Modelling Cocoa Futures Prices: A Statistical Analysis of Ghana's Cocoa Production

Authors: Amanda Ng, Ever Hughes, Nicholas William Susanto, Jenny Oh

Introduction

Cocoa beans are a popular agricultural product and one of the fastest growing commodities of 2024 (Thukral and Tan, 2024). Most cocoa production comes from West Africa, with Ghana being an important producer. Since cocoa prices are sensitive to local and global factors, various stakeholders are interested in predicting prices. From producers to chocolate consumers, cocoa prices are impactful at multiple levels. Prices have been skyrocketing since late 2024 in the midst of a global cocoa shortage (J.P. Morgan, 2024). Climate change and weather patterns are key factors behind cocoa supply and pricing. Using cocoa futures price data from the International Cocoa Organization (ICCO), we will use time series methods to predict cocoa prices based on local climate, global demand, and economic measures. Our main objective is to build an accurate time series model of cocoa prices using predictors of local weather, cocoa demand, and exchange rate. We will compare a more traditional ARIMA model with a random forest model to determine which is most accurate and informative. Our key challenges are determining which potential predictors are most important, disaggregating predictor values, and ensuring model accuracy.

Literature Review

Cocoa prices are considered highly volatile (Tothmihaly, 2018), meaning they change quickly and unpredictably in short periods. Prices have spiked in the last few years due to a global cocoa shortage in West Africa (Kongor et al., 2024). Commodity prices have been forecasted with various time series methods, from ARIMA models to deep learning. Sukiyono et al. (2018) forecasted cocoa prices using exponential smoothing, decomposition, and ARIMA, finding ARIMA to be the most accurate. In comparison, a model based on older data found a mixed ARIMA/GARCH model to perform the best (Assis et al., 2010). More complex machine learning techniques have also been applied to forecasting. In modelling virus outbreaks, Kane et al. (2014) found that a Random Forest model outperformed an ARIMA on prediction accuracy. Our approach builds on the existing literature by comparing an ARIMA with covariates and a Random Forest model. Past research has not made use of known influences on cocoa price, like climate, exchange rate, and demand. Key market factors increasing prices include high demand, climate change, and local production costs (Wiśniewska, 2024). Climate factors have been shown to be important determinants of cocoa prices (Asante, et al., 2022; Yoroba, et al., 2019; Tabe-Ojong, Guedegbe, & Glauber, 2024; Ritchie, 2024; J.P. Morgan, 2024). Climate changes and El Nĩno have been deemed responsible for the significant drop in West African cocoa and spike in cocoa prices since 2023. Specifically, local precipitation and temperature are significant factors affecting cocoa yield (Asante, et al., 2022). Prices are also related to demand, which has been increasing in countries like China (Interesse, 2024). USD exchange rates also

affect cocoa pricing (Olaigbe & Usman, 2025). Increasing exchange rate suggests an appreciation in the value of Ghana's currency, resulting in an increase in purchase price of raw cocoa and final cocoa prices. Changes in cocoa price might be lagged due to the producing country's price regulations. By incorporating these relevant external factors, our goal is to improve model accuracy. Given the current economic situation post-COVID-19, all past models are also outdated. Our model is built off data from during and after the pandemic, which introduced global economic issues.

Data

Data sources

This study utilizes four datasets (cocoa price, Ghana climate, cocoa consumption and exchange rate). The **Cocoa Price** dataset provides a comprehensive record of daily closing prices for cocoa futures contracts. These contracts are traded on major commodity exchanges and reflect market expectations of future cocoa prices. The International Cocoa Organization (ICCO) dataset spans an extensive period from March 10, 1994, to February 27, 2025. We chose to narrow the observation window to the last 10 years, from January 1, 2015 to February 27, 2025, covering a total of 3,711 days. We did this to improve model accuracy, as recent cocoa prices are very different from historical prices and trends. Cocoa prices from the 1990s and 2000s are no longer informative on today's data.

We also chose to aggregate the data to monthly instead of daily values. Since prices are highly volatile (Tothmihaly, 2018), monthly values should show a more stable trend over time and improve predictive accuracy.

The **Ghana Climate Data** consists of daily meteorological observations from Ghana, the world's leading cocoa producer. The dataset, sourced from the National Centers for Environmental Information (NCEI) spans a window from January 1, 1990, to November 28, 2024. It includes crucial climate variables that impact cocoa cultivation. Each entry contains information such as the observation station ID, station name, and the date of observation. Key weather indicators include daily precipitation, which measures rainfall and its impact on soil moisture, and temperature variables such as average daily temperature, maximum daily temperature, and minimum daily temperature. We will be using precipitation and average temperature (see plots below) in our ARIMA model.

The **Per Capita Consumption of Cocoa Beans** dataset, provided by the Food and Agriculture Organization of the United Nations (FAO, 2023), spanning a window from 1961 to 2021. This dataset contains information on the annual average cocoa bean consumption per person at the global and country levels. This data is essential for understanding trends in cocoa consumption and will offer valuable insights into how changes in cocoa-based products' demand affect cocoa price. Specifically, we will be using China's annual consumption rate.

The **GHS-USD Exchange Rate** dataset provides daily exchange rate data between the Ghanaian Cedi (GHS) and the United States Dollar (USD). Sourced from **investment.com**, this dataset captures the fluctuations in the exchange rate over an extended period, offering valuable insights into the economic relationship between Ghana and the United States. The downloaded data spans from January 1, 2015, to February 27, 2025. This dataset is essential for capturing how global economic conditions, inflation, monetary policy, and market sentiment impact the value of the Ghanaian currency relative to the US dollar. We can use this data to incorporate the economic stability of Ghana, and make informed predictions related to cocoa price.

Data cleaning and EDA

All datasets required cleaning to ensure consistency and accuracy.

