Final: National

Ricky Truong

August 2025

Background

Delete everything in environment

{r, message = FALSE, warning = FALSE} # rm(list = ls())

Load libraries

```
library(CausalArima)
library(lubridate)
library(patchwork)
library(readr)
library(readxl)
library(tidyverse)
```

Load data for U.S. transportation-sector CO2 emissions[1], U.S. real GDP[2], U.S. population[3], U.S. annual average temperature[4], real global oil price[5], U.S. urban population[6], and U.S. trade/GDP ratio[7]

```
co2 <- read_excel("co2_sector.xlsx",
                  sheet = "Transportation")
gdp <- read.csv("GDPCA.csv")
population <- read.csv("POPTOTUSA647NWDB.csv")
temperature <- read_excel("statistic_id500472_average-annual-temperature-in-the-united-states-
                          sheet = "Data")
oil <- read.csv("oil-prices-inflation-adjusted.csv")
```

[1] https://www.eia.gov/environment/emissions/state/
[2] https://fred.stlouisfed.org/series/GDPCA
[3] https://fred.stlouisfed.org/series/POPTOTUSA647NWDB
[4] https://www-statista-com.ezp-prod1.hul.harvard.edu/statistics/500472/annual-average-temperature-in-the-us/
[5] https://ourworldindata.org/grapher/oil-prices-inflation-adjusted
[6] https://data.worldbank.org/indicator/SP.URB.TOTL.IN.ZS?locations=US
[7] https://data.worldbank.org/indicator/NE.TRD.GNFS.ZS?locations=US

```r
urbanization <- read.csv("API_SP.URB.TOTL.IN.ZS_DS2_en_csv_v2_22447.csv",
                         skip = 3)
trade <- read.csv("API_NE.TRD.GNFS.ZS_DS2_en_csv_v2_122459.csv",
                  skip = 3)
```

Data wrangling

Wrangle co2

```r
# Rename variables using second row
names(co2) <- as.character(unlist(co2[2,]))

# Delete unnecessary row
co2 <- co2[-c(1),]

# Select for states and years
co2 <- co2 %>%
  select(`1960`:`2023`, State)

# Convert data to tidy format
co2 <- co2 %>%
  pivot_longer(cols = -State, names_to = "Year", values_to = "CO2") %>%
  mutate(Year = as.integer(Year),
         CO2 = as.numeric(CO2))

# Filter for total
co2 <- co2 %>%
  filter(State == "US") %>%
  select(-State)
```

Wrangle gdp

```r
# Convert observation_date to Date type and extract year as new variable
gdp <- gdp %>%
  mutate(observation_date = as.Date(observation_date),
         Year = year(observation_date))

# Rename GDP variable
gdp <- gdp %>%
  rename(GDP = GDPCA)

# Ensure variables are consistent types
gdp <- gdp %>%
```

```r
  mutate(Year = as.integer(Year),
         GDP = as.numeric(GDP))

# Select for relevant variables
gdp <- gdp %>%
  select(Year, GDP)
```

Wrangle `population`

```r
# Convert observation_date to Date type and extract year as new variable
population <- population %>%
  mutate(observation_date = as.Date(observation_date),
         Year = year(observation_date))

# Rename population variable
population <- population %>%
  rename(Population = POPTOTUSA647NWDB)

# Ensure variables are consistent types
population <- population %>%
  mutate(Year = as.integer(Year),
         Population = as.numeric(Population))

# Select for relevant variables
population <- population %>%
  select(Year, Population)
```

Wrangle `temperature`

```r
# Delete unnecessary rows
temperature <- temperature[-c(1, 2),]

# Rename variables
variables <- c("Year", "Temperature")
names(temperature) <- variables

# Ensure variables are consistent types
temperature <- temperature %>%
  mutate(Year = as.integer(Year),
         Temperature = as.numeric(Temperature))
```

Wrangle `oil`

```
# Rename price variable
oil <- oil %>%
  rename(Oil = Oil.price...Crude.prices.since.1861..constant.2024.US..)

# Select for relevant variables
oil <- oil %>%
  select(Year, Oil)
```

Wrangle `urbanization`

```
# Rename variables
variables <- c("Country", "v2", "v3", "v4", 1960:2025)
names(urbanization) <- variables

# Select for relevant variables
years <- as.character(1960:2025)
urbanization <- urbanization %>%
  select(Country, any_of(years))

# Convert data to tidy format
urbanization <- urbanization %>%
  pivot_longer(cols = -Country, names_to = "Year", values_to = "Urbanization") %>%
  mutate(Year = as.integer(Year),
         Urbanization = as.numeric(Urbanization))

# Filter for U.S.
urbanization <- urbanization %>%
  filter(Country == "United States") %>%
  drop_na(Urbanization) %>%
  select(Year, Urbanization)
```

Wrangle `trade`

```
# Rename variables
variables <- c("Country", "v2", "v3", "v4", 1960:2025)
names(trade) <- variables

# Select for relevant variables
years <- as.character(1960:2025)
trade <- trade %>%
  select(Country, any_of(years))
```

```r
# Convert data to tidy format
trade <- trade %>%
  pivot_longer(cols = -Country, names_to = "Year", values_to = "Trade") %>%
  mutate(Year = as.integer(Year),
         Trade = as.numeric(Trade))

# Filter for U.S.
trade <- trade %>%
  filter(Country == "United States") %>%
  drop_na(Trade) %>%
  select(Year, Trade)
```

