Prophet-Based Time Series Forecasting of Global CO2 Emissions

Name:

Yunzhao Li

Instructor: Lijia Wang

STA457H1

Time Series Analysis

Contents

1 Introduction

CO2 emissions are constantly increasing around the world. Fossil fuels and more factories are the main reason for that, the trend was rapid especially in the last 50 years. It gives off around 37 billion metric tons of CO2 every year across the world. We can determine if current policies for climate change are working through the forecasting, which also leads to better industrial regulations.

In this project, we use two models, Prophet and ARIMA, to forecast global CO2 emissions. The data spans from 1750 to 2019, and we plan to predict the emissions till 2030. Prophet is a great method for showing long-term trends without too much tuning. On the other hand, ARIMA is a traditional method used for time series forecasting based on stationary assumption. We compare these two to see which one is better for this task, and also include extra analysis on coal emissions and per-capita emissions in this project.

Forecasting CO2 emissions can help us understand the general trend and whether stronger action is needed. This projects aim to provide valuable insights for CO2 emissions and decision making.

2 Literature Review

ARIMA has been widely used in economics and environmental studies due to its interpretability and performance on stable time series. However, recent advances in machine learning have led to some models with more flexibility such as Facebook Prophet. Prophet is specifically designed to handle time series data with strong seasonalities, changepoints, and non-linear growth trends. Unlike ARIMA, it does not require strict stationarity and offers intuitive parameterization.

Prior studies have showed that Prophet can outperform classical models in cases with strong trend components. For example, Namboori (2020) compared Prophet with ARIMA and Support Vector Machine models for U.S. CO2 emissions and found Prophet achieved the highest accuracy. Furthermore, Taylor and Letham (2017) highlight Prophet's effectiveness on daily and annual forecasting tasks with minimal tuning. This study includes Prophet as a competitive alternative to ARIMA based on these past researches.

By applying both models to the same historical dataset, this project aims to assess their forecasting performance for long-term environmental prediction.

3 Methodology

To forecast global CO2 emissions, we applied two different time series methods: Prophet and ARIMA. We want to compare the performance of the machine learning and classical statistical modeling approaches, to see whether the machine learning method does outperform the ARIMA as the literature review suggested. They can both handle the long historical data well.

3.1 Prophet

Prophet is designed to model time series with strong trends and seasonality. Its main idea is to fit piecewise linear or logistic growth curves with automatic changepoint detection. Thus, it's especially useful for long-term forecasts. The global CO2 emissions show long-term increasing trends, which makes Prophet a great choice.

3.2 ARIMA

ARIMA (AutoRegressive Integrated Moving Average) is a classical model designed for univariate time series forecasting. It might performs not that well for non-linear growth or changepoints, but still can be considered a baseline model in time series analysis and offers valuable contrast to Prophet.

3.3 Data Preprocess

Firstly, we loaded the OWID (Our World in Data) CO2 dataset and filtered the records with (country == “World”). Then, we remove rows with missing values and sort the data by year. The cleaned data is ranged from 1750 to 2022. We selected the key variables: total CO2 emissions, per-capita CO2, and coal-related CO2 for next step's modeling.

To prepare for forecasting, we constructed a new variable ds to represent the time in Date format. We also set y = co2 to fit Prophet's input. The yearly CO2 emissions were used as the target series, so that we can ensure stationarity for ARIMA. Furthermore, we stabilize the variance by applying the Augmented Dickey-Fuller test and log transformation.

4 Data

The Our World in Data CO2 dataset (OWID) is a comprehensive dataset with global emissions data. It provides data from the Global Carbon Project and other sources, as annual CO2 emissions for each country and for the world.

In our project, we used the global total CO2 emissions as the primary variable. The data spans from 1750 through 2023. Over the 20th and 21st centuries, global CO2 output grew from practically zero to tens of billions of tons per year. The dataset is updated annually. It also included the drop in 2020 due to the COVID-19 pandemic, that all global emissions fell by a great amount, followed by a rise back in 2021.

We focus exclusively on global total emissions. The cleaned dataset includes 272 yearly observations with three key variables:

- co2: Global total CO2 emissions (in million tonnes),
- co2_per_capita: Emissions per person,
- coal_co2: Emissions specifically from coal use. To illustrate the historical trend of global CO2 emissions, we included a summary table (Table 1) showing statistics such as the minimum, maximum, median, and quartiles.