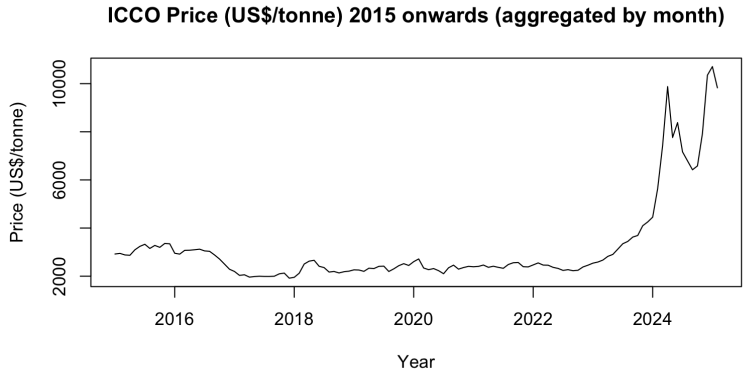
For the **cocoa price** data (January 1, 2015 – February 27, 2025), instances of duplicated dates were identified on four occasions: December 15, 2023, January 9, 2024, January 30, 2024 and January 31, 2024.

Duplicated dates	Values
December 15, 2023	4272, 4272 (identical value)
January 9, 2024	4171, 4171 (identical value)
January 30, 2024	4775, 10676
January 31, 2024	4798, 10888

In the case where duplicated dates have identical values, we only kept one of the observations. In cases where different values were reported for the same date, the lower value was retained, as it aligned better with neighboring data points. Lastly, we aggregated the daily values into averaged monthly values by taking the average of all observations (excluding all observations with missing value) that occurred in the same month.

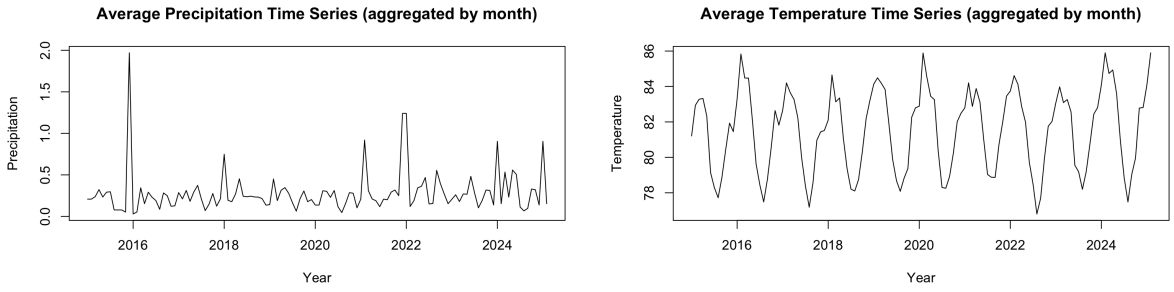
Figure 1 below shows the trajectory plot for monthly cocoa prices. Prices are relatively stable from 2016-2022, but begin increasing in 2022-2024. After 2024, there are dramatic spikes and dips in price.

Figure 1: Monthly Cocoa Prices



For the **Ghana Climate Data**, originally spanning January 1, 1990, to November 28, 2024, the dataset was filtered to match the observation window of January 1, 2015, to November 28, 2024. From this data, we extracted two important variables, precipitation and average temperature, and created two separate datasets (**precipitation data and temperature dataset**), each capturing one of the variables and the date of observations. In both datasets, each row represents an observation from a station. Hence, we took an average across all stations for each date to obtain daily averaged values. Similarly, we aggregated the daily values into averaged monthly values by taking the average of all observations (excluding all observations with missing value) that occurred in the same month. For the **precipitation data**, 3 months contained missing values, which were filled using the last available observation. For the **temperature dataset**, there are no months with missing value. For both **precipitation** and **temperature** datasets, to extend the data beyond November 2024, the values from the previous year were used to impute missing entries up to February, 2025. This approach leverages the seasonal patterns commonly observed in both precipitation and temperature data, where yearly trends tend to repeat. Given the strong seasonality in both datasets, using the previous year's values is a reasonable and efficient method for imputation, ensuring that the temporal structure remains consistent.

Figure 2: Monthly Weather Data in Ghana

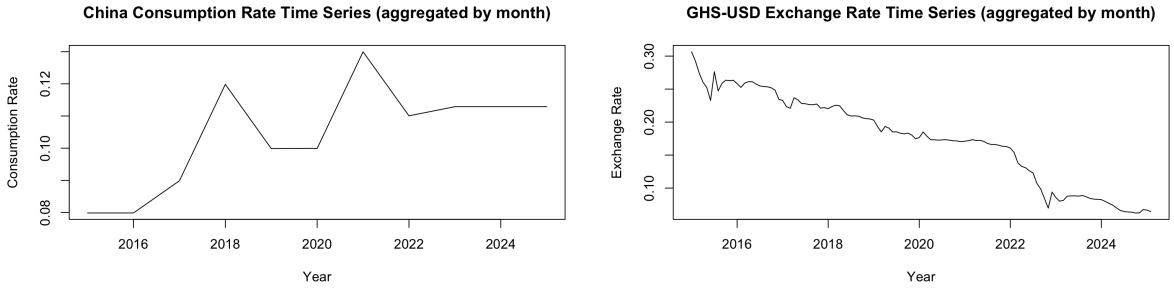


As Figure 2 shows, the average precipitation and temperature in each month seem to follow seasonal cycles. Precipitation was especially high in December of 2015, 2017, 2021, and 2023. Ghana’s temperature appears to peak in the spring of each year and fall in the summer.

For the **Per Capita Consumption of Cocoa Beans** data, originally available on an annual basis from 1961 to 2022, only data for China from 2015 to 2022 was retained. Missing values for the years 2022 to 2025 were imputed using by fitting an Error-Trend-Seasonality model (details available in the Appendix) to the available data and predicting 3 more future yearly values. Since the data was annual, monthly values were assigned by maintaining the same consumption level across all months within a given year.

Finally, we also aggregated the daily values from the **GHS-USD Exchange Rate** data (January 1, 2015 – February 27, 2025) into averaged monthly values by taking the average of all observations (excluding all observations with missing value) that occurred in the same month. There are no months with missing value after aggregation.

Figure 3: Monthly Cocoa Consumption and Exchange Rate



On the left, the cocoa consumption rate has been increasing since 2016, with peaks around 2018 and 2021. On the right, the exchange rate has been steadily falling since 2016, with a severe dip after 2022.

