project title*

project subtitle

a b Kevin You Ziheng Zhong

June 20, 2025

Table of contents

1 Introduction **3**

2 Method **4**

2.1 Data 4

2.2 Model 6

2.2.1 Model 1 6

2.2.2 Model 2 9

3 Result **14**

4 Discussion **14**

5 Appendix **14**

5.1 Project Code 14

References **15**

```
#| include: false
#| warning: false
#| message: false

# load library
library(tidyverse)
library(lubridate)
library(gridExtra)
```

*Code and data supporting this analysis is available at: Link to repository.

```
library(forecast)
library(ggplot2)
library(knitr)
library(readr)
library(dplyr)
library(astsa)
library(here)

# load data
co2 <- read.csv(here("data", "owid-co2-data.csv"))

# load model
```

1 Introduction

Climate change has become a critical global challenge, driven largely by rising carbon dioxide (CO_2) emissions from human activities. The Intergovernmental Panel on Climate Change (IPCC) warns that without immediate and deep emission reductions, the world is on track to exceed the Paris Agreement's 1.5°C warming threshold, underscoring an urgent need to curb CO_2 from all sectors (Intergovernmental Panel on Climate Change (IPCC) 2022). In this context, the United States – historically the single largest national source of CO_2 – plays a pivotal role. Cumulatively, the U.S. has emitted roughly 25% of all fossil-fuel CO_2 since the industrial era began (Ritchie 2019), and it remains one of the top annual emitters today. Understanding the trajectory of U.S. emissions, especially from major sectors like transportation, industry, and energy generation, is therefore of global importance for climate change mitigation.

This project focuses on analyzing the effect of U.S. transportation, industry, and energy generation on CO_2 emissions from 1750 up to 2023. Using historical emissions data from the Our World in Data's dataset (Ritchie et al. 2025), we examine how these three key sectors have contributed to national CO_2 output over time. The study not only reviews historical trends but also develops forecasts for future emissions, providing both a retrospective and forward-looking perspective on U.S. CO_2 emissions.

The motivation for this research is grounded in climate and energy policy relevance. The chosen sectors – transportation, industrial production, and power generation – are the dominant sources of U.S. greenhouse gas emissions (Keerthana et al. 2023). For example, in 2019 the transportation sector accounted for about 29% of U.S. CO_2-equivalent emissions, followed by electricity generation (25%) and industrial processes (23%) (Keerthana et al. 2023). These activities collectively drive the bulk of national emissions, meaning any meaningful climate strategy must address each of them. Analyzing historical patterns in these sectors can reveal how past economic growth, technological changes, and policies (such as vehicle efficiency standards or power plant regulations) have impacted emissions. Moreover, forecasting future emissions is crucial for gauging progress toward sustainability goals. The U.S. has set ambitious targets under the Paris Agreement – pledging a 50–52% reduction in greenhouse gases by 2030 (from 2005 levels) and net-zero emissions by 2050 (Keerthana et al. 2023) – which heightens the real-world significance of this study. By projecting emissions trajectories, we can assess whether current trends align with these climate goals or if additional policy interventions may be required.

The paper is structured as follows: Section 2 TBD, Section 3 TBD, Section 4 TBD.

2 Method

2.1 Data

The dataset is from the U.S. Energy Information Administration's (EIA) Monthly Energy Review, which issues detailed energy statistics, including production, consumption, trade, and emissions metrics. The Monthly Energy Review is released on the last workday of each month. It contains preliminary data that are subject to revision in subsequent releases and historical data that may be updated when source publications are revised. For this project, data were extracted for the period ranging from January 1973 to February 2025, with a total of 626 records at a monthly frequency. The EIA collects these data through standardized survey forms completed by energy industry participants, following review protocols approved every three years to ensure consistency and quality across reporting entities.

```
#| label: tbl-summary
#| echo: false
#| warning: false
#| message: false

data <- read.csv("../Data/cleaned_data.csv")


data <- data %>%
  mutate(
    Date = as.Date(paste(Year, Month, "1", sep = "-"), format = "%Y-%m-%d")
  )

ggplot(data, aes(x = Date, y = Total_CO2_Emissions)) +
  geom_line(size = 1) +
  labs(
    title = "Total CO2 Emissions Over Time",
    x     = "Date",
    y     = "Total CO2 Emissions (million metric tons)"
  ) +
  theme_minimal()


data_zoom <- data %>%
  filter(Date >= as.Date("2015-01-01") & Date <= as.Date("2022-12-31"))


ggplot(data_zoom, aes(x = Date, y = Total_CO2_Emissions)) +
```

```
  geom_line(size = 1) +
  labs(
    title = "Total CO2 Emissions (2015-2022)",
    x     = "Date",
    y     = "Total CO2 Emissions (million metric tons)"
  ) +
  theme_minimal() +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")

acf(
  data$Total_CO2_Emissions,
  main = "ACF of Total CO2 Emissions",
  xlab = "Lag",
  ylab = "Autocorrelation"
)
```