Create `national` by combining data sets

```r
# Combine via inner_join()
national <- inner_join(gdp, co2, by = "Year") %>%
  inner_join(population, by = "Year") %>%
  inner_join(temperature, by = "Year") %>%
  inner_join(oil, by = "Year") %>%
  inner_join(urbanization, by = "Year") %>%
  inner_join(trade, by = "Year") %>%
  select(Year, CO2, GDP, Population, Temperature, Oil, Urbanization, Trade)

# Filter for years less than or equal to 2010 (to avoid confounding)
national <- national %>%
  arrange(Year) %>%
  filter(Year <= 2010)

# Create LogGDP variable
national <- national %>%
  mutate(LogGDP = log(GDP)) %>%
  select(Year, CO2, GDP, LogGDP, Population, Temperature, Oil, Urbanization, Trade)
```

Exporting

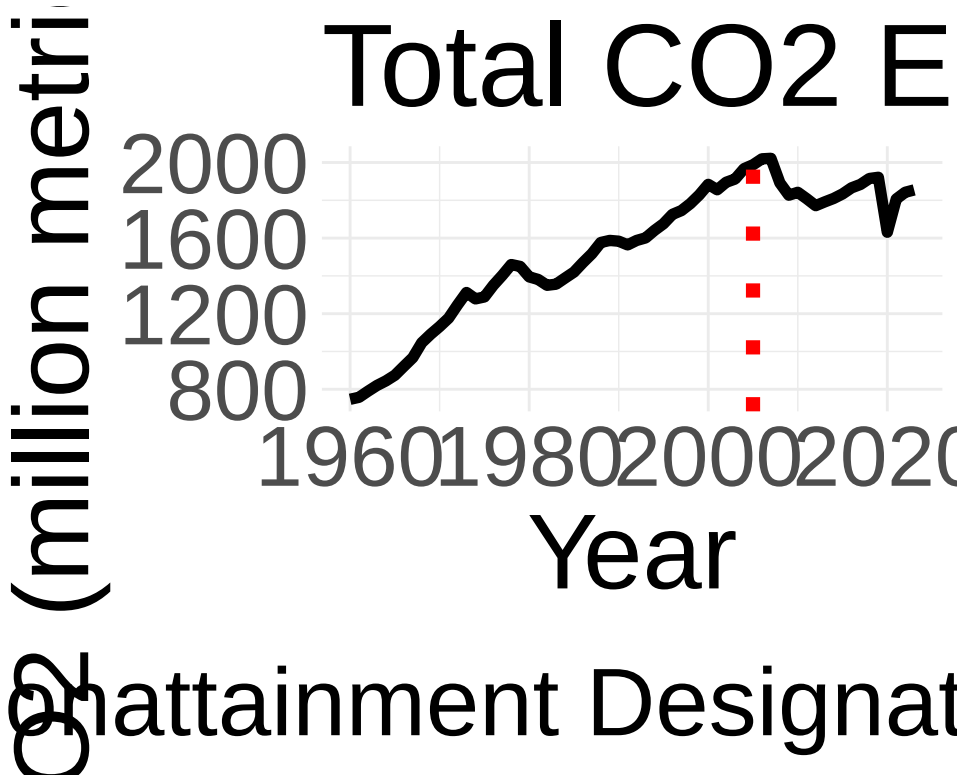
Export `national`

```r
write_csv(national, "~/national.csv")
```

Data visualization

Visualize CO2 (with all available years from co2)

```
# Create graph of CO2 data
co2_plot <- co2 %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "Total CO2 Emissions in U.S. Transporation Sector",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save graph
co2_plot
```



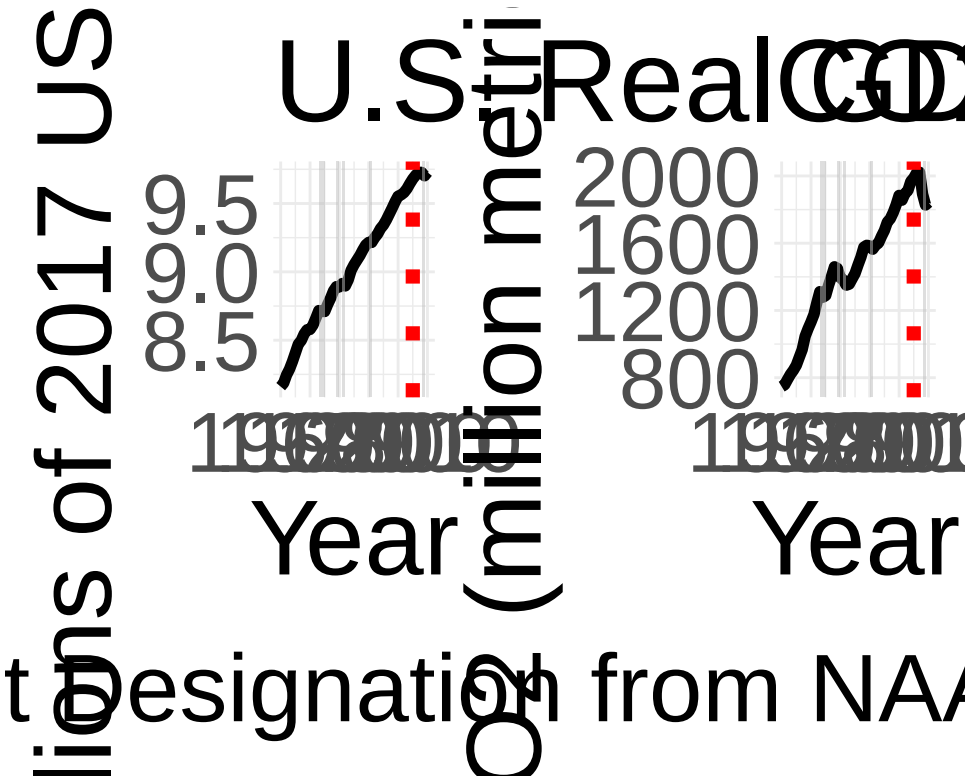

```
ggsave("co2_plot.png",
       plot = co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Visualize LogGDP and CO2

