_mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



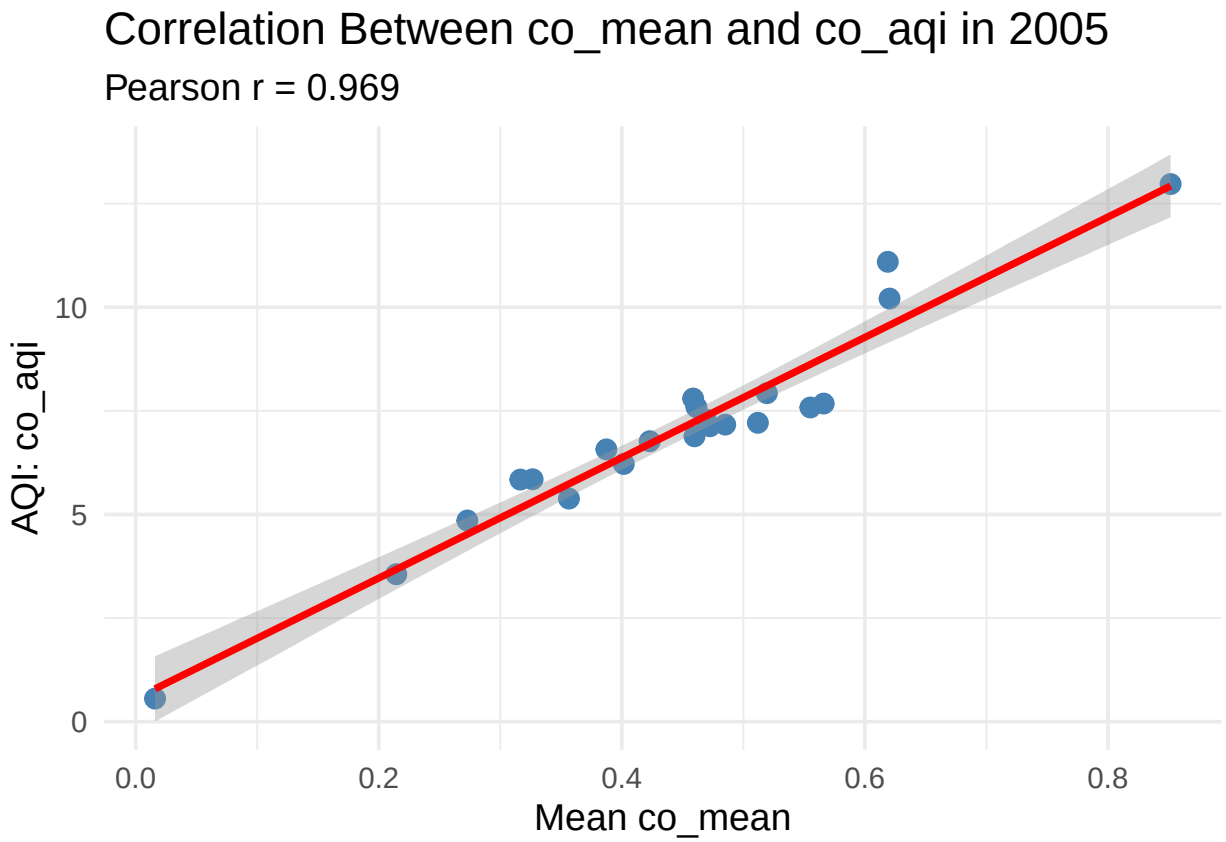

```
plot_correlation_aqi_pollutant(aqi_state_year, pollution_state_year,
                               2000, "so2_aqi", "so2_mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
plot_correlation_aqi_pollutant(aqi_state_year, pollution_state_year,
                               2005, "co_aqi", "co_mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Creating the final dataset

```
state_map <- data.frame(
  state_abb = state.abb,
  state_name = state.name,
  stringsAsFactors = FALSE
)

# Add DC manually (not included in state.name/state.abb defaults)
state_map <- rbind(
  state_map,
  data.frame(state_abb = "DC", state_name = "District Of Columbia", stringsAsFactors = FALSE)
)

asthma_state_year_fixed <- all_data %>%
  mutate(state = trimws(state)) %>%
  left_join(state_map, by = c("state" = "state_abb")) %>%
  mutate(state = state_name) %>%
  select(-state_name)

unique(asthma_state_year_fixed$state)[1:10]
```

```
##  [1] "Alabama"              "Alaska"               "Arizona"
##  [4] "Arkansas"             "California"           "Colorado"
```

```
##  [7] "Connecticut"          "Delaware"             "District Of Columbia"
## [10] "Florida"
```

```
length(intersect(unique(asthma_state_year_fixed$state),
                 unique(aqi_state_year$state)))
```

```
## [1] 48
```

```
full_data <- asthma_state_year_fixed %>%
  inner_join(aqi_state_year, by = c("state", "year")) %>%
  inner_join(pollution_state_year, by = c("state", "year"))

glimpse(full_data)
```

```
## Rows: 690
## Columns: 12
## $ state      <chr> "Arizona", "California", "Colorado", "District Of Columbia"~
## $ prevalence <dbl> 8.6, 7.3, 6.6, 7.9, 5.7, 7.9, 8.1, 7.8, 7.8, 5.0, 7.3, 7.2,~
## $ se         <dbl> 1.17, 0.47, 0.58, 0.75, 0.37, 0.75, 0.58, 0.46, 0.45, 0.35,~
## $ year       <dbl> 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000,~
## $ o3_aqi     <dbl> 44.43750, 36.54211, 30.34160, 31.04132, 44.72753, 26.13656,~
## $ co_aqi     <dbl> 15.169318, 11.267717, 11.559229, 18.584022, 10.342697, 11.0~
## $ so2_aqi    <dbl> 7.479545, 6.044754, 21.570248, 24.787879, 7.556180, 24.6059~
## $ no2_aqi    <dbl> 47.12727, 29.99034, 32.90358, 37.29477, 25.57303, 42.09957,~
## $ o3_mean    <dbl> 0.02613405, 0.02547947, 0.01938080, 0.01934785, 0.02923929,~
## $ co_mean    <dbl> 0.7460748, 0.6473329, 0.6152363, 1.1797488, 0.7268844, 0.70~
## $ so2_mean   <dbl> 2.2979704, 1.7293866, 2.1180400, 8.1822340, 1.9832440, 6.10~
## $ no2_mean   <dbl> 26.492344, 17.577458, 14.981307, 22.719222, 12.453038, 26.4~
```

```
nrow(full_data)
```

```
## [1] 690
```

```
glimpse(full_data)
```

```
## Rows: 690
## Columns: 12
## $ state      <chr> "Arizona", "California", "Colorado", "District Of Columbia"~
## $ prevalence <dbl> 8.6, 7.3, 6.6, 7.9, 5.7, 7.9, 8.1, 7.8, 7.8, 5.0, 7.3, 7.2,~
## $ se         <dbl> 1.17, 0.47, 0.58, 0.75, 0.37, 0.75, 0.58, 0.46, 0.45, 0.35,~
## $ year       <dbl> 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000,~
## $ o3_aqi     <dbl> 44.43750, 36.54211, 30.34160, 31.04132, 44.72753, 26.13656,~
## $ co_aqi     <dbl> 15.169318, 11.267717, 11.559229, 18.584022, 10.342697, 11.0~
## $ so2_aqi    <dbl> 7.479545, 6.044754, 21.570248, 24.787879, 7.556180, 24.6059~
## $ no2_aqi    <dbl> 47.12727, 29.99034, 32.90358, 37.29477, 25.57303, 42.09957,~
## $ o3_mean    <dbl> 0.02613405, 0.02547947, 0.01938080, 0.01934785, 0.02923929,~
## $ co_mean    <dbl> 0.7460748, 0.6473329, 0.6152363, 1.1797488, 0.7268844, 0.70~
## $ so2_mean   <dbl> 2.2979704, 1.7293866, 2.1180400, 8.1822340, 1.9832440, 6.10~
## $ no2_mean   <dbl> 26.492344, 17.577458, 14.981307, 22.719222, 12.453038, 26.4~
```

```r
nrow(full_data)
```

```
## [1] 690
```

```r
unique(full_data$state)[1:20]
```

```
##  [1] "Arizona"              "California"           "Colorado"
##  [4] "District Of Columbia" "Florida"              "Illinois"
##  [7] "Indiana"              "Kansas"               "Kentucky"
## [10] "Louisiana"            "Michigan"             "Missouri"
## [13] "New Jersey"           "New York"             "North Carolina"
## [16] "Oklahoma"             "Pennsylvania"         "Texas"
## [19] "Virginia"             "Washington"
```

```r
range(full_data$year)
```

```
## [1] 2000 2020
```

Linear Regression

#multi-pollutant model ‘

```r
library(plm)
```

```
##
## Attaching package: 'plm'

## The following objects are masked from 'package:dplyr':
##
##     between, lag, lead
```

```r
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:purrr':
##
##     some

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
# 1. Fixed-effects model
model_pollution_plm <- plm(
  prevalence ~ o3_mean + no2_mean + so2_mean + co_mean,
  data = full_data,
  index = c("state","year"),
  model = "within"
)
summary(model_pollution_plm)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ o3_mean + no2_mean + so2_mean + co_mean,
##     data = full_data, model = "within", index = c("state", "year"))
##
## Unbalanced Panel: n = 48, T = 5-21, N = 690
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -2.6113824 -0.4272051  0.0037945  0.4588446  2.4437792
##
## Coefficients:
##            Estimate Std. Error t-value  Pr(>|t|)
## o3_mean  -54.855389  10.084807 -5.4394 7.625e-08 ***
## no2_mean  -0.055068   0.014974 -3.6775 0.0002554 ***
## so2_mean  -0.219978   0.032910 -6.6843 5.068e-11 ***
## co_mean   -0.675983   0.299916 -2.2539 0.0245404 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    470.66
## Residual Sum of Squares: 321.3
## R-Squared:      0.31734
## Adj. R-Squared: 0.26277
## F-statistic: 74.1458 on 4 and 638 DF, p-value: < 2.22e-16
```

```r
# 2. VIF using pooled OLS
model_pollution_pooled <- plm(
  prevalence ~ o3_mean + no2_mean + so2_mean + co_mean,
  data = full_data,
  index = c("state","year"),
  model = "pooling"
)
vif(model_pollution_pooled)
```