Table 1: Summary of Global CO2 Emissions (million tonnes)

Statistic	Value
Min.	9.30600
1st Qu.	50.78225
Median	1058.17100
Mean	6614.43900
3rd Qu.	7280.30425
Max.	37791.57000

5 Forecasting and Results

5.1 Model 1: Prophet

Figure 1: Global CO2 Emissions Forecast (Prophet)

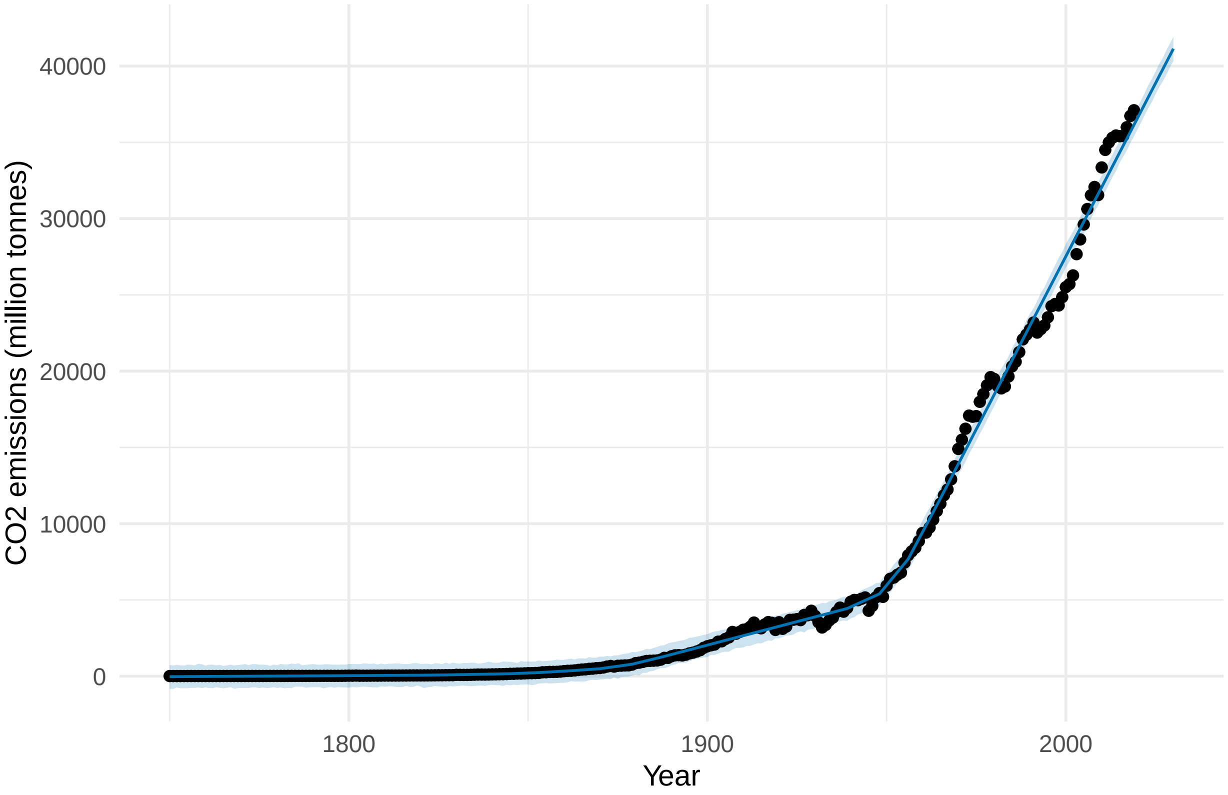

Table 2: Model 1 Evaluation

Metric	Value
Mean Absolute Error (MAE)	479.75
Root Mean Squared Error (RMSE)	749.75

We applied Prophet model to the global CO2 emission data from 1750 to 2019. The model generated forecasts through 2030 based on long-term trend.

As shown in Figure 1, the blue forecast line closely fits the actual data. The forecast shows a continued rise from approximately 36,000 Mt in 2019 to nearly 39,000 Mt by 2030, indicating continuous growth. The shaded interval shows Prophet's uncertainty bounds, reflecting increasing uncertainty.