```
# Create data frame for years where LogGDP decreases year after
years_loggdp_decrease <- data.frame(Year = numeric())
for (n in 1960:2009) {
  if (national[national$Year == n + 1, "LogGDP"] <
      national[national$Year == n, "LogGDP"]) {
    years_loggdp_decrease <- rbind(years_loggdp_decrease, data.frame(Year = n))
  }
}

# Create data frame for these years to be shaded regions in graph
shaded_regions <- data.frame(xmin = years_loggdp_decrease$Year,
                             xmax = years_loggdp_decrease$Year + 1,
                             ymin = -Inf,
                             ymax = Inf)

# Create graph of LogGDP data with shaded regions
```

```r
loggdp_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = LogGDP), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Log(GDP) Decrease"),
            alpha = 0.5) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "U.S. Real GDP, Log Scale",
       x = "Year",
       y = "GDP (Billions of 2017 USD), Log Scale",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Log(GDP) Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Create graph of CO2 data with shaded regions
co2_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Log(GDP) Decrease"),
            alpha = 0.5) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Log(GDP) Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
```

```
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save side-by-side graphs
loggdp_co2_plot <- loggdp_plot + co2_plot +
  plot_layout(guides = "collect") &
  theme(legend.title = element_blank(), legend.position = "bottom")
loggdp_co2_plot
```



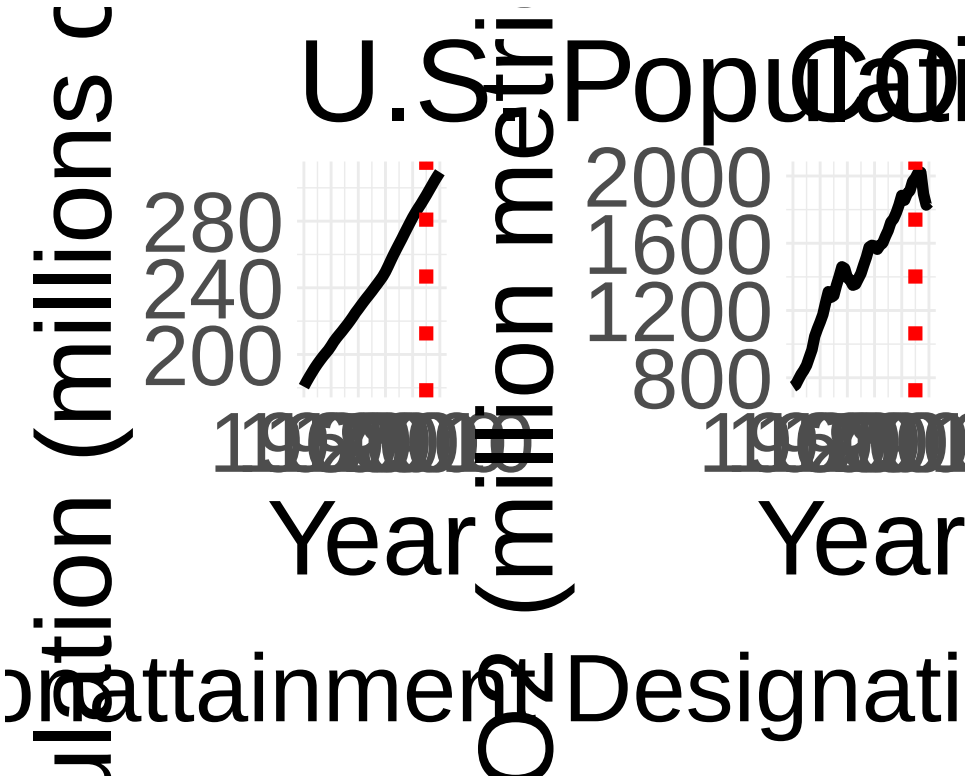