Table 1: Summary Statistics for Model Variables

Variable	Mean	Median	Minimum	Maximum	Standard Deviation
Cocoa prices (monthly)	3169.87	2461.92	1917.68	10709.31	1836.24
Precipitation (mm, monthly average)	0.28	0.23	0.03	1.97	0.25
Daily temperature (°F, (monthly average)	81.52	82.01	76.80	85.90	2.32
Cocoa consumption rate (kg/capita, China, annual)	0.10	0.11	0.080	0.13	0.02
Exchange rate (GHS-USD, monthly)	0.17	0.18	0.06	0.31	0.07

Within our observation window, cocoa prices reached a minimum of $1917.68 in December 2017, and a maximum of $10,709.31 in January 2025. A year earlier in February 2024, the maximum daily temperature of 85.90 °F was reached in Ghana. The exchange rate also reached its minimum of 0.06 in October 2024, a few months before the price peaked. There is likely a lagged effect between change in the predictors and jumps in cocoa price.

Methodology

Model 1: ARIMA

We first fitted a linear regression model with the averaged monthly cocoa price as the response variable and averaged monthly precipitation, temperature, China Per Capita Consumption of Cocoa Beans and GHS-USD Exchange Rate as predictors. Then, we fitted an ARIMA model to the residuals of the model.

Figure 4: Trajectory Plot of Model Residuals

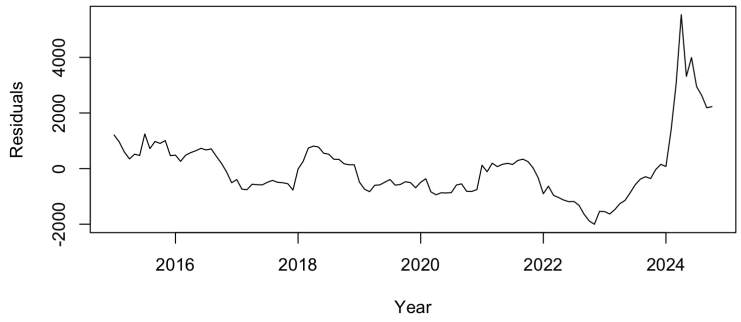
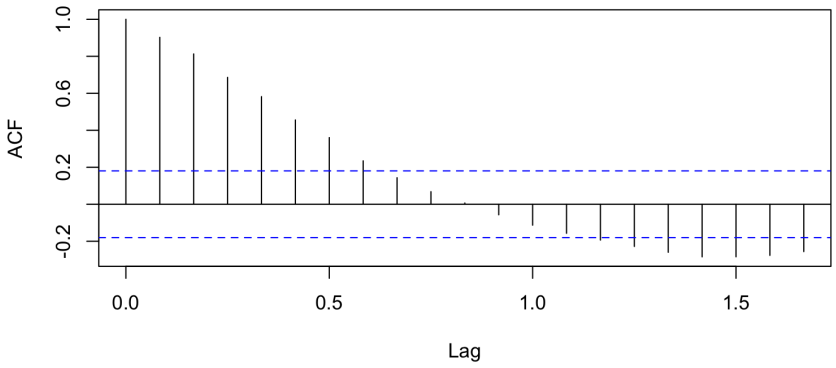


Figure 5: ACF Plot of Model Residuals



The trajectory plot of the residuals (Figure 4) shows a significant trend while the ACF plot (Figure 5) also shows a slow decaying pattern. This implies that the residuals' time series is not stationary, and requires differencing before investigating the AR and MA components' orders.

Figure 6: Trajectory of First Order Difference Residuals

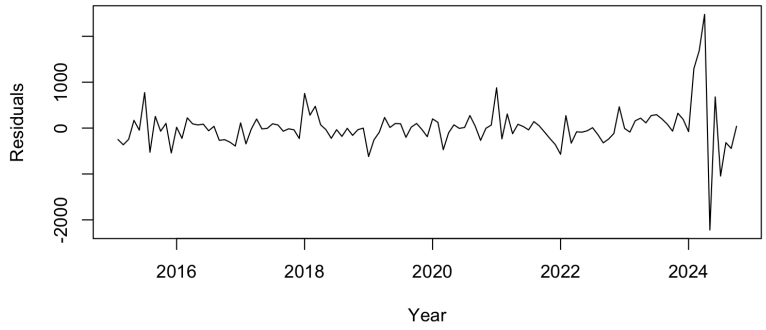
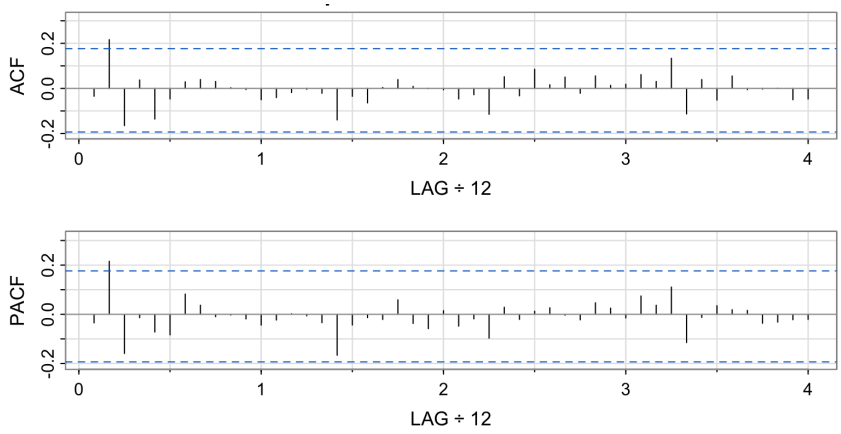


Figure 6: ACF and PACF Plots of First Order Difference Residuals



After differencing the residuals by one order, the trajectory plot shows improvement as it looks more white. We also observed a quicker decay in the corresponding ACF plot. In particular, ACF values become non-significant (i.e. within the blue dotted lines) after lag 2. Similarly, the PACF values also become non-significant after lag 2. There are three possible models to consider if we interpret the plots slightly differently:

- ARIMA(2,1,2): Both ACF and PACF values are tailing off after lag 2.
- ARIMA(2,1,0): PACF values are cut off after lag 2 and ACF values are tailing off.
- ARIMA(0,1,2): ACF values are cut off after lag 2 and PACF values are tailing off.