```
##  o3_mean no2_mean so2_mean  co_mean
## 1.222089 2.110214 1.820768 1.754870
```

#using plm functionality for individual pollutants

```r
library(plm)
```

```r
model_o3_plm <- plm(
  prevalence ~ o3_mean,
  data = full_data,
  index = c("state", "year"),
  model = "within"
)

summary(model_o3_plm)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ o3_mean, data = full_data, model = "within",
##     index = c("state", "year"))
##
## Unbalanced Panel: n = 48, T = 5-21, N = 690
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -2.854461 -0.497691  0.035742  0.544118  2.734254
##
## Coefficients:
##         Estimate Std. Error t-value Pr(>|t|)
## o3_mean  -8.6569    10.9364 -0.7916   0.4289
##
## Total Sum of Squares:    470.66
## Residual Sum of Squares: 470.2
## R-Squared:      0.00097654
## Adj. R-Squared: -0.073833
## F-statistic: 0.626571 on 1 and 641 DF, p-value: 0.42891
```

```r
model_co_plm <- plm(
  prevalence ~ co_mean,
  data = full_data,
  index = c("state", "year"),
  model = "within"
)

summary(model_co_plm)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ co_mean, data = full_data, model = "within",
##     index = c("state", "year"))
##
## Unbalanced Panel: n = 48, T = 5-21, N = 690
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -3.292037 -0.415193  0.027209  0.495331  2.346571
##
```

```
## Coefficients:
##         Estimate Std. Error t-value  Pr(>|t|)
## co_mean -2.69831    0.23565 -11.451 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    470.66
## Residual Sum of Squares: 390.73
## R-Squared:      0.16981
## Adj. R-Squared: 0.10765
## F-statistic: 131.116 on 1 and 641 DF, p-value: < 2.22e-16
```

```
model_so2_plm <- plm(
  prevalence ~ so2_mean,
  data = full_data,
  index = c("state", "year"),
  model = "within"
)

summary(model_so2_plm)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ so2_mean, data = full_data, model = "within",
##     index = c("state", "year"))
##
## Unbalanced Panel: n = 48, T = 5-21, N = 690
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -2.799260 -0.433350 -0.015511  0.455359  2.524664
##
## Coefficients:
##          Estimate Std. Error t-value  Pr(>|t|)
## so2_mean -0.33460    0.02156  -15.52 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    470.66
## Residual Sum of Squares: 342.11
## R-Squared:      0.27313
## Adj. R-Squared: 0.2187
## F-statistic: 240.863 on 1 and 641 DF, p-value: < 2.22e-16
```

```
model_no2_plm <- plm(
  prevalence ~ no2_mean,
  data = full_data,
  index = c("state", "year"),
  model = "within"
)

summary(model_no2_plm)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ no2_mean, data = full_data, model = "within",
##     index = c("state", "year"))
##
## Unbalanced Panel: n = 48, T = 5-21, N = 690
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -2.962831 -0.431988  0.010168  0.488949  2.296582
##
## Coefficients:
##            Estimate Std. Error t-value  Pr(>|t|)
## no2_mean -0.1196606  0.0096492 -12.401 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    470.66
## Residual Sum of Squares: 379.59
## R-Squared:      0.1935
## Adj. R-Squared: 0.1331
## F-statistic: 153.788 on 1 and 641 DF, p-value: < 2.22e-16
```

pollution trends vs asthma trends

Compute Pollution and Asthma Trends for Each State

We will do this using per-state linear trends:

For each state, fit:

Pollutant (t) $= \alpha + \beta$t Asthma (t) $= \gamma + \delta$t

Then compare the slopes β and δ across states.

Which asks: In states where pollution declined faster, did asthma prevalence also decline faster?

```
library(dplyr)
library(purrr)
library(broom)
```

```
## Registered S3 method overwritten by 'broom':
##   method    from
##   nobs.felm lfe
```

```
o3_slopes <- full_data %>%
  group_by(state) %>%
  do(tidy(lm(o3_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, o3_slope = estimate)

no2_slopes <- full_data %>%
  group_by(state) %>%
```

```
  do(tidy(lm(no2_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, no2_slope = estimate)

so2_slopes <- full_data %>%
  group_by(state) %>%
  do(tidy(lm(so2_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, so2_slope = estimate)

co_slopes <- full_data %>%
  group_by(state) %>%
  do(tidy(lm(co_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, co_slope = estimate)
```

```
asthma_slopes <- full_data %>%
  group_by(state) %>%
  do(tidy(lm(prevalence ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, asthma_slope = estimate)
```

```
trend_data <- asthma_slopes %>%
  left_join(o3_slopes, by = "state") %>%
  left_join(no2_slopes, by = "state") %>%
  left_join(so2_slopes, by = "state") %>%
  left_join(co_slopes, by = "state")

trend_data
```

```
## # A tibble: 48 x 6
## # Groups:   state [48]
##    state                asthma_slope   o3_slope no2_slope so2_slope co_slope
##    <chr>                       <dbl>      <dbl>     <dbl>     <dbl>    <dbl>
##  1 Alabama                   0.0774  0.00136      -0.358  -0.00291  0.0156
##  2 Alaska                    0.246   0.00110      -0.264  -0.161   -0.0224
##  3 Arizona                   0.0826  0.000377     -0.657  -0.0883  -0.0209
##  4 Arkansas                  0.154  -0.000188     -0.242  -0.139   -0.0168
##  5 California                0.0618  0.000135     -0.410  -0.0826  -0.0162
##  6 Colorado                  0.0968  0.000560     -0.267  -0.0889  -0.0176
##  7 Connecticut               0.119  -0.0000773    -0.117  -0.154   -0.0109
##  8 Delaware                  0.0636 -0.000136     -0.210  -0.0835  -0.00140
##  9 District Of Columbia      0.154   0.000447     -0.771  -0.401   -0.0515
## 10 Florida                   0.0917  0.0000631    -0.445  -0.0433  -0.00706
## # i 38 more rows
```

#Do states with faster pollution declines experience faster asthma declines?

```
summary(lm(asthma_slope ~ o3_slope, data = trend_data))
```

```
##
```

```
## Call:
## lm(formula = asthma_slope ~ o3_slope, data = trend_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.11355 -0.03469 -0.01360  0.02791  0.25465
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.109536   0.009678  11.319 6.86e-15 ***
## o3_slope    -23.159424  19.984207  -1.159    0.252
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06696 on 46 degrees of freedom
## Multiple R-squared:  0.02837,    Adjusted R-squared:  0.007245
## F-statistic: 1.343 on 1 and 46 DF,  p-value: 0.2525
```

```
summary(lm(asthma_slope ~ no2_slope, data = trend_data))
```

```
##
## Call:
## lm(formula = asthma_slope ~ no2_slope, data = trend_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.10671 -0.03797 -0.01222  0.02945  0.24988
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.1102742  0.0122081   9.033 9.33e-12 ***
## no2_slope   0.0005573  0.0249377   0.022    0.982
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06793 on 46 degrees of freedom
## Multiple R-squared:  1.086e-05,  Adjusted R-squared:  -0.02173
## F-statistic: 0.0004994 on 1 and 46 DF,  p-value: 0.9823
```

```
summary(lm(asthma_slope ~ so2_slope, data = trend_data))
```

```
##
## Call:
## lm(formula = asthma_slope ~ so2_slope, data = trend_data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102175 -0.038854 -0.009809  0.026933  0.255044
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10289    0.01397   7.367 2.57e-09 ***
## so2_slope   -0.05393    0.07474  -0.722    0.474
```

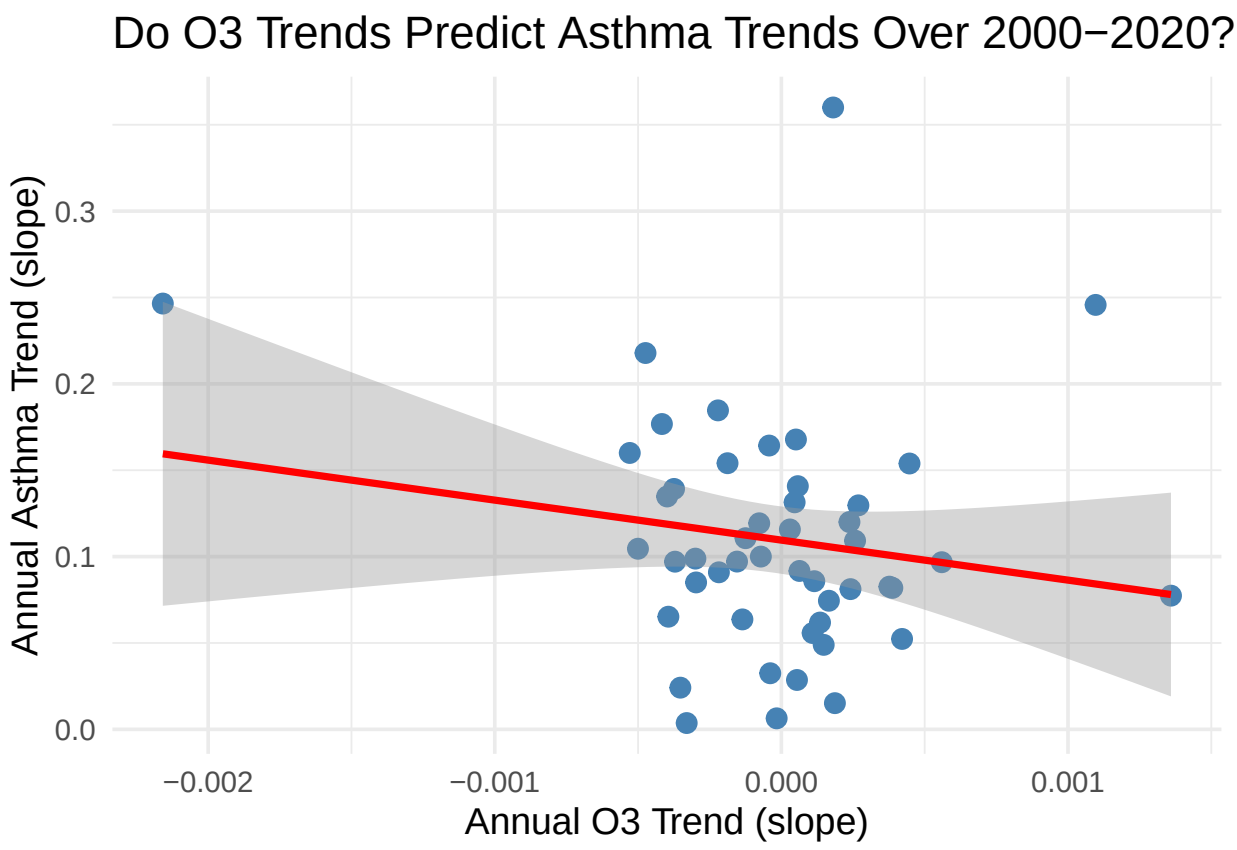
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06755 on 46 degrees of freedom
## Multiple R-squared:  0.01119,    Adjusted R-squared:  -0.0103
## F-statistic: 0.5207 on 1 and 46 DF,  p-value: 0.4742
```

```
summary(lm(asthma_slope ~ co_slope, data = trend_data))
```

```
##
## Call:
## lm(formula = asthma_slope ~ co_slope, data = trend_data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.126449 -0.036672 -0.002976  0.026982  0.207627
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.09793    0.01182   8.283 1.13e-10 ***
## co_slope    -1.13548    0.65573  -1.732     0.09 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06582 on 46 degrees of freedom
## Multiple R-squared:  0.0612, Adjusted R-squared:  0.04079
## F-statistic: 2.999 on 1 and 46 DF,  p-value: 0.09004
```