```
ggsave("loggdp_co2_plot.png",
       plot = loggdp_co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Visualize Population and CO2

```
# Create data frame for years where population decreases year after (empty)
years_population_decrease <- data.frame(Year = numeric())
for (n in 1960:2009) {
  if (national[national$Year == n + 1, "Population"] <
      national[national$Year == n, "Population"]) {
    years_population_decrease <- rbind(years_population_decrease, data.frame(Year = n))
```

```
  }
}

# Create graph of population data (with no shaded regions)
population_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = Population / 1e6), linewidth = 2) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "U.S. Population",
       x = "Year",
       y = "Population (millions of people)",
       fill = "",
       linetype = "") +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Create graph of CO2 data (with no shaded regions)
co2_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save side-by-side graphs
population_co2_plot <- population_plot + co2_plot +
  plot_layout(guides = "collect") &
  theme(legend.title = element_blank(), legend.position = "bottom")
```

```
population_co2_plot
```



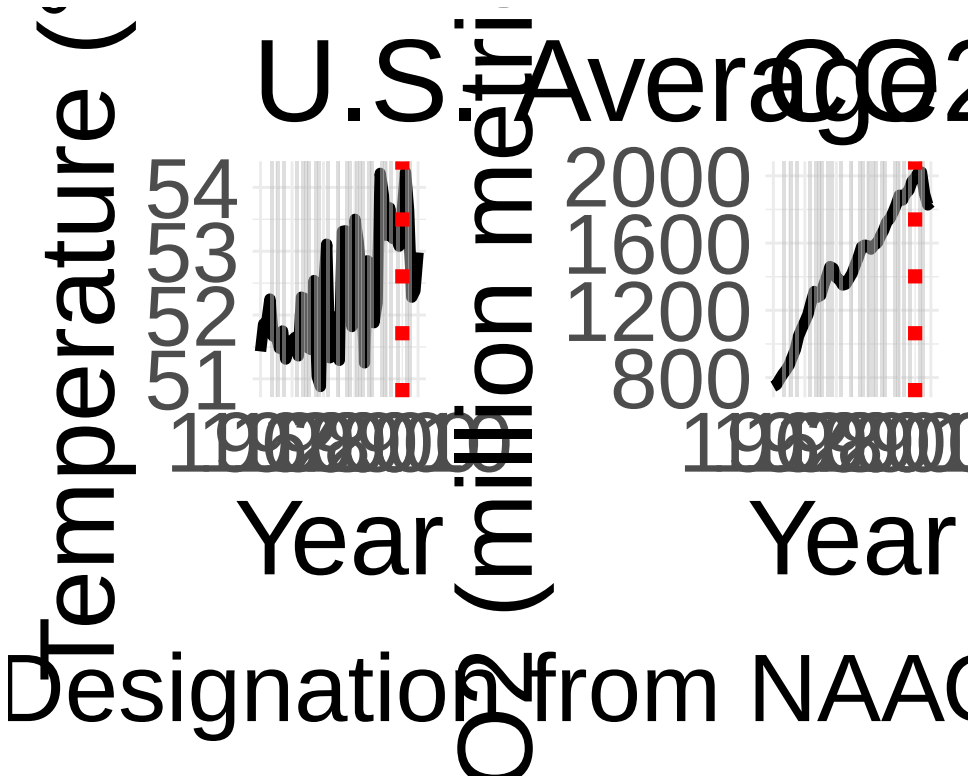

```
ggsave("population_co2_plot.png",
       plot = population_co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Visualize Temperature and CO2

```
# Create data frame for years where temperature decreases year after
years_temperature_decrease <- data.frame(Year = numeric())
for (n in 1960:2009) {
  if (national[national$Year == n + 1, "Temperature"] <
      national[national$Year == n, "Temperature"]) {
    years_temperature_decrease <- rbind(years_temperature_decrease, data.frame(Year = n))
  }
}

# Create data frame for these years to be shaded regions in graph
shaded_regions <- data.frame(xmin = years_temperature_decrease$Year,
                             xmax = years_temperature_decrease$Year + 1,
                             ymin = -Inf,
                             ymax = Inf)
```

```r
# Create graph of temperature data with shaded regions
temperature_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = Temperature), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Temperature Decrease"),
            alpha = 0.5) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "U.S. Average Annual Temperature",
       x = "Year",
       y = "Temperature (°F)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Temperature Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Create graph of CO2 data with shaded regions
co2_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Temperature Decrease"),
            alpha = 0.5) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Temperature Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
```