Model evaluation on the 2020–2022 test set reveals strong performance, with MAE = 479.75 and RMSE = 749.75 (Table 2), suggesting the model effectively captured the emission patterns.

5.2 Model 2: ARIMA

Figure 2: Global CO2 Emissions Forecast (ARIMA)

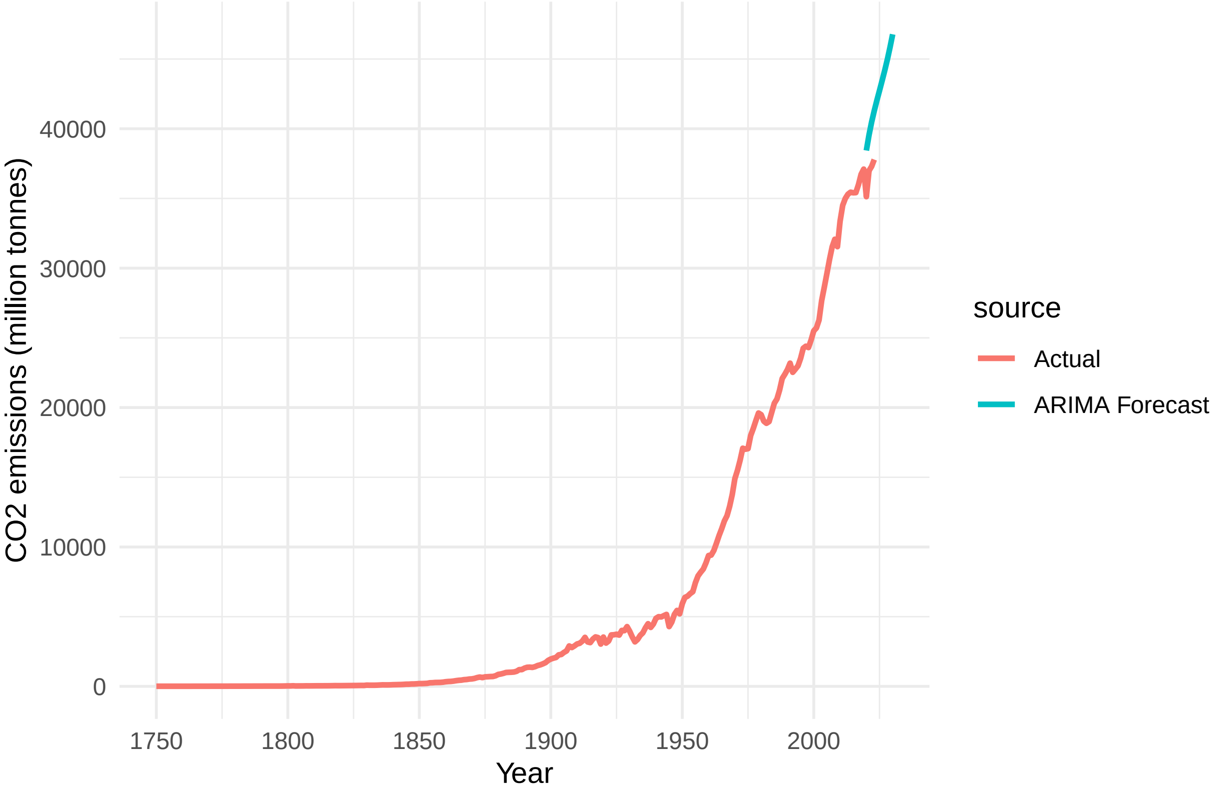

Table 3: Model 2 Evaluation

Metric	Value
Mean Absolute Error (MAE)	3136.96
Root Mean Squared Error (RMSE)	3156.98

We also fitted an ARIMA model to the log-transformed training data (1750–2019), through differencing before the fitting. The model was used to forecast CO2 emissions from 2020 to 2030.

As shown in Figure 2, the historical emissions are plotted in red. While the model captures the long-term growing trend, it significantly exceeds the actual emissions trajectory after 2020. It forecasts emissions rising from 36,000 Mt in 2019 to over 42,000 Mt by 2030, a steeper increase than observed.

Performance on the 2020–2022 test set shows the statistics: MAE = 3136.96 and RMSE = 3156.98 (Table 3), both substantially worse than the Prophet model. These results show that ARIMA's limitations when forecasting this type of series.