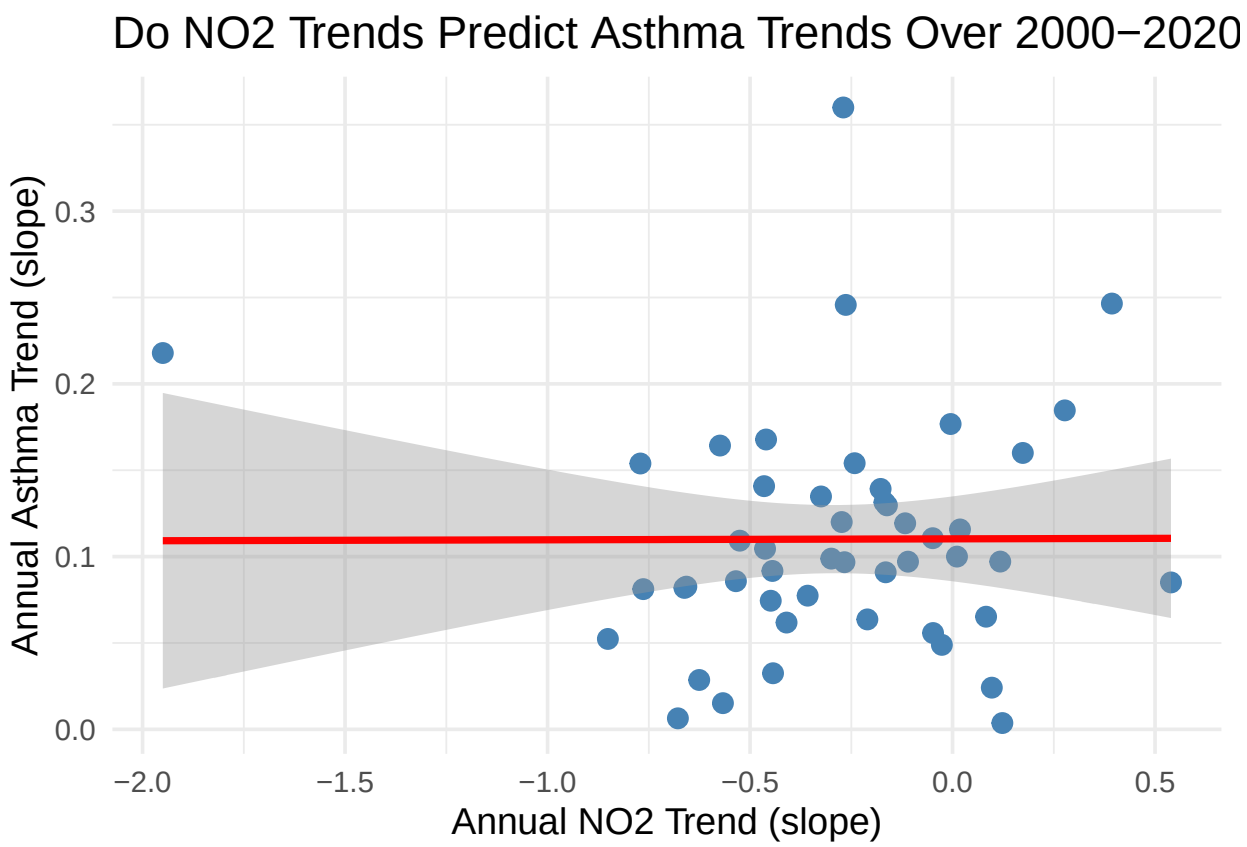
```
ggplot(trend_data, aes(x = o3_slope, y = asthma_slope)) +
  geom_point(size = 3, color = "steelblue") +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(
    title = "Do O3 Trends Predict Asthma Trends Over 2000-2020?",
    x = "Annual O3 Trend (slope)",
    y = "Annual Asthma Trend (slope)"
  ) +
  theme_minimal(base_size = 14)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



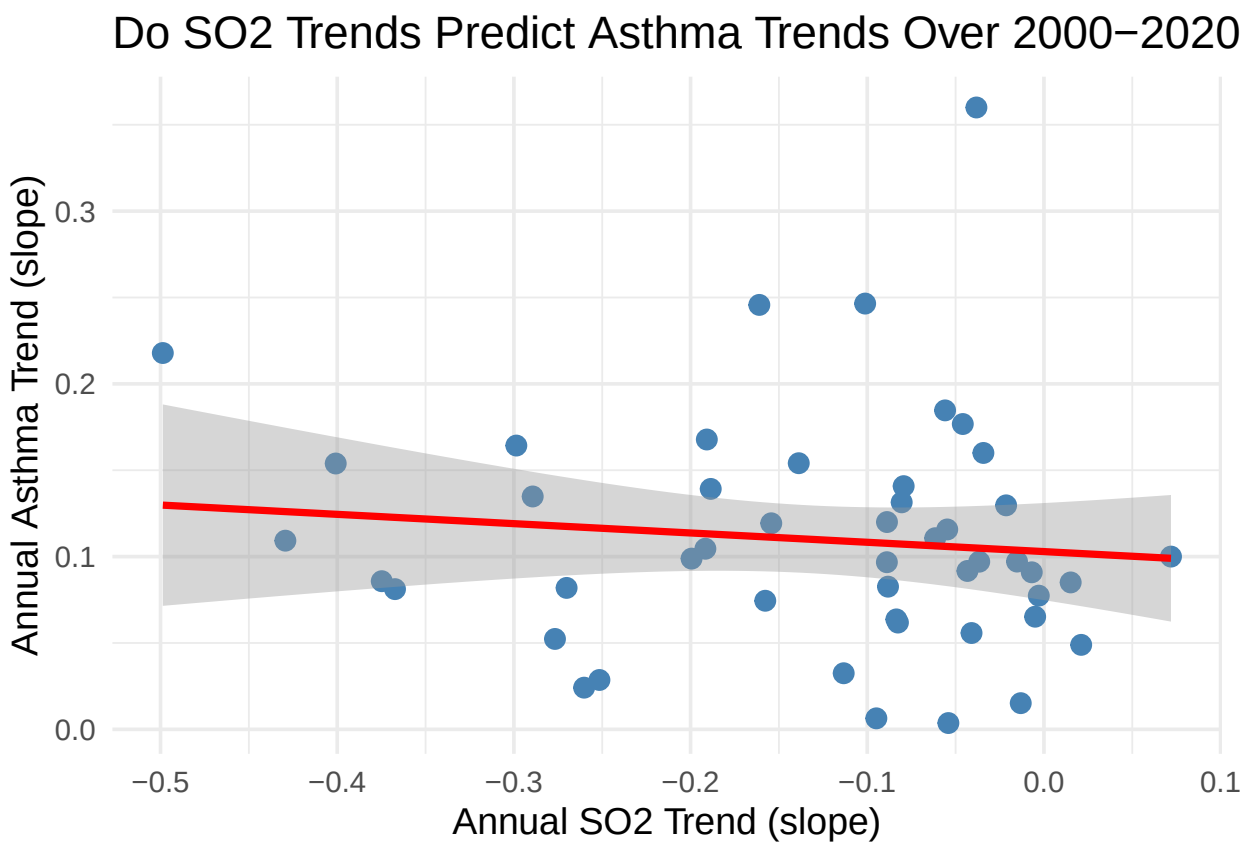

```
ggplot(trend_data, aes(x = no2_slope, y = asthma_slope)) +
  geom_point(size = 3, color = "steelblue") +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(
    title = "Do NO2 Trends Predict Asthma Trends Over 2000-2020?",
    x = "Annual NO2 Trend (slope)",
    y = "Annual Asthma Trend (slope)"
  ) +
  theme_minimal(base_size = 14)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