Since it is not clear whether the ACF values and PACF values are cut off or tailing off after lag 2, we considered all 3 possible models.

The resulting best model is an ARIMA(0,1,2) model, referring to a time domain model with first order differencing and a 2 order moving average component. It is chosen because it

demonstrates the best performance in diagnostic tests and has the lowest AIC value (14.90201) among all candidates. (AIC values and diagnostic plots of other models are available in the Appendix.) Below is the summary of the model coefficient estimates and standard errors.

Table 2: ARIMA Model Summary

Term	Coefficient Estimate	Standard error	P-value
MA(1)	0.059	0.0916	0.5209
MA(2)	0.2288	0.0883	0.0108
Precipitation	-13.3965	97.2938	0.8907
Temperature	13.8097	27.1564	0.6121
China consumption	-2527.337	7424.593	0.7342
Exchange Rate	2431.34	4584.58	0.5969

The model can be written out in the following equation:

$$x_t - x_{t-1} = w_t + 0.0598 w_{t-1} + 0.2288 w_{t-2}$$

where x_t are the residuals of the linear regression model

$$Price_t = -13.3965\ Precipitation_t + 13.8097\ Temperature_t - 2527.337\ China\ consumption_t + 2431.34\ Exchange\ rate_t + x_t$$

Justification of the covariate estimates

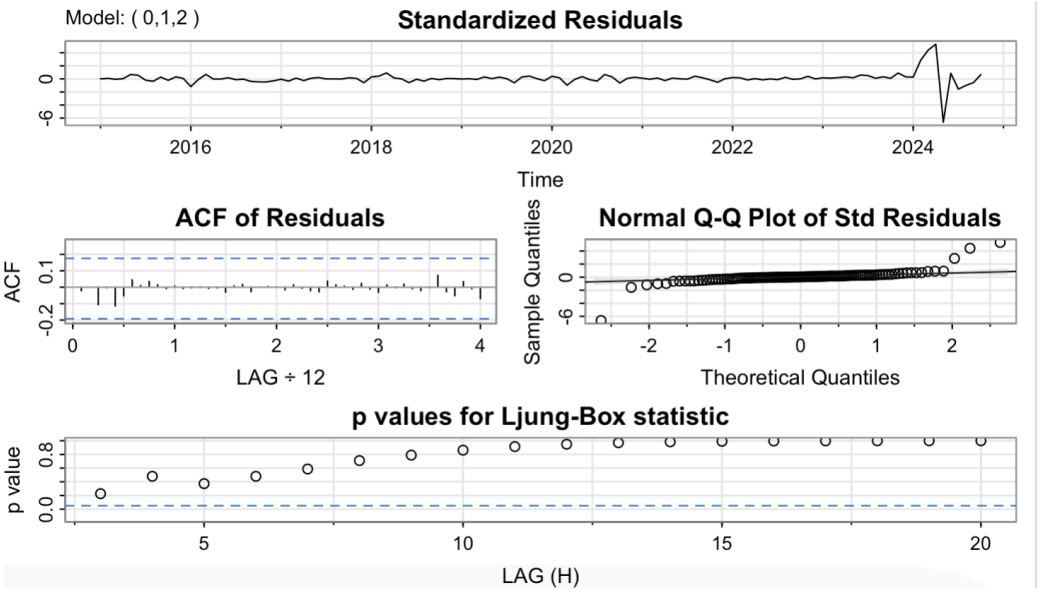
Accounting for the time-related components, our model suggests some key relationships between our predictors and cocoa prices. We can start by looking at climate-related factors in Ghana: precipitation and temperature. Precipitation typically increases cocoa's yield gap due to higher humidity, increasing expected yield from crops. However, damp environments also lead to an increase in black pod disease which significantly reduces crop yield (Asante, et al., 2022; Ritchie, 2024). Despite the prevalence of black pod disease, precipitation appears to increase yield which increases global supply and decreases price. Temperature, on the other hand, appeared to generally reduce cocoa yield by increasing crops' water demand thus slowing down growth (Asante, et al., 2022; Yoroba, et al., 2019). Hence, temperature's effect on cocoa price appears to be opposite with precipitation with higher temperature leading to reduction in supply and increase in price. Our model seems to agree with these correlations with precipitation being negatively correlated with price and temperature being positively correlated with price. However, both variables appear to be insignificant predictors in the model despite prior studies stating their high relevance in cocoa yield. While holding all other predictors constant, it is possible that precipitation or temperature are not significant individually.

Looking at economic factors, we focus on both consumption of chocolate in the China market and exchange rate of Ghana's Cedi (GHS) with U.S. dollar. China, while trailing in total chocolate, has shown significant growth in demand for chocolate products (Interesse, 2024). Increases in China's chocolate consumption would then result in an increase in global chocolate demand which could lead to an increase in price. Interestingly, our model seems to suggest that China's consumptions appear to be negatively correlated with chocolate price. Global chocolate prices appear to decrease as consumption increases which appears opposite from our expectations. The discrepancy in correlation could potentially be explained by the relatively small contribution of China's population in the global cocoa market (Interesse, 2024). The focus on Ghana's data could also affect the estimation of our model as it might not capture the overall trend seen.

Exchange rate, on the other hand, plays a more interesting role towards cocoa prices. Ghana and Cote d'Ivoire implement a price-setting system which periodically sets cocoa's sale price in the country (Olaigbe & Usman, 2025; Tabe-Ojong, Guedegbe, & Glauber, 2024). While the fixed pricing system protects farmers from exploitation and drop in cocoa price, it also means that Ghana farmers' are not able to instantly react to price changes in the cocoa market. Hence, despite the rapid spike in cocoa prices, foreign exchange rate could be expected to have positive but lagged effects towards prices. On the other hand, the exchange rate appears to agree with our expectation as it is positively correlated which suggests that chocolate price tends to increase as GHS appreciates in value. However, our model again suggested that both variables are insignificant in predicting cocoa prices.