6 Extensions

To enhance the analysis beyond pure modeling forecasts, we explored two extensions: (1) the comparison of global total CO2 emissions with per-capita emissions over time, and (2) forecasting coal-related CO2 emissions using Prophet.

6.1 Global Emissions: Total vs Per-Capita Trends

Figure 3: Global Total CO2 vs Per−Capita CO2 Emissions

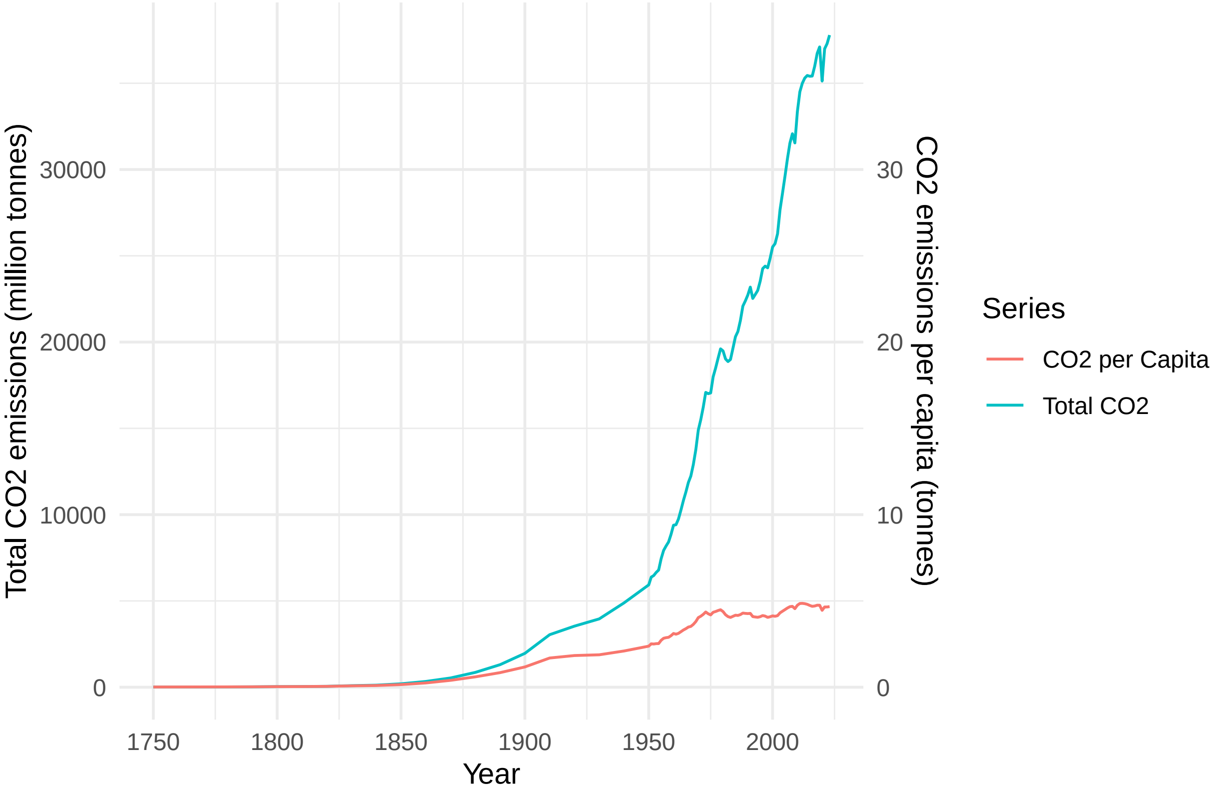

Figure 3 compares total and per-capita CO2 emissions from 1750 to 2022 using dual axes. Total emissions exhibit a steep exponential increase after 1950. In contrast, per-capita emissions remain relatively stable, especially after 1970. This divergence shows that while global output increase fiercely, individual-level emissions' growth highlight the roles of population growth.