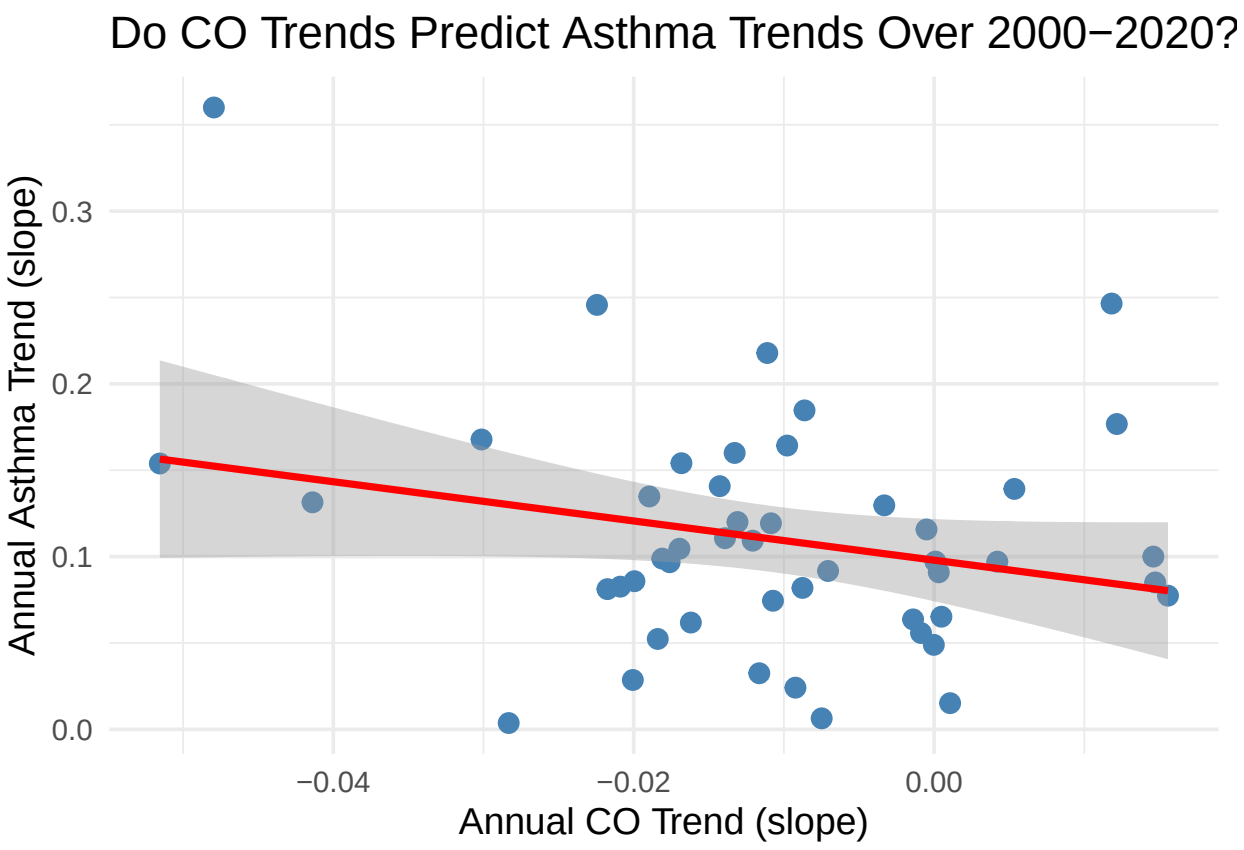

```
ggplot(trend_data, aes(x = so2_slope, y = asthma_slope)) +
  geom_point(size = 3, color = "steelblue") +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(
    title = "Do SO2 Trends Predict Asthma Trends Over 2000-2020?",
    x = "Annual SO2 Trend (slope)",
    y = "Annual Asthma Trend (slope)"
  ) +
  theme_minimal(base_size = 14)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
ggplot(trend_data, aes(x = co_slope, y = asthma_slope)) +
  geom_point(size = 3, color = "steelblue") +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(
    title = "Do CO Trends Predict Asthma Trends Over 2000-2020?",
    x = "Annual CO Trend (slope)",
    y = "Annual Asthma Trend (slope)"
  ) +
  theme_minimal(base_size = 14)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Selecting 10 most polluted states

```
library(dplyr)

# List of pollutant mean columns (modify as needed)
pollutants <- c("o3_mean", "no2_mean", "so2_mean", "co_mean")

# Compute average pollution across 2000-2020 for each state
state_pollution <- full_data %>%
  group_by(state) %>%
  summarise(across(all_of(pollutants), mean, na.rm = TRUE)) %>%
  ungroup()
```

```
## Warning: There was 1 warning in `summarise()`.
## i In argument: `across(all_of(pollutants), mean, na.rm = TRUE)`.
## i In group 1: `state = "Alabama"`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
```

```
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

```
# Create a combined pollution index (average of standardized pollutants)
state_pollution <- state_pollution %>%
  mutate(pollution_index = rowMeans(scale(across(all_of(pollutants)))))
```

```
top_states <- state_pollution %>%
  arrange(desc(pollution_index)) %>%
  slice(1:10) %>%
  pull(state)

top_states
```

```
##  [1] "District Of Columbia" "Michigan"             "New York"
##  [4] "Arizona"              "New Jersey"           "Indiana"
##  [7] "Colorado"             "Missouri"             "Wisconsin"
## [10] "Illinois"
```

```
highpoll_data <- full_data %>%
  filter(state %in% top_states)
```

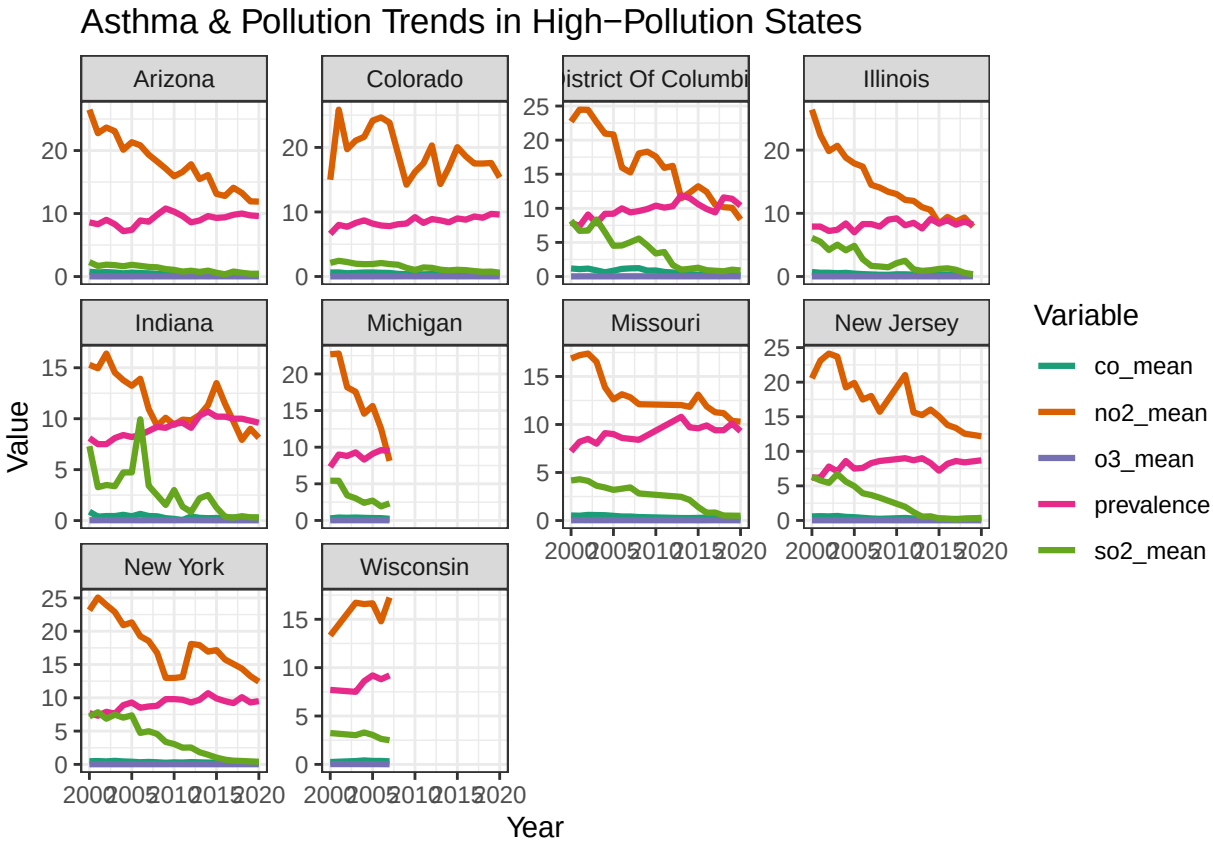
```
library(plm)

model_highpoll <- plm(
  prevalence ~ o3_mean + no2_mean + so2_mean + co_mean,
  data = highpoll_data,
  index = c("state","year"),
  model = "within"
)

summary(model_highpoll)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ o3_mean + no2_mean + so2_mean + co_mean,
##     data = highpoll_data, model = "within", index = c("state",
##         "year"))
##
## Unbalanced Panel: n = 10, T = 6-21, N = 174
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -1.9561481 -0.3801070 -0.0071268  0.4028495  1.5906918
##
## Coefficients:
##            Estimate Std. Error t-value Pr(>|t|)
## o3_mean  -27.634993  16.558707 -1.6689  0.09709 .
## no2_mean  -0.044519   0.022408 -1.9867  0.04866 *
## so2_mean  -0.163757   0.044190 -3.7057  0.00029 ***
## co_mean   -1.116486   0.456312 -2.4468  0.01550 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    129.14
## Residual Sum of Squares: 67.517
## R-Squared:      0.47717
## Adj. R-Squared: 0.43469
## F-statistic: 36.5068 on 4 and 160 DF, p-value: < 2.22e-16
```

```r
library(ggplot2)
library(dplyr)
library(tidyr)

# convert dataset to long format for easy plotting
plot_data <- highpoll_data %>%
  select(state, year, prevalence, o3_mean, no2_mean, so2_mean, co_mean) %>%
  pivot_longer(cols = c(prevalence, o3_mean, no2_mean, so2_mean, co_mean),
               names_to = "variable", values_to = "value")

# facet plot for trends
ggplot(plot_data, aes(x = year, y = value, color = variable)) +
  geom_line(size = 1.1) +
  facet_wrap(~ state, scales = "free_y") +
  theme_bw() +
  labs(
    title = "Asthma & Pollution Trends in High-Pollution States",
    x = "Year",
    y = "Value",
    color = "Variable"
  ) +
  scale_color_brewer(palette = "Dark2")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



1. compairing the 10 most pollution states

```
library(dplyr)
library(broom)

# Compute slopes for asthma for the 10 polluted states
asthma_slopes_high <- highpoll_data %>%
  group_by(state) %>%
  do(tidy(lm(prevalence ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, asthma_slope = estimate)

# Compute pollution slopes for each pollutant
o3_slopes_high <- highpoll_data %>%
  group_by(state) %>%
  do(tidy(lm(o3_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, o3_slope = estimate)

no2_slopes_high <- highpoll_data %>%
  group_by(state) %>%
  do(tidy(lm(no2_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, no2_slope = estimate)

so2_slopes_high <- highpoll_data %>%
```

```
  group_by(state) %>%
  do(tidy(lm(so2_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, so2_slope = estimate)

co_slopes_high <- highpoll_data %>%
  group_by(state) %>%
  do(tidy(lm(co_mean ~ year, data = .))) %>%
  filter(term == "year") %>%
  select(state, co_slope = estimate)