Model diagnostics

Figure 7: Residual Diagnostics



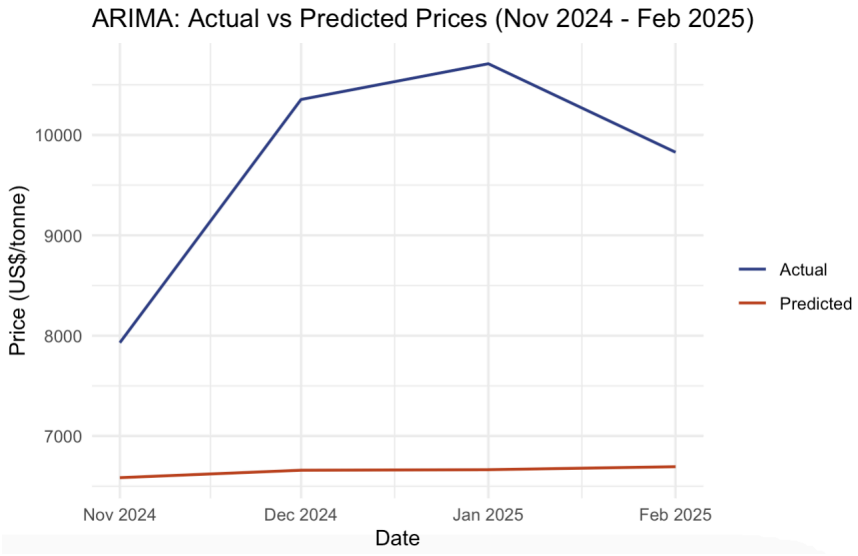
The trajectory plot of the standardized residuals shows an insignificant pattern, resembling white noise. The ACF plot on the residuals indicates that all sample ACFs at any lag values fall within the error bounds, suggesting no significant correlation structure. The Normal Q-Q plot of the residuals shows no stark deviation from the diagonal line, implying that the

residuals are generally normally distributed. The Ljung-Box-Pierce test results reveal that all p-values at any lag are significantly above 0, confirming that the residuals are not correlated.

ARIMA Forecasting and Results

Evaluating model accuracy during the test period (November 2024-February 2025) illustrated a Root Mean Squared Error (RMSE) of approximately 3225.57. As observed from the forecasting plot (Figure 8), the model predicted trajectory (red) fails to capture the characteristics of the actual trajectory (blue) since they are not closely aligned with each other. In particular, the predicted trajectory is not increasing at a comparable rate as the actual trajectory.

Figure 8: Model Forecast



Model 2: Random Forest

To further enhance our forecasting capabilities, we implemented a Random Forest model using the same set of predictors used in the ARIMA model—precipitation, temperature, Chinese cocoa consumption, and exchange rates—alongside additional lagged price variables (1-month and 7-month lags). These lagged features help capture time-dependent relationships more effectively.

Random Forest models, though traditionally associated with classification and regression, have demonstrated effectiveness in time series forecasting due to their ability to detect non-linear relationships and complex interactions among predators. Importantly, Random Forest models do not rely on strict assumptions of linearity or stationarity, and their ensemble-based approach mitigates the risk of overfitting, particularly when dealing with correlated variables and makes them especially effective for forecasting volatile commodities such as cocoa (Breiman, 2001).

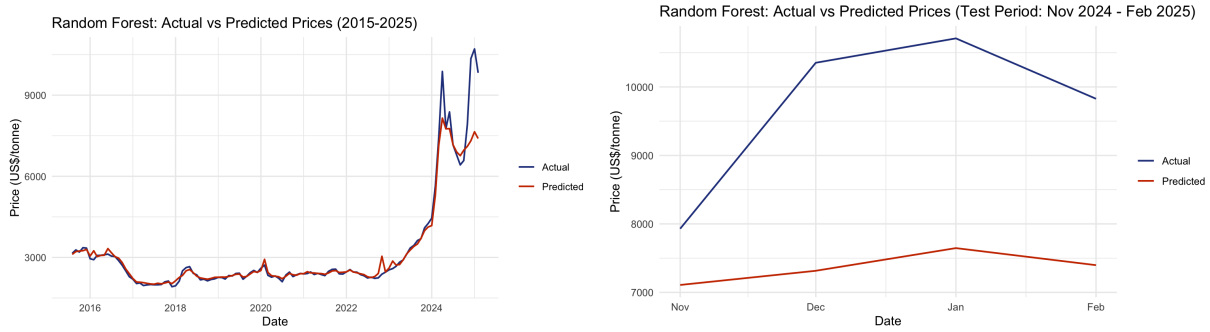
Random Forest Forecasting and Results

The fitted Random Forest model explained approximately 91.03% of the variance in cocoa prices which indicates strong overall predictive power. Among the predictors, the 1-month lagged price variable was most influential, closely followed by the 7-month lagged price and the exchange rate. Climate variables (precipitation and temperature) and Chinese cocoa consumption contributed less substantially which highlights the predominant role of past prices and exchange rates within this predictive context.

Evaluating model accuracy during the test period (November 2024–February 2025) showed a Root Mean Squared Error (RMSE) of approximately 2508.90 and a Mean Absolute Percentage Error (MAPE) of 23.25%. While the Random Forest model effectively captured historical trends throughout the training period (2015-2024), it notably underestimated the sharp price rise occurring in early 2025. This outcome highlights a key limitation that Random Forest models may struggle when faced with abrupt and extreme changes although they are robust in capturing historical patterns.

Visual inspection through Actual vs. Predicted price plots further shows this limitation. The predictions closely match the observed prices throughout most of the analyzed period but deviate significantly during sudden market shifts. Future enhancements could explore incorporating additional influential external variables, adjusting the model's lag structures, or employing hybrid approaches that integrate strengths from both traditional and machine learning methods.

Figure 9: Random Forest Forecast Plots



Our results suggest that the Random Forest model performs slightly better than the ARIMA model when it comes to forecasting cocoa prices. One key difference is that the Random Forest model is better at capturing the general trend, especially during the test period from November 2024 to February 2025. While the ARIMA forecast stays relatively flat, the Random Forest shows a clearer upward shift, which is more in line with what actually happened in early 2025.