The cleaned dataset is constructed by concatenating monthly U.S. data from different categories. It includes energy production, sectoral energy use, transportation fuel use, and CO_2 emissions. Emissions from other sources are aggregated by date, and the total emissions across all sectors are calculated. Additional statistics on petroleum consumption, fossil fuel and renewable energy output, and residential, commercial, and industrial energy consumption are merged after standardized Year and Month variables have been extracted. All datasets are joined in year and month.

The resulting dataset captures key indicators such as total CO_2 emissions (million metric tons), transportation petroleum consumption (million barrels per day), total fossil fuel and renewable energy output (quadrillion Btu), and sectoral energy consumption (trillion Btu). Units adhere to the conventions of the Monthly Energy Review, with energy metrics predominantly expressed in U.S. customary units—such as Btu for energy and barrels for petroleum volumes—and emissions in metric units.

From plot 1, we can say that total CO2 emissions have a clear upward trend from the early 1970s through 2005 and then a decay from 2005 to the present. The sharp dip around 2020 likely reflects the economic slowdown during the COVID-19 pandemic. During this period, widespread lockdowns resulted in a significant reduction in transportation, industrial activity, and fossil fuel consumption, which contributed to the observed decline in emissions. The ACF plot shows that the total CO2 emission is non-stationary since the lag-1 autocorrelation is significant and the ACF decays very slowly. This confirms the presence of a non-stationary mean in the series.

2.2 Model

First, run an ordinary regression of y_t on $z_{t1}, \cdots, z_{tr}$ (acting as if the errors are uncorrelated). Retain the residuals, $\hat{x}_t = y_t - \sum_{j=1}^{r} \beta_j z_{tj}$.

Second, Identify ARMA model(s) for the residuals $\hat{x}_t$.

Third, Run weighted least squares (or MLE) on the regression model with autocorrelated errors using the model specified in step (ii).

Fourth, Inspect the residuals $\hat{w}_t$ for whiteness, and adjust the model if necessary.