```
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save side-by-side graphs
temperature_co2_plot <- temperature_plot + co2_plot +
  plot_layout(guides = "collect") &
  theme(legend.title = element_blank(), legend.position = "bottom")
temperature_co2_plot
```



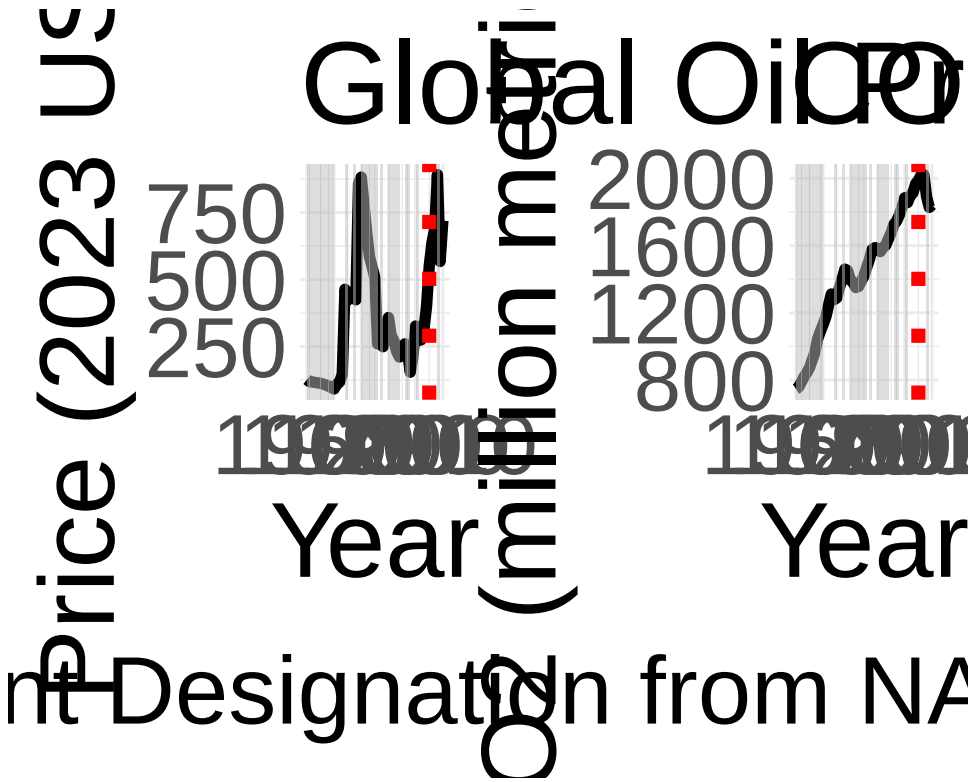

```
ggsave("temperature_co2_plot.png",
       plot = temperature_co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Visualize Oil and CO2

```
# Create data frame for years where oil decreases year after
years_oil_decrease <- data.frame(Year = numeric())
for (n in 1960:2009) {
  if (national[national$Year == n + 1, "Oil"] <
      national[national$Year == n, "Oil"]) {
```

```
    years_oil_decrease <- rbind(years_oil_decrease, data.frame(Year = n))
  }
}

# Create data frame for these years to be shaded regions in graph
shaded_regions <- data.frame(xmin = years_oil_decrease$Year,
                             xmax = years_oil_decrease$Year + 1,
                             ymin = -Inf,
                             ymax = Inf)

# Create graph of oil data with shaded regions
oil_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = Oil), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Oil Price Decrease"),
            alpha = 0.5) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "Global Oil Price per Cubic Meter",
       x = "Year",
       y = "Price (2023 USD)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Oil Price Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Create graph of CO2 data with shaded regions
co2_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Oil Price Decrease"),
            alpha = 0.5) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
```

```
            linewidth = 2.5) +
  labs(title = "CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Oil Price Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save side-by-side graphs
oil_co2_plot <- oil_plot + co2_plot +
  plot_layout(guides = "collect") &
  theme(legend.title = element_blank(), legend.position = "bottom")
oil_co2_plot
```

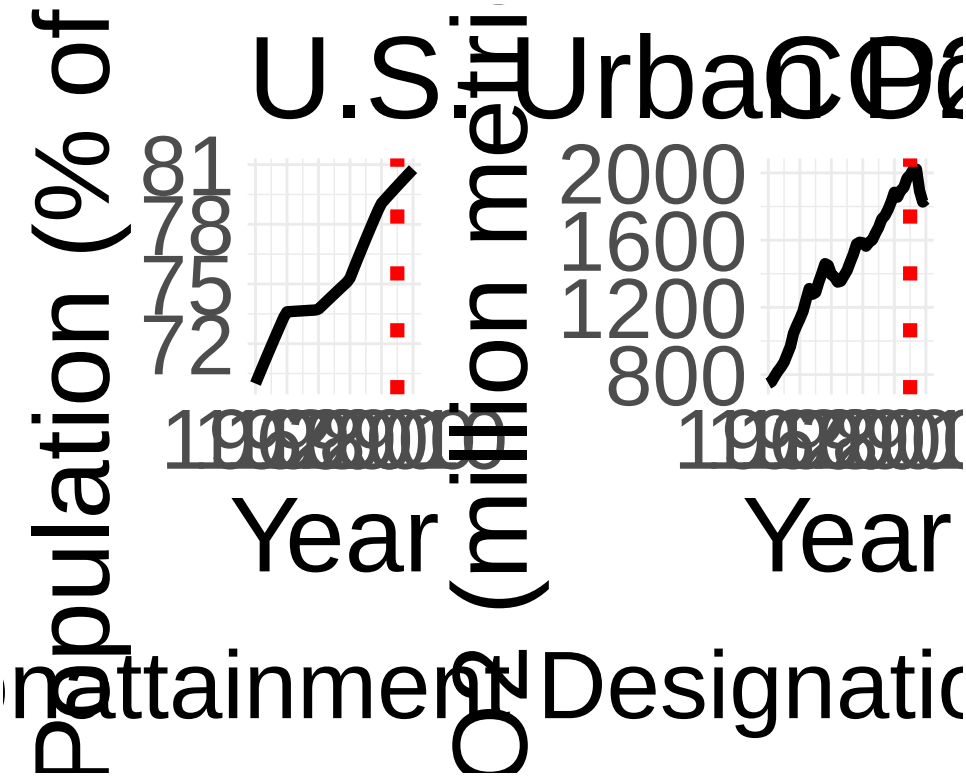



```
ggsave("oil_co2_plot.png",
       plot = oil_co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Visualize Urbanization and CO2