This likely comes from the Random Forest model's ability to pick up on more complex patterns and relationships in the data. By using lagged price variables, it can better reflect how past values influence future prices. On the other hand, the ARIMA model relies on linear

assumptions and tends to smooth over sudden changes, which can make it less responsive during volatile periods.

Although the Random Forest model did underestimate the sharp price spikes in early 2025, it still gave a more realistic picture overall. Moving forward, it might be useful to test other models or combinations of models that can handle both long-term trends and sudden shifts. It could also help to bring in more covariates that reflect other known drivers of cocoa prices, such as political instability or pest outbreaks.

Discussion

Conclusion

Our results highlight key correlations between global cocoa prices and various climatic and economic factors. We found that precipitation and China's cocoa consumption were positively correlated with cocoa prices, whereas temperature and the exchange rate (GHS-USD) showed a negative correlation. However, the ARIMA model suggested that none of these factors were significant predictors of cocoa prices. The ARIMA model also exhibited relatively poor forecasting performance, which was improved using a random forest model. Despite this improvement, both models indicate that more advanced time series approaches should be explored. The insignificance of our predictors and the limited predictive performance suggest that additional data and covariates are necessary to better capture global trends. Incorporating data from other major cocoa producers such as Côte d'Ivoire and Indonesia could enhance the model's applicability. Factors including political conditions, pest and disease data, and other economic factors may also improve predictive accuracy.

Limitations

Our approach has several limitations. First, we modelled global cocoa prices based on climate data from Ghana, which only accounts for about 12% of global cocoa production (ICCO, 2025). As a result, temperature and precipitation trends in Ghana may not fully explain the impact of weather on cocoa production in other West African or international countries. Next, we aggregated cocoa prices, temperature, precipitation, and exchange rate from daily values to monthly averages, which could mask finer trends. Similarly, we treated the annual Chinese cocoa consumption as equal across all 12 months of the year, which is likely inaccurate. Imputing the data differently could lead to different models and results. We faced challenges in obtaining predictor data which matched the daily format of cocoa prices, leading to our aggregation decisions. Furthermore, we struggled to find data on other important influences on cocoa prices, such as pest outbreaks and political stability. These missing variables may have limited our models' ability to fully explain cocoa price fluctuations. Another limitation is our observation window, which was restricted to the past 10 years. Notably, the largest residuals in our models occurred in recent years, when cocoa prices spiked dramatically. This suggests that our models may not adequately capture sudden market shifts.

Next Steps

Future work should explore models that operate on daily data to capture short-term fluctuations more effectively. Obtaining predictor data in a daily format, rather than aggregating it, could enhance the precision of our findings. Additionally, expanding the dataset to include a longer historical window could help develop models that account for both recent price surges and past trends while maintaining predictive accuracy. While our random forest model improved predictive performance, its interpretability remains a challenge. Future studies should compare predictive accuracy across other machine learning and traditional econometric models to balance both accuracy and interpretability. Exploring ensemble methods or hybrid approaches may also yield better forecasting results.

References

Addai, B., Gyimah, A. G., & Poku-Agyemang, K. (2020). Exchange rate regimes and global cocoa trade: to float or to peg? *Cogent Economics & Finance, 8*(1).

Asante, P. A., Rahn, E., Zuidema, P. A., Rosendaal, D. M., van der Baan, M. E., Läderach, P., . . . Anten, N. P. (2022, August). The cocoa yield gap in Ghana: A quantification and an analysis of factors that could narrow the gap. *Agricultural Systems, 201*.

Assis, K., Amran, A., & Remali, Y. (2010). Forecasting cocoa bean prices using univariate time series models. *Researchers World*, *1*(1), 71.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Food and Agriculture Organization of the United Nations. (2022) Food Balances: Food Balances (-2013, old methodology and population. [Data set]. https://www.fao.org/faostat/en/#search/Cocoa%20Beans%20and%20products

J.P. Morgan. (2024, December 2). *Rising cocoa prices: Will the chocolate crisis continue in 2025?* https://www.jpmorgan.com/insights/global-research/commodities/cocoa-prices

Kane, M.J., Price, N., & Scotch, M. (2014). Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks. *BMC Bioinformatics* 15, 276. https://doi.org/10.1186/1471-2105-15-276

Kongor, J.E., Owusu, M. & Oduro-Yeboah, C. Cocoa production in the 2020s: challenges and solutions. *CABI Agric Biosci* 5, 102 (2024). https://doi.org/10.1186/s43170-024-00310-6

Interesse, G. (2024, May 23). *China's Chocolate Market – Trends and Industry Overview.* Retrieved March 2025, from China Briefing: https://www.china-briefing.com/news/chinas-chocolate-market-trends-and-industry-overview/

International Cocoa Organization. (2025). *Cocoa daily prices.* [Data set]. https://www.icco.org/statistics/#tab-id-7

Investing.com. (2025). GHS/USD historical data. [Data set]. https://ca.investing.com/currencies/ghs-usd-historical-data

National Centers for Environmental Information. (2025). Ghana Climate Data. [Data set]. https://www.ncei.noaa.gov/

Olaigbe, O., & Usman, I. K. (2025, January 2). *Ghana Is the Second Largest Cocoa Producer; Why Are Its Farmers Still Poor?* Retrieved March 2025, from Pulitzer Center: https://pulitzercenter.org/stories/ghana-second-largest-cocoa-producer-why-are-its-farmers-still-poor

Ritchie, H. (2024, April 1). *The chocolate price spike: what's happening to global cocoa production?* Retrieved March 2025, from Sustainability by Numbers: https://www.sustainabilitybynumbers.com/p/cocoa-prices

Sukiyono, K., Nabiu, M., Sumantri, B., Novanda, R. R., Arianti, N. N., Yuliarso, M. Z., ... & Mustamam, H. (2018, November). Selecting an accurate cacao price forecasting model. In *Journal of Physics: Conference Series* (Vol. 1114, No. 1, p. 012116). IOP Publishing.