2.2.1 Model 1

```
data_train <- data[data$Year <= 2020, ]
data_test  <- data[data$Year > 2020, ]

model1 <- lm(
  Total_CO2_Emissions ~ .
    - Date - Year - Month,
  data = data_train
)

summary(model1)
```

```
plot(model1$fitted.values, resid(model1),
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Residuals vs Fitted")
abline(h = 0, col = "red")

plot(data_train$Date, resid(model1),
     type = "l",
     xlab = "Date",
     ylab = "Residuals",
     main = "Residuals Over Time")
abline(h = 0, col = "red")


hist(resid(model1),
     breaks = 30,
     main = "Histogram of Residuals",
```

```
     xlab = "Residuals")

# Residuals satisfied identical normal distribution
qqnorm(resid(model1))
qqline(resid(model1), col = "red")
```

```
acf(resid(model1), main = "ACF of Residuals")
pacf(resid(model1), main = "PACF of Residuals")

# ljung-box test shows that the residuals are auto-correlated
Box.test(resid(model1), lag = 20, type = "Ljung-Box")
```

```
# auto fitted arima
resid_ts <- ts(resid(model1))
resid_arima <- auto.arima(resid_ts)
summary(resid_arima)
```

```
forecast::auto.arima(resid(model1))
```

```
y_hat_lm <- fitted(model1)

resid_hat_arima <- fitted(resid_arima)
y_hat_combined <- y_hat_lm + resid_hat_arima

plot(data_train$Date, data_train$Total_CO2_Emissions, type = "l", col = "black", lwd = 2,
     ylab = "Total CO2 Emissions (million metric tons)",
     xlab = "Date",
     main = "Combined Model Fit",
     cex.lab = 1.2, cex.main = 1.4)

lines(data_train$Date, y_hat_combined, col = "red", lwd = 2)

legend("topright",
       legend = c("Observed", "LM + ARIMA Fit"),
       col = c("black", "red"),
       lty = 1,
       lwd = 2,
       cex = 1.1,
       bty = "n")
```

```r
y_true <- data_train$Total_CO2_Emissions
y_lm   <- fitted(model1)

plot(data_train$Date, y_true, type = "l", col = "black", lwd = 2,
     ylab = "Total CO2 Emissions (million metric tons)",
     xlab = "Date",
     main = "Linear Model Fit vs Actual",
     cex.lab = 1.2, cex.main = 1.4)

lines(data_train$Date, y_lm, col = "red", lwd = 2)

legend("topright",
       legend = c("Observed", "Linear Model Fit"),
       col = c("black", "red"),
       lty = 1,
       lwd = 2,
       cex = 1.1,
       bty = "n")
```

```r
rss_lm <- sum((y_true - y_lm)^2)
rss_combined <- sum((y_true - y_hat_combined)^2)

cat("RSS of Linear Model:", rss_lm, "\n")
cat("RSS of Linear + ARIMA Model:", rss_combined, "\n")

rmse_lm <- sqrt(mean((y_true - y_lm)^2))
rmse_combined <- sqrt(mean((y_true - y_hat_combined)^2))

cat("RMSE of Linear Model:", rmse_lm, "\n")
cat("RMSE of Linear + ARIMA Model:", rmse_combined, "\n")
```

```r
# Predict data adter 2020
pred_result <- predict(model1, newdata = data_test, interval = "prediction", level = 0.95)

y_test_true <- data_test$Total_CO2_Emissions
y_test_pred <- pred_result[, "fit"]
y_test_lower <- pred_result[, "lwr"]
y_test_upper <- pred_result[, "upr"]

plot(data_test$Date, y_test_true, type = "l", col = "black", lwd = 2,
     ylab = "Total CO2 Emissions (million metric tons)",
```

```
     xlab = "Date",
     main = "Out-of-Sample Forecast with 95% Prediction Interval",
     cex.lab = 1.2, cex.main = 1.4)

lines(data_test$Date, y_test_pred, col = "blue", lwd = 2)
lines(data_test$Date, y_test_lower, col = "blue", lty = 2, lwd = 1.5)
lines(data_test$Date, y_test_upper, col = "blue", lty = 2, lwd = 1.5)

legend("topright",
       legend = c("Observed", "Predicted", "95% Prediction Interval"),
       col = c("black", "blue", "blue"),
       lty = c(1, 1, 2),
       lwd = 2,
       cex = 1.1,
       bty = "n")
```

2.2.2 Model 2

```
import os
import random
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from tensorflow.keras.callbacks import EarlyStopping




df = pd.read_csv("../Data/cleaned_data.csv")
```

```
df['Date'] = pd.to_datetime(df['Year'].astype(str) + '-' + df['Month'].astype(str) + '-01')
df = df.sort_values('Date')
```

```
features = [
    'Transportarion Petroleum Consumption',
    'Total Fossil Fuels Production',
    'Total Renewable Energy Production',
    'Commercial_Consumption',
    'Industrial_Consumption',
    'Residential_Consumption',
    'Month'
]
target = 'Total_CO2_Emissions'
data_all = df[features + [target, 'Date']].dropna()
```