# Final combined trend dataset
trend_high <- asthma_slopes_high %>%
  left_join(o3_slopes_high, by = "state") %>%
  left_join(no2_slopes_high, by = "state") %>%
  left_join(so2_slopes_high, by = "state") %>%
  left_join(co_slopes_high, by = "state")

trend_high
```

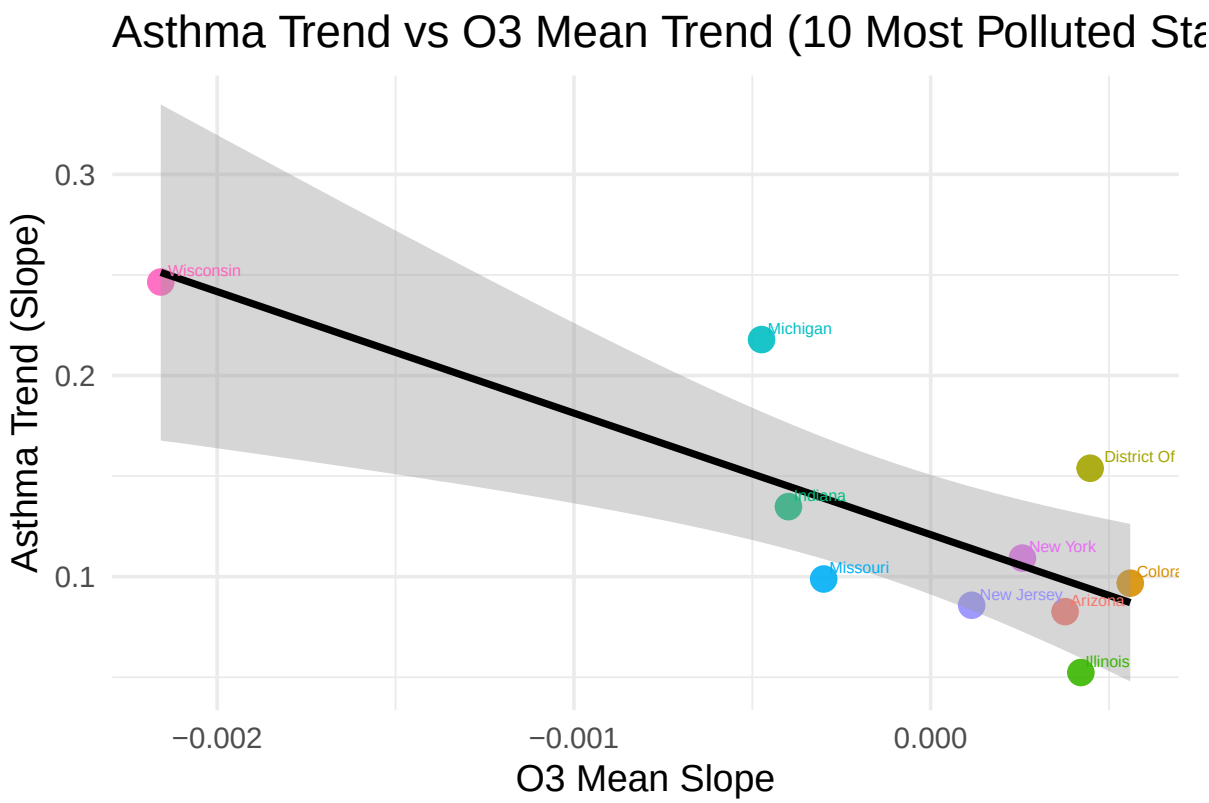
```
## # A tibble: 10 x 6
## # Groups:   state [10]
##    state                asthma_slope  o3_slope no2_slope so2_slope co_slope
##    <chr>                       <dbl>     <dbl>     <dbl>     <dbl>    <dbl>
##  1 Arizona                    0.0826  0.000377    -0.657   -0.0883  -0.0209
##  2 Colorado                   0.0968  0.000560    -0.267   -0.0889  -0.0176
##  3 District Of Columbia       0.154   0.000447    -0.771   -0.401   -0.0515
##  4 Illinois                   0.0523  0.000421    -0.851   -0.277   -0.0184
##  5 Indiana                    0.135  -0.000399    -0.325   -0.289   -0.0190
##  6 Michigan                   0.218  -0.000474    -1.95    -0.499   -0.0111
##  7 Missouri                   0.0988 -0.000300    -0.299   -0.199   -0.0181
##  8 New Jersey                 0.0858  0.000115    -0.535   -0.375   -0.0199
##  9 New York                   0.109   0.000257    -0.525   -0.429   -0.0121
## 10 Wisconsin                  0.246  -0.00216      0.393   -0.101    0.0118
```

```
library(ggplot2)

plot_trend_comparison <- function(data, pollutant_col, pollutant_label) {

  ggplot(data, aes_string(x = pollutant_col,
                          y = "asthma_slope",
                          color = "state")) +
    geom_point(size = 4, alpha = 0.9) +
    geom_smooth(method = "lm", se = TRUE, color = "black") +
    geom_text(aes(label = state),
              hjust = -0.1, vjust = -0.5, size = 2, show.legend = FALSE) +
    labs(
      title = paste("Asthma Trend vs", pollutant_label, "Trend (10 Most Polluted States)"),
      x = paste(pollutant_label, "Slope"),
      y = "Asthma Trend (Slope)"
    ) +
    theme_minimal(base_size = 14) +
    theme(
```

```
    legend.position = "none",
    plot.margin = margin(10, 10, 20, 20)
  )
  }


plot_trend_comparison(trend_high, "o3_slope",  "O3 Mean")
```

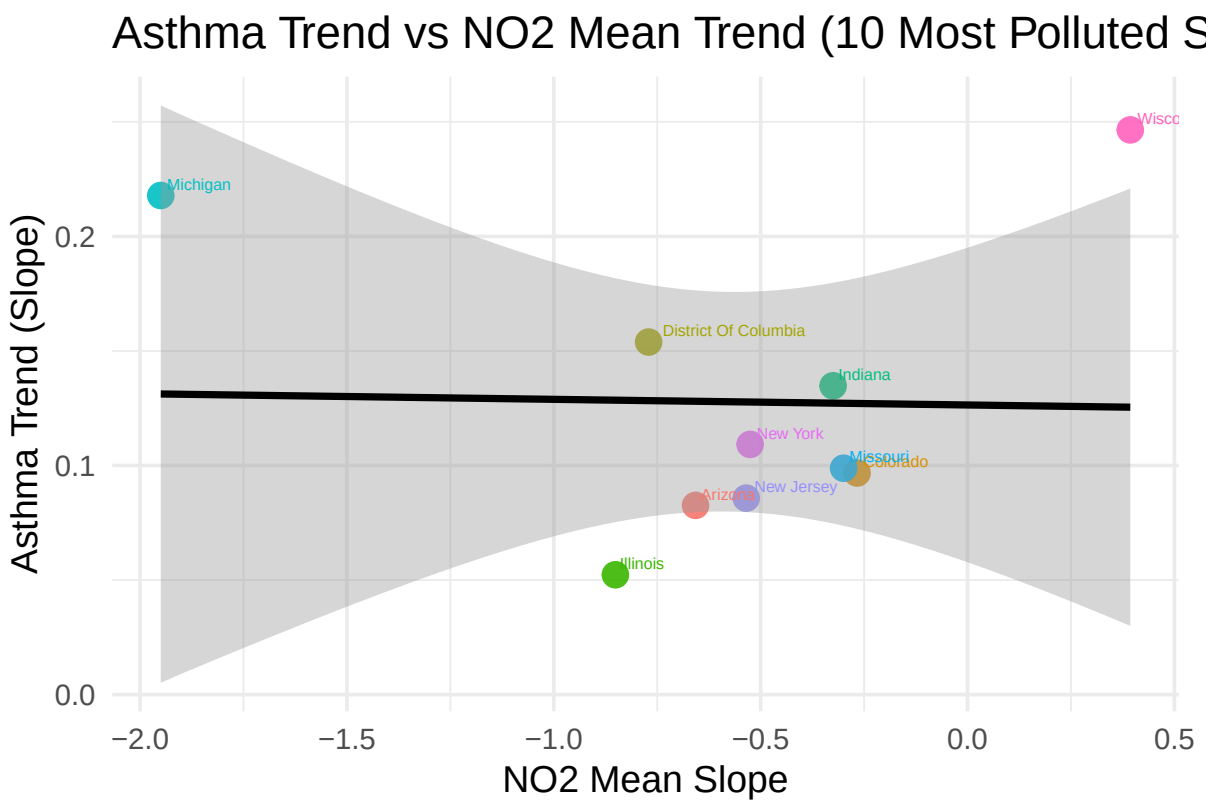
```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation idioms with `aes()`.
## i See also `vignette("ggplot2-in-packages")` for more information.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