Tabe-Ojong, M., Guedegbe, O. T., & Glauber, J. (2024, May 8). *Soaring cocoa prices: Diverse impacts and implications for key West African producers.* Retrieved March 2025, from IFPRI Blog: Issue Post | Markets, Trade, and Institutions: https://www.ifpri.org/blog/soaring-cocoa-prices-diverse-impacts-and-implications-key-west-african-producers/

Thrukral, N. & Tan, F. (2024, December 31). Cocoa tops global commodities rally for 2nd year, steel ingredients struggle on China demand. *Reuters*. https://www.reuters.com/markets/commodities/cocoa-tops-global-commodities-rally-2nd-year-steel-ingredients-struggle-china-2024-12-31/

Tothmihaly, A. (2018). How low is the price elasticity in the global cocoa market? African Journal of Agricultural and Resource Economics, 13(3), 209–223. https://doi.org/10.22004/ag.econ.284986

Wiśniewska, K. (2024, July 26). Cocoa prices - a review of key market factors. *Foodcom.* https://foodcom.pl/en/cocoa-prices-a-review-of-key-market-factors/

Yoroba, F., Kouassi, B. K., Diawara, A., Yapo, L. A., Kouadio, K., Tiemoko, D. T., . . . Assamoi, P. (2019, January 13). Evaluation of Rainfall and Temperature Conditions for a Perennial Crop in Tropical Wetland: A Case Study of Cocoa in Côte d'Ivoire. *Advances in Meteorology, 2019*(1).

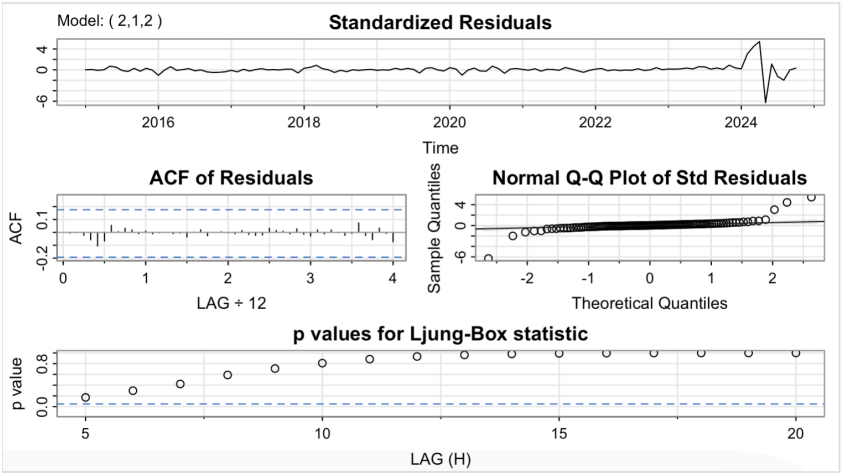
Appendix

- Error-Trend-Seasonality model for China cocoa consumption data imputation
  An ETS(A,N,N) model, which was selected by R function automatically, was fitted to the raw data. In this model, we decomposed the data (y_t) into level (l_t) and addictive error (ε_t) without any trend or seasonality component. The fitted model equations are shown below:

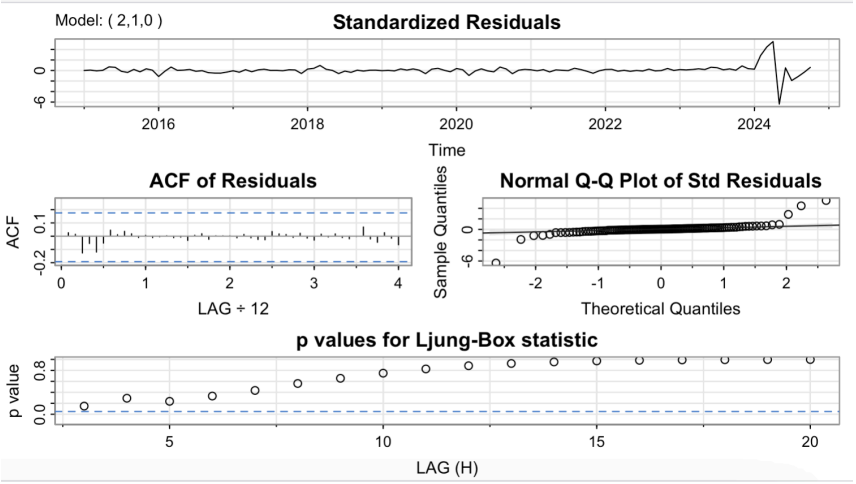
$$y_t = l_{t-1} + \varepsilon_t$$
$$l_t = l_{t-1} + 0.5236\varepsilon_t$$

  where $l_0 = 0.0846$.

- Diagnostic plots and AIC values of additional ARIMA models
  ARIMA(2,1,2)



  AIC = 14.92473
  ARIMA(2,1,0)