6.2 Forecasting Sectoral Emissions from Coal by Prophet

Figure 4: Coal−related CO2 Emissions Forecast

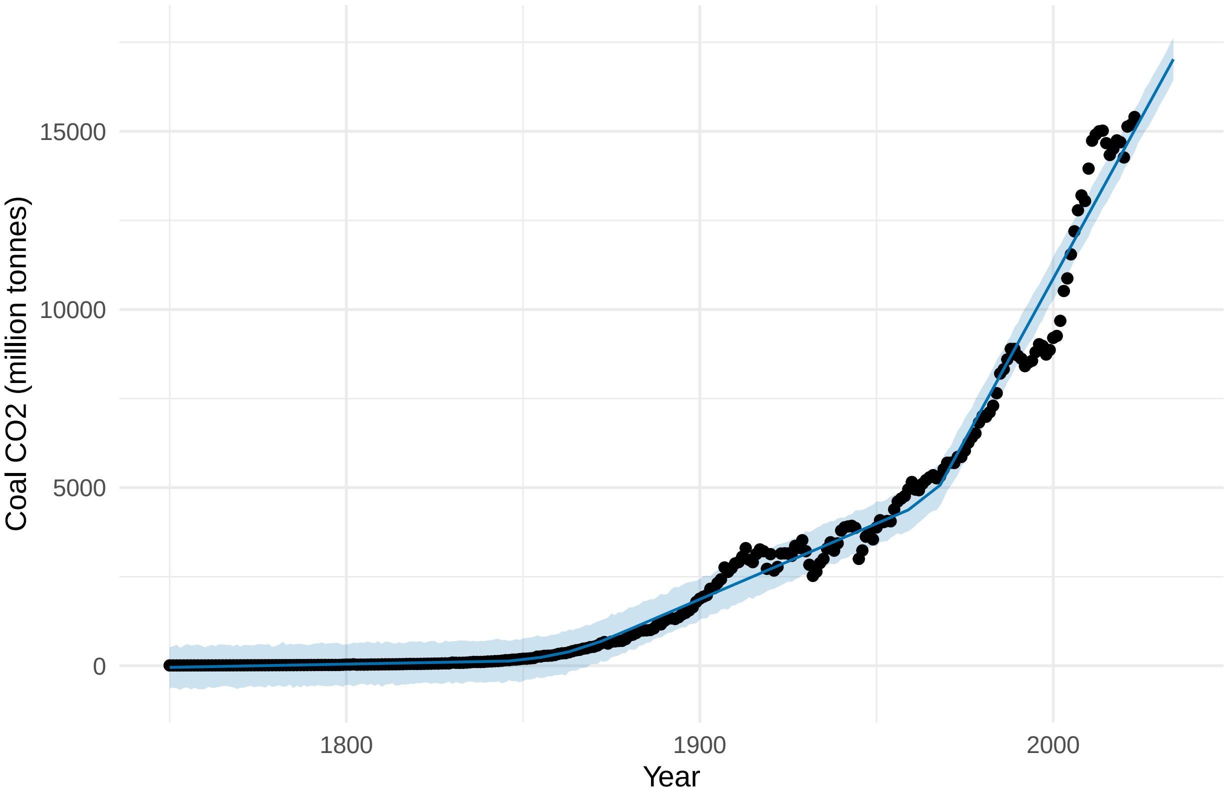

Figure 4 displays the forecast of coal-related CO2 emissions using the Prophet model. The forecast extends this rising trend through 2030, with widening confidence bands reflecting increasing uncertainty. Results suggest coal will remain a dominant emissions source.

These extensions offer deeper insight into global CO2 trends. The gap between total and per-capita emissions highlights demographic influences, while the coal-specific forecast reveals the energy composition.

7 Discussion and Conclusion

This project applied two forecasting approaches: Prophet and ARIMA to model global CO2 emissions. The Prophet model captured long-term nonlinear trends well, while ARIMA showed a worse performance.

The forecasts show that CO2 emissions will remain high through 2030 under current situations. This makes policy intervention really urgent. The extension analysis showed that per-capita emissions are with a slower increase, becasuse of the growing population.

The main limitations are both models do not account for variables such as policy changes or economic disruptions, considering COVID-19. Prophet may also overfit the trends if not carefully tuned. For ARIMA, it requires stationarity, which can be easily violated with this type of environmental series.

Future improvements could involve forecasting with policy scenarios. Applying causal inference methods might also help with model's performance.