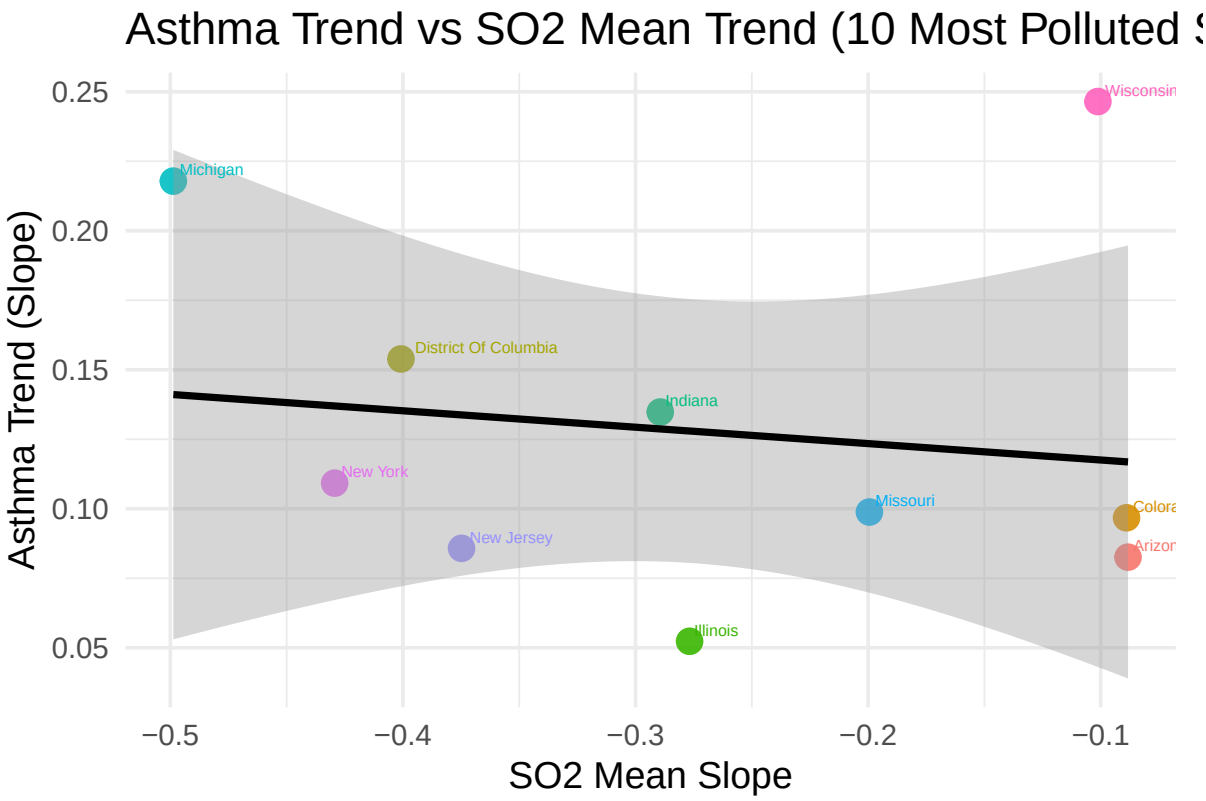

```
plot_trend_comparison(trend_high, "no2_slope", "NO2 Mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



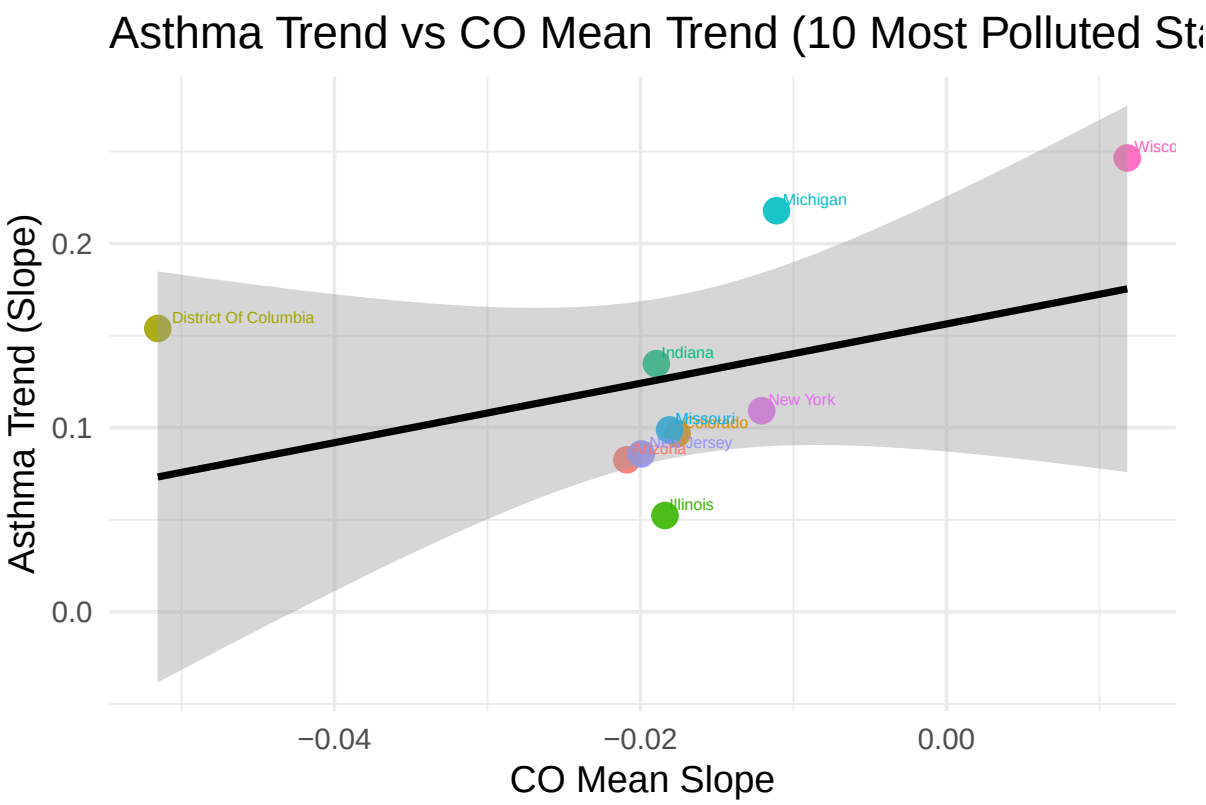

```
plot_trend_comparison(trend_high, "so2_slope", "SO2 Mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
plot_trend_comparison(trend_high, "co_slope",  "CO Mean")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



adding the income bracket

```
get_url <- function(year) {
  if (year >= 2000 & year <= 2009) {
    # 2 digit folder + "current"
    two_digit <- sprintf("%02d", year %% 100)
    paste0("https://www.cdc.gov/asthma/brfss/", two_digit, "/current/tableC7.htm")

  } else if (year == 2010) {
    # 2010 only: has /current/
    "https://www.cdc.gov/asthma/brfss/2010/current/tableC7.htm"

  } else if (year >= 2011 & year <= 2020) {
    # 2011+ uses only year, NO 'current' folder
    paste0("https://www.cdc.gov/asthma/brfss/", year, "/tableC7.htm")

  } else {
    stop("Year outside supported range 2000-2020.")
  }
}
```

```
library(rvest)
library(dplyr)
library(stringr)
library(purrr)
```

```r
library(janitor)

scrape_income_table <- function(year) {

  url <- get_url(year)
  message("Scraping: ", url)

  page <- read_html(url)

  # Extract all HTML tables - usually the first is the income table
  tables <- page %>% html_table(fill = TRUE)

  # if table is missing, return empty df
  if (length(tables) == 0) return(data.frame())

  tbl <- tables[[1]] %>% clean_names()

  # Standardize column names (table structures vary slightly by year)
  names(tbl)[1:5] <- c("state", "income", "sample_size", "prevalence", "se")

  # Clean table
  tbl <- tbl %>%
    filter(
      !state %in% c("U.S. Total", "U.S. Total**", "", NA),
      !str_detect(state, "Total")   # removes "State Total"
    ) %>%
    mutate(
      year = year,
      prevalence = suppressWarnings(as.numeric(prevalence)),
      se = suppressWarnings(as.numeric(se))
    )

  return(tbl)
}
```

```r
years <- 2000:2020

income_data <- map_df(years, ~ tryCatch(scrape_income_table(.x), error = function(e) NULL))
```

```
## Scraping: https://www.cdc.gov/asthma/brfss/00/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/01/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/02/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/03/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/04/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/05/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/06/current/tableC7.htm
```

```
## Scraping: https://www.cdc.gov/asthma/brfss/07/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/08/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/09/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2010/current/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2011/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2012/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2013/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2014/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2015/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2016/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2017/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2018/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2019/tableC7.htm

## Scraping: https://www.cdc.gov/asthma/brfss/2020/tableC7.htm
```

```
glimpse(income_data)
```

```
## Rows: 4,812
## Columns: 16
## $ state                          <chr> "AL", "AL", "AL", "AL", "AL", "AK", "AK~
## $ income                         <chr> "< $15,000", "$15-$24,999", "$25-$49,99~
## $ sample_size                    <chr> "357", "390", "687", "275", "273", "169~
## $ prevalence                     <dbl> 7.4, 10.2, 6.1, 3.5, 1.5, 8.5, 10.9, 7.~
## $ se                             <dbl> 1.43, 1.85, 1.01, 1.06, NA, 2.49, 3.58,~
## $ lower_95_percent_ci            <dbl> 4.6, 6.5, 4.1, 1.5, NA, 3.6, 3.8, 4.2, ~
## $ upper_95_percent_ci            <dbl> 10.3, 13.8, 8.1, 5.6, NA, 13.5, 17.9, 9~
## $ prevalence_number              <chr> "34,130", "58,015", "63,038", "15,564",~
## $ lower_95_percent_ci_2          <chr> "21,239", "36,221", "42,186", "6,283", ~
## $ upper_95_percent_ci_2          <chr> "47,022", "79,808", "83,890", "24,845",~
## $ year                           <int> 2000, 2000, 2000, 2000, 2000, 2000, 200~
## $ x95_percent_ci_percent         <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ x95_percent_ci_number          <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ x                              <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ weighted_number                <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ x95_percent_ci_weighted_number <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
```

#CLEANING THE DATA SET

```r
library(dplyr)
library(stringr)

income_clean <- income_data %>%
  # Keep the useful columns
  select(
    state,
    income,
    sample_size,
    prevalence,
    se,
    year
  ) %>%

  # Convert sample_size to numeric
  mutate(
    sample_size = str_replace_all(sample_size, ",", ""),
    sample_size = as.numeric(sample_size)
  ) %>%

  # Standardize income labels (optional)
  mutate(
    income = str_trim(income),
    income = case_when(
      income == "< $15,000" ~ "<15k",
      income == "$15-$24,999" ~ "15-25k",
      income == "$25-$49,999" ~ "25-50k",
      income == "$50-$74,999" ~ "50-75k",
      income == ">= $75,000" ~ "75k+",
      TRUE ~ income
    )
  ) %>%

  # Remove rows with missing prevalence (rare)
  filter(!is.na(prevalence))

glimpse(income_clean)
```

```
## Rows: 4,795
## Columns: 6
## $ state       <chr> "AL", "AL", "AL", "AL", "AL", "AK", "AK", "AK", "AK", "AK"~
## $ income      <chr> "<15k", "15-25k", "25-50k", "50-75k", ">=$75,000", "<15k",~
## $ sample_size <dbl> 357, 390, 687, 275, 273, 169, 311, 646, 373, 413, 188, 608~
## $ prevalence  <dbl> 7.4, 10.2, 6.1, 3.5, 1.5, 8.5, 10.9, 7.0, 3.5, 5.2, 17.7, ~
## $ se          <dbl> 1.43, 1.85, 1.01, 1.06, NA, 2.49, 3.58, 1.41, 1.08, 1.25, ~
## $ year        <int> 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000~
```

```r
head(income_clean)
```

```
## # A tibble: 6 x 6
##   state income    sample_size prevalence    se  year
##   <chr> <chr>           <dbl>      <dbl> <dbl> <int>
```

```
## 1 AL    <15k              357        7.4  1.43  2000
## 2 AL    15-25k            390       10.2  1.85  2000
## 3 AL    25-50k            687        6.1  1.01  2000
## 4 AL    50-75k            275        3.5  1.06  2000
## 5 AL    >=$75,000         273        1.5 NA     2000
## 6 AK    <15k              169        8.5  2.49  2000
```

```
income_clean <- income_clean %>%
  mutate(
    state = state.name[match(state, state.abb)]
  )
head(unique(income_clean$state), 10)
```

```
##  [1] "Alabama"     "Alaska"      "Arizona"     "Arkansas"    "California"
##  [6] "Colorado"    "Connecticut" "Delaware"    NA            "Florida"
```