```
# Create data frame for years where urbanization decreases year after (empty)
years_urbanization_decrease <- data.frame(Year = numeric())
for (n in 1960:2009) {
  if (national[national$Year == n + 1, "Urbanization"] <
      national[national$Year == n, "Urbanization"]) {
    years_urbanization_decrease <- rbind(years_urbanization_decrease, data.frame(Year = n))
  }
}

# Create graph of urbanization data (with no shaded regions)
urbanization_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = Urbanization), linewidth = 2) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "U.S. Urban Population",
       x = "Year",
       y = "Urban Population (% of Population)",
       fill = "",
       linetype = "") +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Create graph of CO2 data (with no shaded regions)
co2_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
```

```
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save side-by-side graphs
urbanization_co2_plot <- urbanization_plot + co2_plot +
  plot_layout(guides = "collect") &
  theme(legend.title = element_blank(), legend.position = "bottom")
urbanization_co2_plot
```



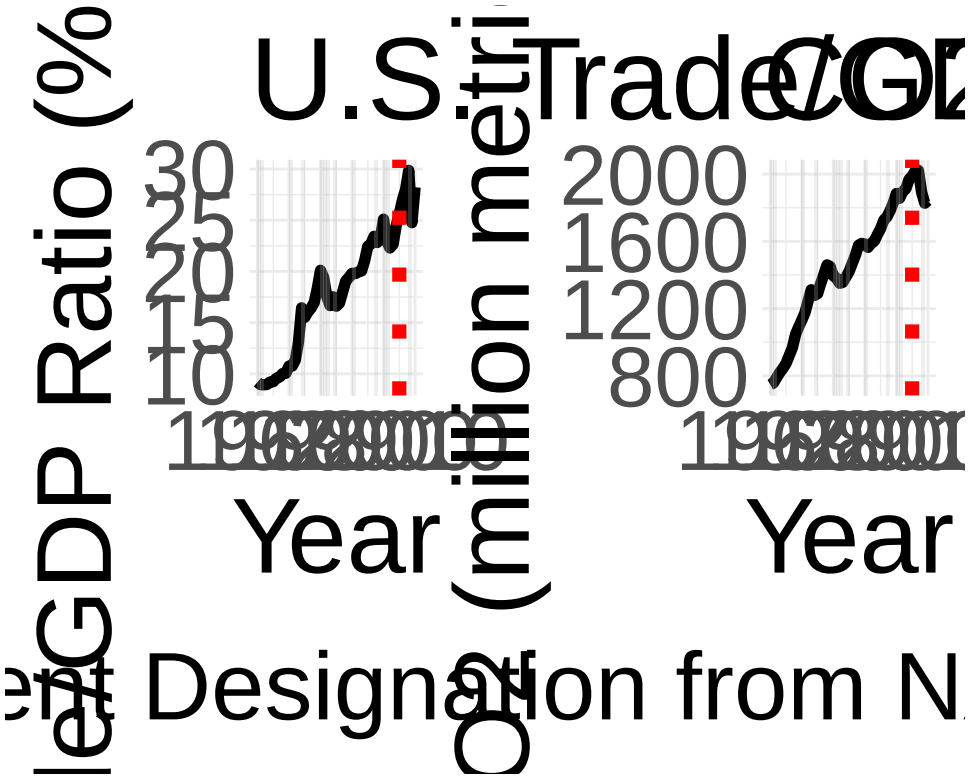