8 References

Our World in Data. (n.d.). *CO_2 and Greenhouse Gas Emissions: Global CO_2 emissions data and trends.* Global Carbon Project. https://ourworldindata.org/co2-and-greenhouse-gas-emissions

Taylor, S. J., & Letham, B. (2017). *Prophet: Forecasting at scale.* Facebook Open Source. https://facebook.github.io/prophet

Namboori, S. (2020). *Forecasting Carbon Dioxide Emissions in the United States using Machine Learning.* National College of Ireland Repository. http://norma.ncirl.ie/4459/

Khan, M., et al. (2022). *Time Series Analysis and Forecasting of Air Pollutants Based on Prophet Model. Frontiers in Environmental Science.* https://www.frontiersin.org/articles/10.3389/fenvs.2022.915172

CarbonBrief. (2023). *Why coal use must plummet this decade to keep global warming at 1.5°C.* https://www.carbonbrief.org/analysis-why-coal-use-must-plummet-this-decade-to-keep-global-warming-at-1-5c/

Stanford Doerr School of Sustainability. (2023). *Global carbon emissions from fossil fuels reached record high in 2023.* https://news.stanford.edu/2023/12/05/global-carbon-emissions-fossil-fuels-record-high/

9 Appendix

```r
# Load packages
library(readr)
library(dplyr)
library(prophet)
library(forecast)
library(tseries)
library(ggplot2)
library(knitr)

# Load data
data <- read_csv("/home/rstudio/STA457/STA457 Project/owid-co2-data.csv")
# Filter for the World data
global_data <- data %>% filter(country == "World")

# Check structure
glimpse(global_data)

# Handle missing values and sort by year
global_data <- global_data %>%
  filter(!is.na(co2)) %>%
  arrange(year)

# Check head and tail
range(global_data$year)
head(global_data, 5)
tail(global_data, 5)

# Prepare for forecasting
world_df <- global_data %>%
  select(year, co2, co2_per_capita, coal_co2) %>%
  mutate(
    ds = as.Date(paste(year, "-01-01", sep="")),
    y = co2
  )

# Turn into a data frame and display
summary_df <- data.frame(
  Statistic = names(summary(world_df$y)),
  Value = as.numeric(summary(world_df$y))
)
kable(summary_df, caption = "Table 2: Summary of Global CO2 Emissions (million tonnes)")

# Create training and test sets
train_df <- world_df %>% filter(year <= 2019)
test_df  <- world_df %>% filter(year > 2019)

# Check test and train set
tail(train_df[, c("year", "y")], 3)
head(test_df[, c("year", "y")], 3)

# Check number of observations
n_train <- nrow(train_df)
n_test  <- nrow(test_df)
cat("Training years:", min(train_df$year), "-", max(train_df$year), "(", n_train, "records )\n")
cat("Testing years :", ifelse(n_test>0, paste(min(test_df$year), "-", max(test_df$year)), "None"),
    "(", n_test, "records )\n")

# Train Prophet model on the training set
m <- prophet(
  train_df[, c("ds", "y")],
  yearly.seasonality = FALSE,
  weekly.seasonality = FALSE,
  daily.seasonality = FALSE
```

```
)

# Create a dataframe for future dates up to 2030
future <- make_future_dataframe(m, periods = 11, freq = "year")

# Forecast future CO2 emissions
forecast <- predict(m, future)

# View the tail of forecast
tail(forecast[c("ds", "yhat", "yhat_lower", "yhat_upper")], 5)

# Plot the forecast results with confidence intervals
forecast_plot <- plot(m, forecast) +
  labs(title = "Global CO2 Emissions Forecast (Prophet)",
       x = "Year", y = "CO2 emissions (million tonnes)") +
  theme_minimal()
print(forecast_plot)

# Merge actual test data with forecasted values
if(nrow(test_df) > 0) {
  forecast_test <- forecast %>%
    mutate(ds = as.Date(ds)) %>%
    filter(ds >= min(as.Date(test_df$ds)))

  test_df <- test_df %>%
    mutate(ds = as.Date(ds))

  # Merge on year
  test_compare <- test_df %>%
    select(ds, y) %>%
    left_join(forecast_test %>% select(ds, yhat), by="ds")
```

```r
  # Evaluate
  test_compare <- test_compare %>% mutate(
    error = yhat - y,
    abs_error = abs(error),