#Encoding the income as an order parameter and merging all the data sets

```
income_clean <- income_clean %>%
  mutate(
    income_factor = factor(
      income,
      levels = c("<15k", "15-25k", "25-50k", "50-75k", "75k+"),
      ordered = TRUE
    )
  )


full_merged <- income_clean %>%
  left_join(
    pollution_state_year,
    by = c("state", "year")
  )


income_pollution <- income_clean %>%
  left_join(pollution_state_year, by = c("state", "year"))


names(income_pollution)
```

```
##  [1] "state"         "income"        "sample_size"   "prevalence"
##  [5] "se"            "year"          "income_factor" "o3_mean"
##  [9] "co_mean"       "so2_mean"      "no2_mean"
```

#Centering pollutant variables

```
poll_vars <- c("pm10_mean","o3_mean","no2_mean","so2_mean","co_mean")
poll_vars <- poll_vars[poll_vars %in% names(income_pollution)]
poll_vars
```

```
## [1] "o3_mean"  "no2_mean" "so2_mean" "co_mean"
```

```r
income_pollution <- income_pollution %>%
  mutate(
    across(
      all_of(poll_vars),
      ~ as.numeric(scale(.x, center = TRUE, scale = FALSE)),
      .names = "{.col}_c"
    )
  )

summary(income_pollution)
```

```
##     state              income           sample_size      prevalence
##  Length:4795        Length:4795        Min.   :   41   Min.   : 0.700
##  Class :character   Class :character   1st Qu.:  627   1st Qu.: 6.800
##  Mode  :character   Mode  :character   Median :  984   Median : 8.400
##                                        Mean   : 1233   Mean   : 9.216
##                                        3rd Qu.: 1535   3rd Qu.:10.900
##                                        Max.   :10487   Max.   :27.700
##
##        se             year      income_factor    o3_mean
##  Min.   :0.35   Min.   :2000   <15k  :959    Min.   :0.01112
##  1st Qu.:0.90   1st Qu.:2004   15-25k:959    1st Qu.:0.02569
##  Median :1.17   Median :2008   25-50k:959    Median :0.02818
##  Mean   :1.34   Mean   :2008   50-75k:959    Mean   :0.02832
##  3rd Qu.:1.60   3rd Qu.:2013   75k+  :  0    3rd Qu.:0.03092
##  Max.   :9.83   Max.   :2017   NA's  :959    Max.   :0.04502
##  NA's   :24                                  NA's   :2080
##     co_mean           so2_mean           no2_mean         o3_mean_c
##  Min.   :0.00284   Min.   :-0.02521   Min.   : 0.5427   Min.   :-0.01720
##  1st Qu.:0.22134   1st Qu.: 0.56193   1st Qu.: 7.4032   1st Qu.:-0.00264
##  Median :0.28875   Median : 1.07541   Median :11.0747   Median :-0.00014
##  Mean   :0.32075   Mean   : 1.72369   Mean   :11.4206   Mean   : 0.00000
##  3rd Qu.:0.39238   3rd Qu.: 2.42404   3rd Qu.:14.9813   3rd Qu.: 0.00260
##  Max.   :1.00000   Max.   : 9.94356   Max.   :33.2082   Max.   : 0.01670
##  NA's   :2080      NA's   :2080       NA's   :2080      NA's   :2080
##    no2_mean_c         so2_mean_c        co_mean_c
##  Min.   :-10.8779   Min.   :-1.7489   Min.   :-0.31791
##  1st Qu.: -4.0174   1st Qu.:-1.1618   1st Qu.:-0.09941
##  Median : -0.3459   Median :-0.6483   Median :-0.03199
##  Mean   :  0.0000   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.:  3.5607   3rd Qu.: 0.7004   3rd Qu.: 0.07164
##  Max.   : 21.7876   Max.   : 8.2199   Max.   : 0.67925
##  NA's   :2080       NA's   :2080      NA's   :2080
```

Base fixed effect model

```r
library(plm)

model_base <- plm(
  prevalence ~ income_factor + o3_mean_c + no2_mean_c + so2_mean_c + co_mean_c,
```

```
  data  = income_pollution,
  index = c("state", "year"),
  model = "within"
)
```

```
## Warning in pdata.frame(data, index = index, ...): duplicate couples (id-time) in resulting pdata.fra
##  to find out which, use, e.g., table(index(your_pdataframe), useNA = "ifany")
```

```
## Warning in pdata.frame(data, index = index, ...): at least one NA in at least one index dimension in
##  to find out which, use, e.g., table(index(your_pdataframe), useNA = "ifany")
```

```
summary(model_base)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ income_factor + o3_mean_c + no2_mean_c +
##     so2_mean_c + co_mean_c, data = income_pollution, model = "within",
##     index = c("state", "year"))
##
## Unbalanced Panel: n = 47, T = 8-72, N = 2172
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -8.115060 -1.189073 -0.041454  1.094670 10.623703
##
## Coefficients:
##                   Estimate Std. Error  t-value  Pr(>|t|)
## income_factor.L  -5.198632   0.083586 -62.1953 < 2.2e-16 ***
## income_factor.Q   1.580663   0.083586  18.9107 < 2.2e-16 ***
## income_factor.C  -0.053699   0.083586  -0.6424  0.520658
## o3_mean_c       -81.641575  15.146995  -5.3900 7.831e-08 ***
## no2_mean_c       -0.085001   0.022929  -3.7071  0.000215 ***
## so2_mean_c       -0.311376   0.049777  -6.2554 4.782e-10 ***
## co_mean_c        -0.472890   0.502058  -0.9419  0.346350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    24838
## Residual Sum of Squares: 8035
## R-Squared:      0.6765
## Adj. R-Squared: 0.66841
## F-statistic: 632.74 on 7 and 2118 DF, p-value: < 2.22e-16
```

```
income_pollution$income_factor <- factor(income_pollution$income_factor, ordered = FALSE)

levels(income_pollution$income_factor)
```

```
## [1] "<15k"   "15-25k" "25-50k" "50-75k"
```

```r
income_pollution$income_factor <- relevel(income_pollution$income_factor, ref = "<15k")


library(plm)

model_income_dummies <- plm(
  prevalence ~ income_factor + o3_mean_c + no2_mean_c + so2_mean_c + co_mean_c,
  data  = income_pollution,
  index = c("state", "year"),
  model = "within"
)
```