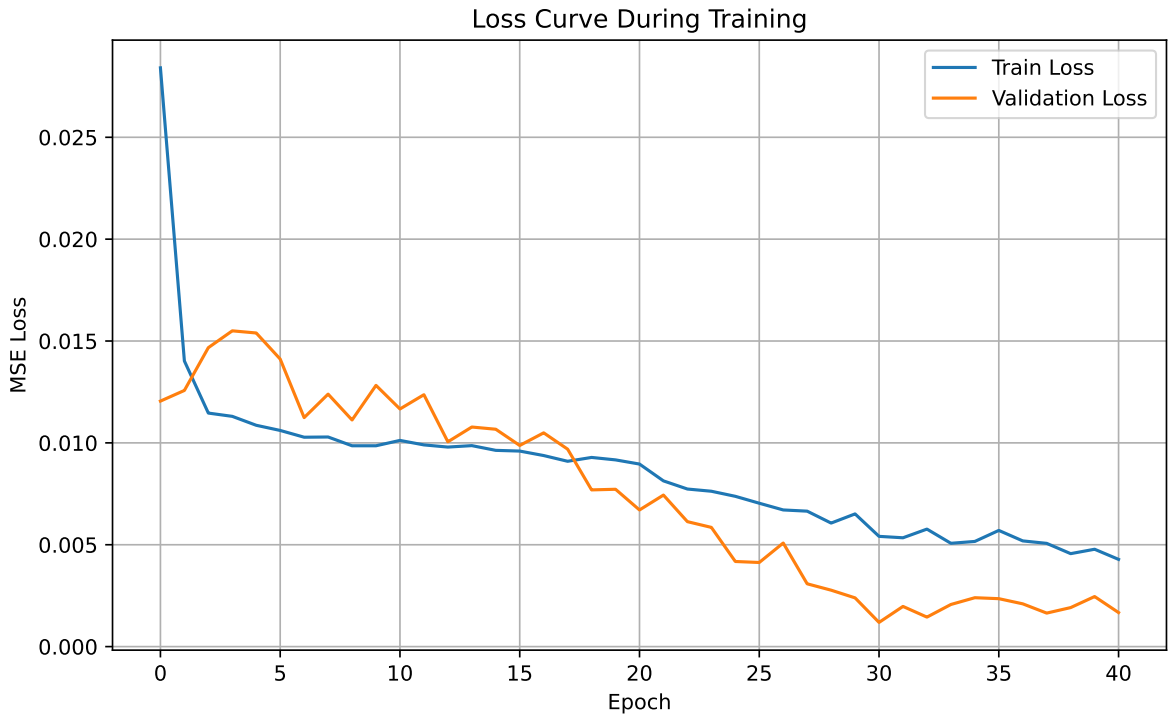
```
# normalize the data
feature_scaler = MinMaxScaler()
target_scaler = MinMaxScaler()

data_scaled = data_all.copy()
data_scaled[features] = feature_scaler.fit_transform(data_all[features])
data_scaled[[target]] = target_scaler.fit_transform(data_all[[target]])

# Construct sliding window samples
def make_sequences(data, seq_len=12):
    X, y, dates = [], [], []
    for i in range(len(data) - seq_len):
        X.append(data.iloc[i:i+seq_len][features].values)
        y.append(data.iloc[i+seq_len][target])
        dates.append(data.iloc[i+seq_len]['Date'])
    return np.array(X), np.array(y), np.array(dates)

seq_len = 24
X_all, y_all, dates_all = make_sequences(data_scaled, seq_len)

# train validation test set split
train_end = pd.Timestamp("2020-12-31")
valid_end = pd.Timestamp("2021-12-31")

train_idx = dates_all <= train_end
valid_idx = (dates_all > train_end) & (dates_all <= valid_end)
test_idx = dates_all > valid_end

X_train, y_train = X_all[train_idx], y_all[train_idx]
```

```
X_valid, y_valid = X_all[valid_idx], y_all[valid_idx]
X_test, y_test = X_all[test_idx], y_all[test_idx]
```

```
def set_seed(seed=42):
    os.environ['PYTHONHASHSEED'] = str(seed)
    random.seed(seed)
    np.random.seed(seed)
    tf.random.set_seed(seed)
    tf.config.experimental.enable_op_determinism()

set_seed(42)

# Construct the model
model = Sequential([
    LSTM(128, return_sequences=True, input_shape=(X_train.shape[1], X_train.shape[2])),
    Dropout(0.3),
    LSTM(64),
    Dense(1)
])
model.compile(optimizer='adam', loss='mse')

early_stop = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights=True)

history = model.fit(
    X_train, y_train,
    validation_data=(X_valid, y_valid),
    epochs=100,
    batch_size=16,
    callbacks=[early_stop],
    verbose=0
)
```

```
# loss function
plt.figure(figsize=(8, 5))
plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Validation Loss')
plt.title("Loss Curve During Training")
plt.xlabel("Epoch")
plt.ylabel("MSE Loss")
plt.legend()
plt.grid(True)
plt.tight_layout()
```

```
plt.show()
```