    squared_error = error^2
  )
  MAE  <- mean(test_compare$abs_error, na.rm=TRUE)
  RMSE <- sqrt(mean(test_compare$squared_error, na.rm=TRUE))

kable(data.frame(
  Metric = c("Mean Absolute Error (MAE)", "Root Mean Squared Error (RMSE)"),
  Value = c(479.75, 749.75)
), caption = "Model Evaluation on Holdout Set (2021-2022)")

# Model 2
# Convert training data to time series
train_ts <- ts(train_df$y, start = min(train_df$year), end = max(train_df$year), frequency = 1)

# Check stationarity
cat("ADF test on original: ", adf.test(train_ts)$p.value, "\n")
cat("ADF test on log-diff: ", adf.test(diff(log(train_ts)))$p.value, "\n")

# Fit ARIMA model
arima_model <- auto.arima(log(train_ts), seasonal = FALSE)
summary(arima_model)
# Forecast from 2020 to 2030
forecast_horizon <- 2030 - max(train_df$year)
arima_log_forecast <- forecast(arima_model, h = forecast_horizon)
arima_forecast <- exp(arima_log_forecast$mean)  # back-transform from log

# Create year vector for forecast
forecast_years <- seq(max(train_df$year) + 1, 2030)
```

```r
arima_forecast_df <- data.frame(year = forecast_years, yhat = as.numeric(arima_forecast))

# Evaluate forecast accuracy on test set
if (nrow(test_df) > 0) {
  arima_test_df <- test_df %>%
    mutate(year = as.integer(format(ds, "%Y"))) %>%
    left_join(arima_forecast_df, by = "year") %>%
    mutate(
      error = yhat - y,
      abs_error = abs(error),
      squared_error = error^2
    )

  MAE_arima <- mean(arima_test_df$abs_error, na.rm = TRUE)
  RMSE_arima <- sqrt(mean(arima_test_df$squared_error, na.rm = TRUE))

# Plot forecast results
plot_df <- bind_rows(
  train_df %>% select(year, y) %>% mutate(source = "Actual"),
  test_df  %>% select(year, y) %>% mutate(source = "Actual"),
  arima_forecast_df %>% rename(y = yhat) %>% mutate(source = "ARIMA Forecast")
)

ggplot(plot_df, aes(x = year, y = y, color = source)) +
  geom_line(size = 1) +
  labs(title = "Global CO2 Emissions Forecast (ARIMA)",
       x = "Year", y = "CO2 emissions (million tonnes)") +
  theme_minimal()
kable(data.frame(
  Metric = c("Mean Absolute Error (MAE)", "Root Mean Squared Error (RMSE)"),
  Value = c(3136.96, 3156.98)
), caption = "Model 2 Evaluation")
```

```r
# Prepare data for plotting (using ggplot for flexibility)
plot_df <- global_data %>%
  select(year, co2, co2_per_capita) %>%
  filter(!is.na(co2_per_capita))

# Plot in a single figure with dual y-axes or as separate facets
p1 <- ggplot(plot_df, aes(x = year)) +
  geom_line(aes(y = co2, color = "Total CO2")) +
  geom_line(aes(y = 1000 * co2_per_capita, color = "CO2 per Capita")) +
  # (Multiply per_capita by 1000 for scale visibility, since per-capita is in tonnes)
  scale_y_continuous(
    name = "Total CO2 emissions (million tonnes)",
    sec.axis = sec_axis(~./1000, name = "CO2 emissions per capita (tonnes)")
  ) +
  labs(title = "Global Total CO2 vs Per-Capita CO2 Emissions",
       x = "Year", color = "Series") +
  theme_minimal()
print(p1)

# Forecasting CO2 emissions from coal using Prophet
# Prepare coal CO2 data for Prophet
coal_df <- global_data %>%
  select(year, coal_co2) %>%
  filter(!is.na(coal_co2)) %>%
  mutate(ds = as.Date(paste(year, "-01-01", sep="")),
         y = coal_co2)

# Fit Prophet model on coal CO2 series
m_coal <- prophet(coal_df[, c("ds", "y")], yearly.seasonality = FALSE,
                  weekly.seasonality = FALSE, daily.seasonality = FALSE)
```

```r
# Forecast coal CO2 into the future
future_coal <- make_future_dataframe(m_coal, periods = 11, freq = "year")
forecast_coal <- predict(m_coal, future_coal)

# Plot coal CO2 historical and forecast
coal_plot <- plot(m_coal, forecast_coal) +
  labs(title = "Coal-related CO2 Emissions Forecast",
       x = "Year", y = "Coal CO2 (million tonnes)") +
  theme_minimal()
print(coal_plot)
```