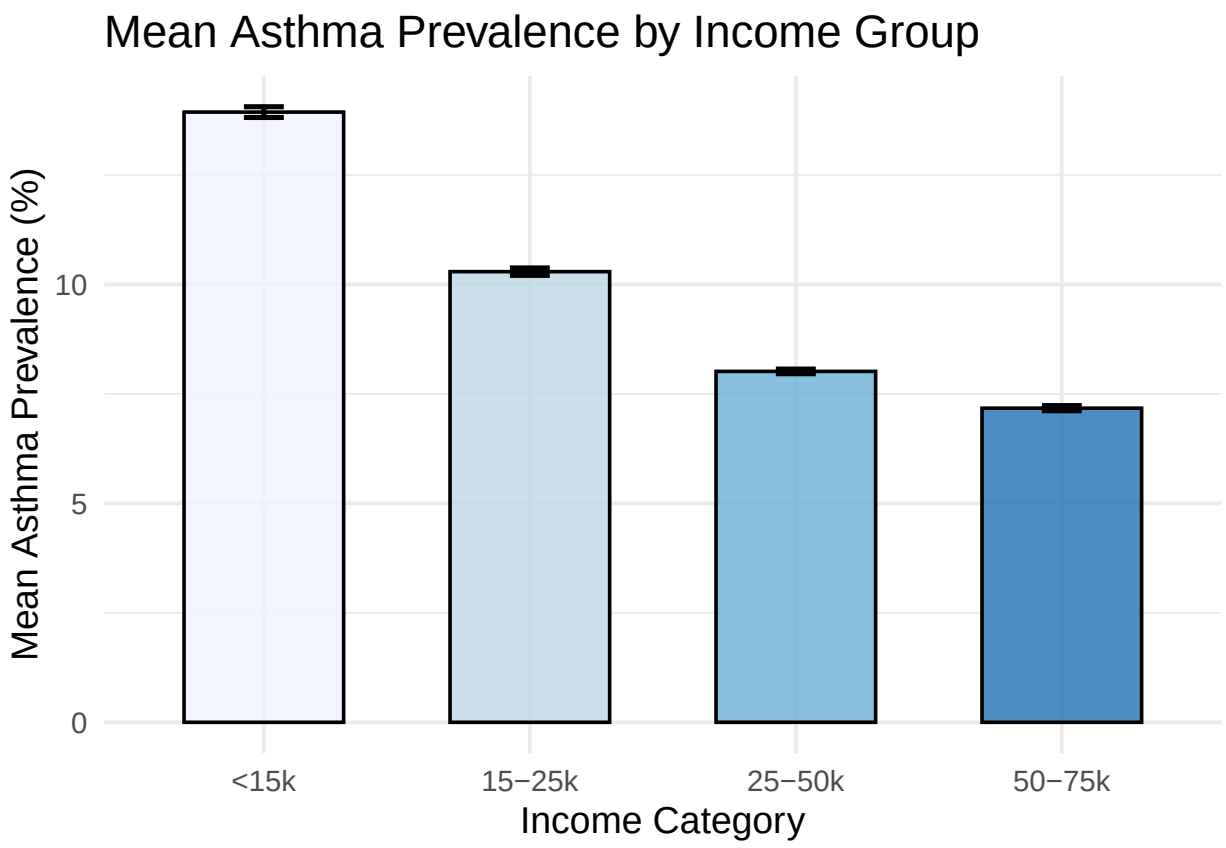
```
## Warning in pdata.frame(data, index = index, ...): duplicate couples (id-time) in resulting pdata.fra
##  to find out which, use, e.g., table(index(your_pdataframe), useNA = "ifany")
```

```
## Warning in pdata.frame(data, index = index, ...): at least one NA in at least one index dimension in
##  to find out which, use, e.g., table(index(your_pdataframe), useNA = "ifany")
```

```r
summary(model_income_dummies)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = prevalence ~ income_factor + o3_mean_c + no2_mean_c +
##     so2_mean_c + co_mean_c, data = income_pollution, model = "within",
##     index = c("state", "year"))
##
## Unbalanced Panel: n = 47, T = 8-72, N = 2172
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -8.115060 -1.189073 -0.041454  1.094670 10.623703
##
## Coefficients:
##                       Estimate Std. Error  t-value  Pr(>|t|)
## income_factor15-25k  -3.953591   0.118208 -33.4461 < 2.2e-16 ***
## income_factor25-50k  -6.206446   0.118208 -52.5045 < 2.2e-16 ***
## income_factor50-75k  -6.998711   0.118208 -59.2068 < 2.2e-16 ***
## o3_mean_c           -81.641575  15.146995  -5.3900 7.831e-08 ***
## no2_mean_c           -0.085001   0.022929  -3.7071  0.000215 ***
## so2_mean_c           -0.311376   0.049777  -6.2554 4.782e-10 ***
## co_mean_c            -0.472890   0.502058  -0.9419  0.346350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    24838
## Residual Sum of Squares: 8035
## R-Squared:      0.6765
## Adj. R-Squared: 0.66841
## F-statistic: 632.74 on 7 and 2118 DF, p-value: < 2.22e-16
```

```r
library(dplyr)
library(ggplot2)

income_summary <- income_pollution %>%
  filter(!is.na(income_factor)) %>%   # remove NA income levels
  group_by(income_factor) %>%
  summarise(
    mean_prev = mean(prevalence, na.rm = TRUE),
    se_prev   = sd(prevalence, na.rm = TRUE) / sqrt(n())
  )
ggplot(income_summary, aes(x = income_factor, y = mean_prev, fill = income_factor)) +
  geom_col(width = 0.6, color = "black", alpha = 0.8) +
  geom_errorbar(aes(ymin = mean_prev - se_prev, ymax = mean_prev + se_prev),
                width = 0.15, size = 0.9) +
  labs(
    title = "Mean Asthma Prevalence by Income Group",
    x = "Income Category",
    y = "Mean Asthma Prevalence (%)"
  ) +
  scale_fill_brewer(palette = "Blues") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "none")
```




```r
linearHypothesis(
  model_income_dummies,
  c(
```

```
    "income_factor15-25k = 0",
    "income_factor25-50k = 0",
    "income_factor50-75k = 0"
  ),
  vcov. = vcovHC(model_income_dummies, type = "HC1")
)
```

```
##
## Linear hypothesis test:
## income_factor15-25k = 0
## income_factor25-50k = 0
## income_factor50-75k = 0
##
## Model 1: restricted model
## Model 2: prevalence ~ income_factor + o3_mean_c + no2_mean_c + so2_mean_c +
##     co_mean_c
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1   2121
## 2   2118  3 471.51  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This project investigates national asthma prevalence trends and their relationship with air pollution and socioeconomic factor mainly the income across all U.S. states from 2000-2020. Asthma prevalence data were scraped from CDC BRFSS tables, while pollutant concentration data (O_{3}, NO_{2}, SO_{2}, CO) and AQI metrics were derived from EPA pollution datasets from Kaggle. Income-stratified asthma data were also incorporated to examine disparities across economic groups. After extensive cleaning and merging, a unified state-year panel dataset was constructed to facilitate visualization, correlation analysis, regression modeling, and long-term trend estimation.

Visual exploration reveals substantial state-to-state variation in asthma rates and strong downward trends in key pollutants—particularly NO_{2}, SO_{2}, and CO. Regression analyses demonstrate statistically significant associations between asthma prevalence and certain pollutants (notably CO and SO_{2}), while fixed-effects modeling highlights persistent socioeconomic inequalities independent of pollution exposure. Trend comparison models show that declines in pollution do not necessarily coincide with improvements in asthma prevalence, underscoring asthma's multifactorial nature. In sum, the project provides a comprehensive look at environmental and socioeconomic contributors to asthma and demonstrates how integrated datasets can be used to study population-level health outcomes.

Pollution has declined substantially, but asthma prevalence has not, indicating additional drivers beyond outdoor pollutants.

Geographic variability in asthma is pronounced and persistent.

Socioeconomic disparities are strong and consistent-low-income groups experience meaningfully higher asthma prevalence.

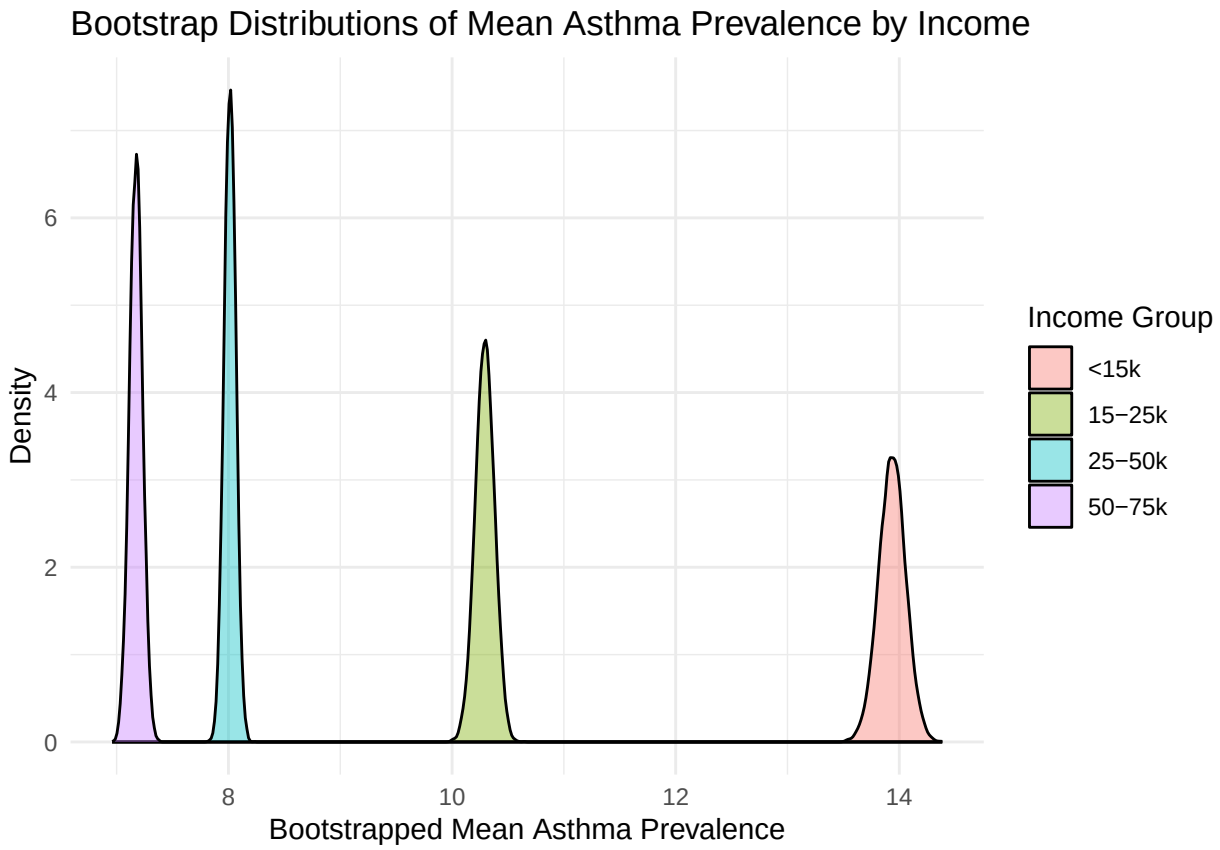
Pollutant-AQI correlations are extremely high, confirming dataset validity.

Regression and fixed-effects models identify statistically significant pollutant effects but explain only a modest portion of variation, especially compared to income.

```r
set.seed(123)
B <- 10000

# keep only rows with income category
tmp <- income_pollution %>% filter(!is.na(income_factor))

# bootstrap mean
boot_income_means <- tmp %>%
  group_by(income_factor) %>%
  summarise(
    boot_means = list(
      replicate(B, {
        boot_sample <- sample(prevalence, replace = TRUE)
        mean(boot_sample, na.rm = TRUE)
      })
    ),
    .groups = "drop"
  )


boot_income_means %>%
  rowwise() %>%
  mutate(
    lower95 = quantile(unlist(boot_means), 0.025),
    upper95 = quantile(unlist(boot_means), 0.975)
  )
```

```
## # A tibble: 4 x 4
## # Rowwise:
##   income_factor boot_means     lower95 upper95
##   <fct>         <list>           <dbl>   <dbl>
## 1 <15k          <dbl [10,000]>   13.7    14.2
## 2 15-25k        <dbl [10,000]>   10.1    10.5
## 3 25-50k        <dbl [10,000]>    7.91    8.12
## 4 50-75k        <dbl [10,000]>    7.06    7.29
```

```r
library(tidyr)

plot_df <- boot_income_means %>%
  unnest_longer(boot_means) %>%
  mutate(income_factor = factor(income_factor))

ggplot(plot_df, aes(x = boot_means, fill = income_factor)) +
  geom_density(alpha = 0.4) +
  labs(
    title = "Bootstrap Distributions of Mean Asthma Prevalence by Income",
    x = "Bootstrapped Mean Asthma Prevalence",
    y = "Density",
    fill = "Income Group"
  ) +
  theme_minimal()
```



The bootstrap distributions reveal a strong income gradient in asthma prevalence. The lowest income group consistently exhibits substantially higher mean asthma prevalence, with minimal overlap with higher income groups. This suggests that income-related disparities are stable and not driven by sampling variability