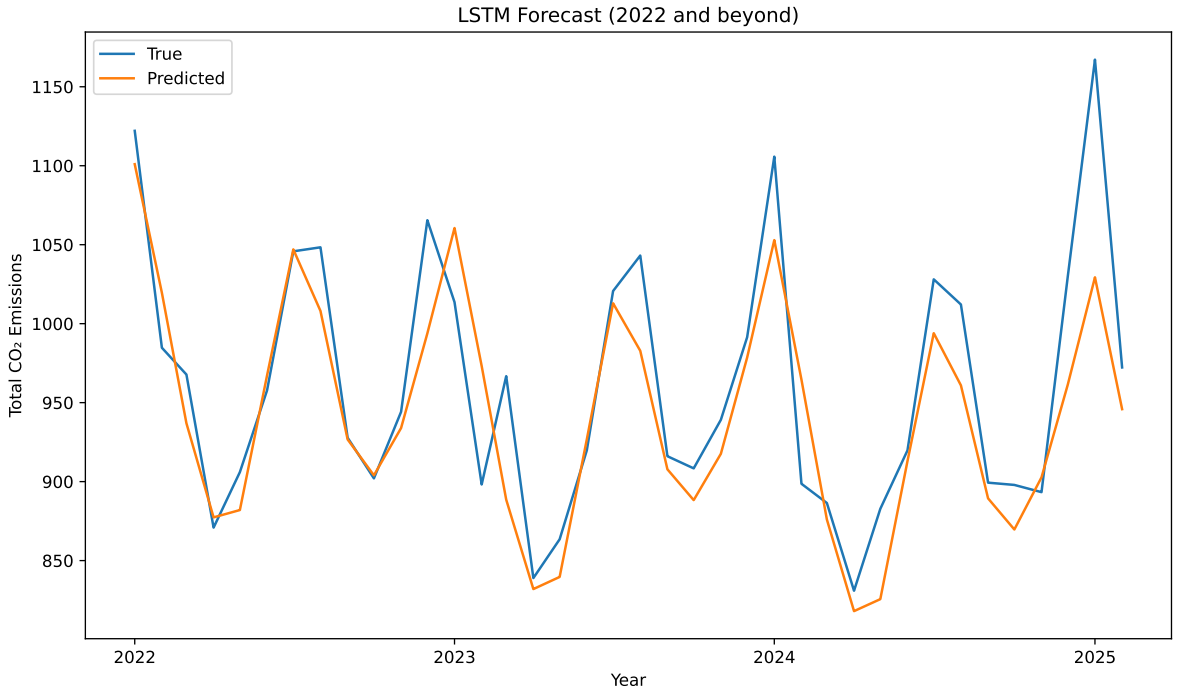



```
def predict_and_inverse(X, y, scaler):
    y_pred = model.predict(X, verbose=0)
    y_pred_inv = scaler.inverse_transform(y_pred.reshape(-1, 1))
    y_true_inv = scaler.inverse_transform(y.reshape(-1, 1))
    return y_true_inv, y_pred_inv

y_train_inv, y_pred_train_inv = predict_and_inverse(X_train, y_train, target_scaler)
y_valid_inv, y_pred_valid_inv = predict_and_inverse(X_valid, y_valid, target_scaler)
y_test_inv,  y_pred_test_inv  = predict_and_inverse(X_test,  y_test,  target_scaler)

# MSE evaluation for both training validation and prediction set
def print_rmse(label, y_true, y_pred):
    mse = mean_squared_error(y_true, y_pred)
    rmse = np.sqrt(mse)
    print(f"{label} MSE: {mse:.2f}, RMSE: {rmse:.2f}")
    return rmse

train_rmse = print_rmse("Train", y_train_inv, y_pred_train_inv)
```

```
valid_rmse = print_rmse("Valid", y_valid_inv, y_pred_valid_inv)
test_rmse  = print_rmse("Test",  y_test_inv,  y_pred_test_inv)
```

```
Train MSE: 2329.53, RMSE: 48.27
Valid MSE: 469.45, RMSE: 21.67
Test MSE: 1828.54, RMSE: 42.76
```

```
# Prediction on Test dataset
plt.figure(figsize=(10, 6))
plt.plot(dates_all[test_idx], y_test_inv, label='True')
plt.plot(dates_all[test_idx], y_pred_test_inv, label='Predicted')
plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y'))
plt.gca().xaxis.set_major_locator(mdates.YearLocator())
plt.title('LSTM Forecast (2022 and beyond)')
plt.xlabel("Year")
plt.ylabel("Total CO  Emissions")
plt.legend()
plt.tight_layout()
plt.show()
```



3 Result

4 Discussion

5 Appendix

5.1 Project Code

References

Intergovernmental Panel on Climate Change (IPCC). 2022. "Climate Change 2022: Mitigation of Climate Change."

Keerthana, Krishnamurthy Baskar, Shih-Wei Wu, Mu-En Wu, and Thangavelu Kokulnathan. 2023. "The United States Energy Consumption and Carbon Dioxide Emissions: A Comprehensive Forecast Using a Regression Model." *Sustainability* 15 (10): 7932. https://doi.org/10.3390/su15107932.

Ritchie, Hannah. 2019. "Who Has Contributed Most to Global CO2 Emissions?" *Our World in Data*.

Ritchie, Hannah, Max Roser, Edouard Mathieu, Bobbie Macdonald, and Pablo Rosado. 2025. "Our World in Data CO_2 and greenhouse gas emissions dataset." https://github.com/owid/co2-data; Global Change Data Lab / University of Oxford.