```
ggsave("urbanization_co2_plot.png",
       plot = urbanization_co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Visualize Trade and CO2

```
# Create data frame for years where trade decreases year after
years_trade_decrease <- data.frame(Year = numeric())
for (n in 1960:2009) {
  if (national[national$Year == n + 1, "Trade"] <
      national[national$Year == n, "Trade"]) {
    years_trade_decrease <- rbind(years_trade_decrease, data.frame(Year = n))
```

```
  }
}

# Create data frame for these years to be shaded regions in graph
shaded_regions <- data.frame(xmin = years_trade_decrease$Year,
                             xmax = years_trade_decrease$Year + 1,
                             ymin = -Inf,
                             ymax = Inf)

# Create graph of trade data with shaded regions
trade_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = Trade), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Trade Decrease"),
            alpha = 0.25) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
  labs(title = "U.S. Trade/GDP Ratio",
       x = "Year",
       y = "Trade/GDP Ratio (% of GDP)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Trade Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Create graph of CO2 data with shaded regions
co2_plot <- national %>%
  ggplot() +
  geom_line(aes(x = Year, y = CO2), linewidth = 2) +
  geom_rect(data = shaded_regions,
            aes(xmin = xmin, xmax = xmax,
                ymin = ymin, ymax = ymax,
                fill = "Trade Decrease"),
            alpha = 0.25) +
  geom_vline(aes(xintercept = 2005, linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red",
             linewidth = 2.5) +
```

```
  labs(title = "CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
       linetype = "") +
  scale_fill_manual(values = c("Trade Decrease" = "grey")) +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save side-by-side graphs
trade_co2_plot <- trade_plot + co2_plot +
  plot_layout(guides = "collect") &
  theme(legend.title = element_blank(), legend.position = "bottom")
trade_co2_plot
```




```
ggsave("trade_co2_plot.png",
       plot = trade_co2_plot,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```

Causal inference

Run CausalArima to estimate statistics

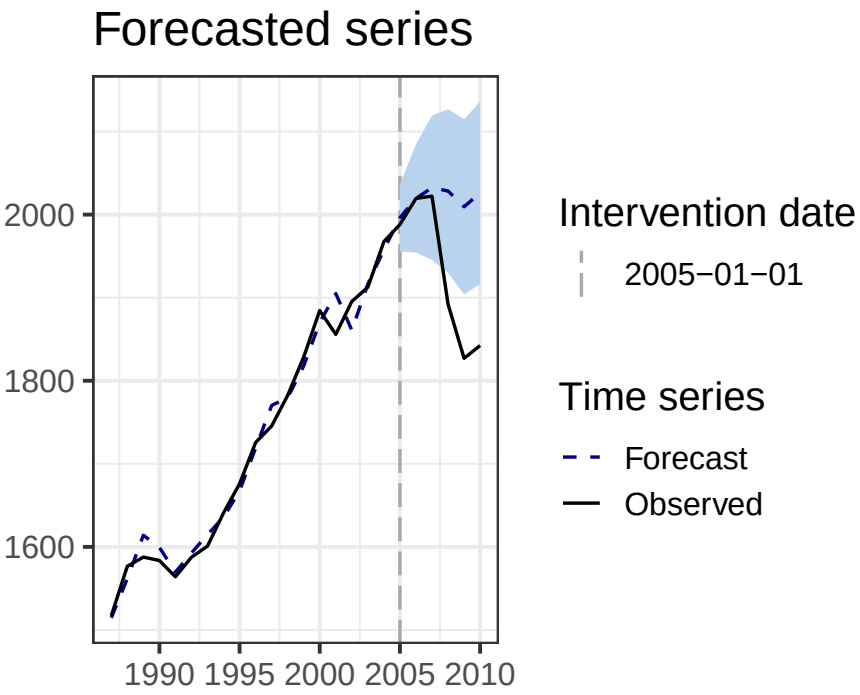
```
# Define intervention time point
intervention_date <- as.Date("2005-01-01")

# Create vector for dates
years <- paste0(seq(1960, 2010), "-01-01")
all_dates <- as.Date(years)

# Create time series for outcome with yearly seasonality
y_ts <- ts(national$CO2, frequency = 1)

# Create covariate matrix with yearly seasonality
cov_mat <- as.matrix(national %>%
                       select(LogGDP, Urbanization, Trade))

# Run CausalArima()
ce <- CausalArima(y = y_ts,
                  x = cov_mat,
                  dates = all_dates,
                  int.date = intervention_date,
                  nboot = 1000)

# Get impact as list
imp <- impact(ce)

# Extract via impact_norm cumulative effect and other statistics
norm_effect <- imp$impact_norm$sum
cumulative_effect <- norm_effect$estimate
sd_norm <- norm_effect$sd
ci_lower_norm <- cumulative_effect - 1.96 * sd_norm
ci_upper_norm <- cumulative_effect + 1.96 * sd_norm

# Calculate relative cumulative effect as a decimal
post_years <- 2005:2010
post_indices <- which(national$Year %in% post_years)
observed_post <- national$CO2[post_indices]
predicted_post <- ce$forecast
sum_obs <- sum(observed_post, na.rm = TRUE)
sum_pred <- sum(predicted_post, na.rm = TRUE)
rel_cumulative_effect <- (sum_obs - sum_pred) / sum_pred

# Extract via impact_boot cumulative effect and other statistics
boot_list <- imp$impact_boot
boot_effect <- boot_list$effect_cum[3, ]
```

```
# Store data as tibble
object <- tibble(cumulative_effect = cumulative_effect,
                 rel_cumulative_effect = rel_cumulative_effect,
                 sd_norm = sd_norm,
                 ci_lower_norm = ci_lower_norm,
                 ci_upper_norm = ci_upper_norm,
                 p_value_left_norm = norm_effect$p_value_left,
                 sd_boot = boot_effect$sd,
                 ci_lower_boot = boot_effect$inf,
                 ci_upper_boot = boot_effect$sup,
                 p_value_left_boot = as.numeric(boot_list$p_values["p"]))
object
```

```
## # A tibble: 1 x 10
##   cumulative_effect rel_cumulative_effect sd_norm ci_lower_norm ci_upper_norm
##               <dbl>                 <dbl>   <dbl>         <dbl>         <dbl>
## 1             -521.               -0.0430    225.         -961.         -80.2
## # i 5 more variables: p_value_left_norm <dbl>, sd_boot <dbl>,
## #   ci_lower_boot <dbl>, ci_upper_boot <dbl>, p_value_left_boot <dbl>
```

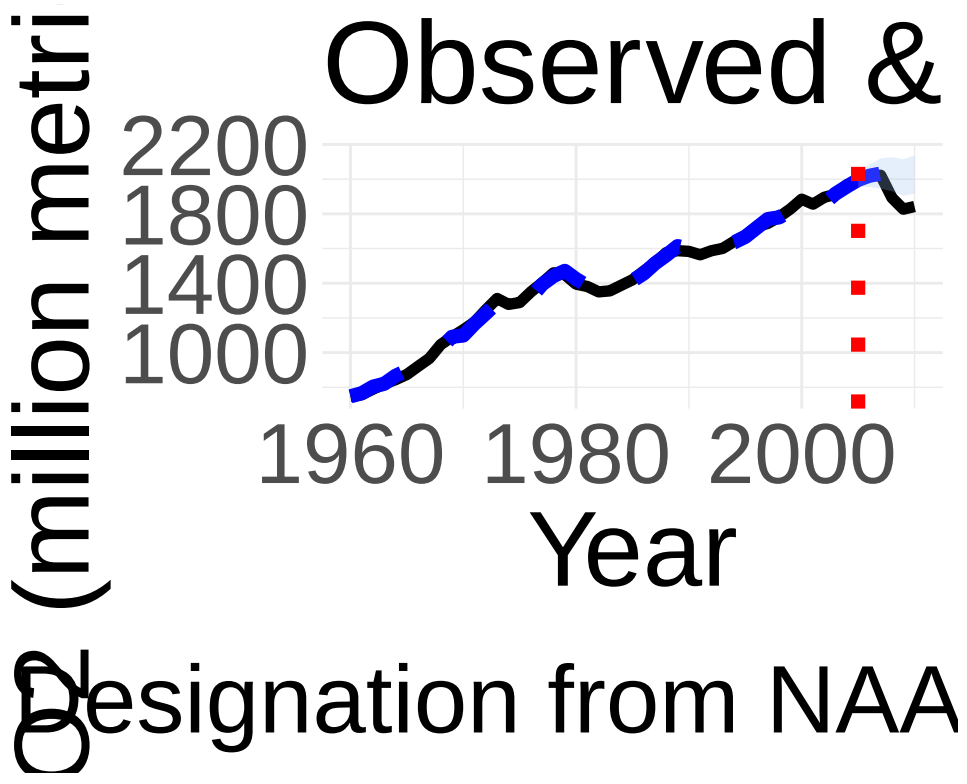
Visualize observed and counterfactual

```
# Plot counterfactual graph (default)
co2_plot_counterfactual_default <- plot(ce, type="forecast")
co2_plot_counterfactual_default
```



```r
# Extract data used in ribbon layer of default plot (i.e., CI)
ribbon_data <- layer_data(co2_plot_counterfactual_default, 2)

# Convert CI to data frame
ci_df <- ribbon_data %>%
  transmute(date = as.Date(x, origin = "1970-01-01"),
            lower = ymin,
            upper = ymax)

# Extract forecasted values from ARIMA model and combine with CI
forecast_df <- tibble(date = as.Date(paste0(2005:2010, "-01-01")),
                      CO2 = ce$forecast) %>%
  left_join(ci_df, by = "date")

# Extract fitted values from ARIMA model (before treatment)
fitted_df <- tibble(date = as.Date(paste0(national$Year[national$Year < 2005],
                                          "-01-01")),
                    CO2 = as.numeric(fitted(ce$model)),
                    lower = NA,
                    upper = NA)

# Combine everything into one counterfactual data frame
counterfactual_df <- bind_rows(fitted_df, forecast_df)

# Extract observed values into data frame
obs_df <- national %>%
  mutate(date = as.Date(paste0(Year, "-01-01")))

# Create counterfactual plot (with extended domain and range)
co2_plot_counterfactual_extended <- ggplot() +
  geom_line(data = obs_df,
            aes(x = date, y = CO2),
            color = "black", linewidth = 2) +
  geom_line(data = counterfactual_df,
            aes(x = date, y = CO2, linetype = "Counterfactual"),
            color = "blue", linewidth = 2.5) +
  geom_ribbon(data = counterfactual_df,
              aes(x = date, ymin = lower, ymax = upper, fill = "95% CI"),
              alpha = 0.25) +
  geom_vline(aes(xintercept = as.Date("2005-01-01"),
                 linetype = "2005: Nonattainment Designation from NAAQS"),
             color = "red", linewidth = 2.5) +
  coord_cartesian(xlim = as.Date(c("1960-01-01", "2010-01-01"))) +
  labs(title = "Observed & Counterfactual CO2 Emissions",
       x = "Year",
       y = "CO2 (million metric tons)",
       fill = "",
```

```
      linetype = "") +
  scale_linetype_manual(values = c("2005: Nonattainment Designation from NAAQS" = "dotted",
                                   "Counterfactual" = "dashed")) +
  scale_fill_manual(values = c("95% CI" = "#97C2F0")) +
  guides(linetype = guide_legend(order = 1),
         fill = guide_legend(order = 2)) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(size = 44),
        axis.text = element_text(size = 32),
        axis.title = element_text(size = 40),
        legend.text = element_text(size = 36))

# Plot and save counterfactual graph (with extended domain and range)
co2_plot_counterfactual_extended
```




```
ggsave("co2_plot_counterfactual_extended.png",
       plot = co2_plot_counterfactual_extended,
       width = 20, height = 12, dpi = 300,
       bg = "white")
```