  AIC = 14.90808

Full codes

```
#   Load Packages
library(astsa)
library(forecast)
library(ggplot2)
library(gridExtra)
library(tidyverse)
library(dplyr)
library(jsonlite)
library(randomForest)
library(Metrics)
library(lubridate)

#---------------------------- Price Data ---------------------------#
# Load data
price_data = read_csv("Daily Prices_ICCO.csv")

# Format the date column properly
price_data$Date <- as.Date(price_data$Date, format = "%d/%m/%Y")

# Take average of any duplicated days
# 2024-01-31: 4798, 10888
# 2024-01-30: 4775,  10676
# 2024-01-09: 4171, 4171
# 2023-12-15: 4272, 4272

# Select the smaller value for duplicated dates
price_data <- price_data %>%
  group_by(Date) %>%
  summarise(`ICCO daily price (US$/tonne)` = min(`ICCO daily price (US$/tonne)`), .groups =
"drop")

# Aggregate to averaged monthly data
monthly_price = price_data%>%mutate(month = floor_date(Date,"month"))%>%
  group_by(month)%>%
  summarize(monthly_price = mean(`ICCO daily price (US$/tonne)`))

# Convert to time series
price_ts <- ts(monthly_price$monthly_price,
               start = c(1994,
                         10),
               frequency = 12) # Assuming monthly data
```

```
# Trim away 2024 Dec 31
price_ts <- window(price_ts,
            start = c(2015,
                    1),
            frequency = 12)
# 122 months
# Start = c(2015, 1)
# End = c(2025, 2)

# Original time series
# Trajectory plot
plot(price_ts, main = "ICCO Price (US$/tonne) 2015 onwards (aggregated by month)",
    ylab = "Price (US$/tonne)", xlab = "Year",
    type = "l")

# Subset the time series Jan 1 2015 - Dec 31 2024
price_ts_2024 <- window(price_ts, start = c(2015,1), end = c(2024, 10))

#---------------------------- Ghana data--------------------------#
# Load data
ghana_data = read_csv("Ghana_data.csv")
ghana_data$DATE <- as.Date(ghana_data$DATE, format = "%d/%m/%Y")

# Jan 1 1990 to Nov 28 2024

# Create a full sequence of dates
full_months<- seq(as.Date("2015-01-01"), as.Date("2024-11-01"), by = "month")

# Precipitation
# Summarize daily average precipitation
daily_avg_prep <- ghana_data %>%
  filter(!is.na(PRCP)) %>%  # Remove rows where PRCP is NA
  group_by(DATE) %>%
  summarize(avg_prep = mean(PRCP), .groups = "drop")

# Aggregate to averaged monthly data
monthly_avg_prep = daily_avg_prep%>%mutate(month = floor_date(DATE,"month"))%>%
  group_by(month)%>%
  summarize(monthly_prep = mean(avg_prep))

# Merge with full sequence, filling missing dates with NA
monthly_avg_prep <- data.frame(month = full_months) %>%
  left_join(monthly_avg_prep, by = "month")
```

```
# Extract dates where the price is NA
missing_dates_prep <- monthly_avg_prep %>%
  filter(is.na(monthly_prep)) %>%
  select(month)

# Output missing dates
print(missing_dates_prep)
# 3 missing months: 2015-10-01, 2020-02-01, 2022-01-01

# Fill missing values with the immediate previous value
monthly_avg_prep <- monthly_avg_prep %>%
  fill(monthly_prep, .direction = "down")

prep_ts <- ts(monthly_avg_prep$monthly_prep,
              start = c(2015,1),
              frequency = 12)

# Impute future values by copying previous year's data

# Create a sequence of months for missing future values
future_months <- seq(as.Date("2024-12-01"), as.Date("2025-02-01"), by = "month")

# Find the matching months from the previous year
previous_year_months <- seq(as.Date("2023-12-01"), as.Date("2024-02-01"), by = "month")

# Extract the values for the previous year to impute
imputed_values <- monthly_avg_prep$monthly_prep[which(monthly_avg_prep$month %in%
previous_year_months)]

# Convert the imputed values to a ts object
imputed_ts <- ts(imputed_values, start = c(2024, 12), frequency = 12)

# Combine forecasted values with original ts
extended_prep_ts <- ts(c(prep_ts, imputed_ts),
                       start = start(prep_ts),
                       frequency = 12)


# Trajectory plot of daily average prep
plot(extended_prep_ts, main = "Average Precipitation Time Series (aggregated by month)",
     ylab = "Precipitation", xlab = "Year", type = "l")

# Subset the time series up to the last date of 2024
prep_ts_2024 <- window(extended_prep_ts, end = c(2024,10))
```

```
# Temparature
# Summarize daily average precipitation
daily_avg_temp <- ghana_data %>%
  filter(!is.na(TAVG)) %>%  # Remove rows where TAVG is NA
  group_by(DATE) %>%
  summarize(avg_temp = mean(TAVG), .groups = "drop")

# Aggregate to averaged monthly data
monthly_avg_temp = daily_avg_temp%>%mutate(month = floor_date(DATE,"month"))%>%
  group_by(month)%>%
  summarize(monthly_temp = mean(avg_temp))

# Merge with full sequence, filling missing dates with NA
monthly_avg_temp <- data.frame(month = full_months) %>%
  left_join(monthly_avg_temp, by = "month")

# Extract dates where the price is NA
missing_dates_temp <- monthly_avg_temp %>%
  filter(is.na(monthly_temp)) %>%
  select(month)

# 0 missing months

temp_ts <- ts(monthly_avg_temp$monthly_temp,
              start = c(2015,1),
              frequency = 12)

# Impute future values by copying previous year's data

# Extract the values for the previous year to impute
imputed_values <- monthly_avg_temp$monthly_temp[which(monthly_avg_temp$month %in%
previous_year_months)]

# Convert the imputed values to a ts object
imputed_ts <- ts(imputed_values, start = c(2024, 12), frequency = 12)

# Combine forecasted values with original ts
extended_temp_ts <- ts(c(temp_ts, imputed_ts),
                       start = start(temp_ts),
                       frequency = 12)

# Subset the time series up to the last date of 2024
```

```
temp_ts_2024 <- window(extended_temp_ts, end = c(2024,10))

# Trajectory plot of daily average prep
plot(extended_temp_ts, main = "Average Temperature Time Series (aggregated by month)",
    ylab = "Temperature", xlab = "Year", type = "l")

#---------------------------- Consumption data--------------------------#
# Load data
consumption_data <-
read.csv("https://ourworldindata.org/grapher/chocolate-consumption-per-person.csv?v=1&csvTy
pe=full&useColumnShortNames=true")

# Extract only years from 2015 to 2022
consumption_data = consumption_data %>%filter(Year>2014)
consumption_data <- consumption_data %>% rename(Cocoa_Consumption =
cocoa_beans__00002633__food_available_for_consumption__0645pc__kilograms_per_year_p
er_capita)
# Focus on China
china_consumption_data = consumption_data%>%filter(Entity == "China")

# Yearly time series
china_cocoa_ts <- ts(china_consumption_data$Cocoa_Consumption, start =
min(china_consumption_data$Year), frequency = 1)
# Trajectory plot of cocoa consumption
plot(china_cocoa_ts,
    main = "Cocoa bean consumption per person in China Over Time",
    xlab = "Year",
    ylab = "Kilograms per person",
    type = "l")

# ETS
china_ets = ets(china_cocoa_ts)
china_cocoa_ts_predict = predict(china_ets, 3)$mean

# Create new time series
extended_china_cocoa_ts <- ts(c(china_cocoa_ts, china_cocoa_ts_predict), start =
start(china_cocoa_ts), frequency = frequency(china_cocoa_ts))

# Function to expand annual data to monthly data
expand_to_monthly <- function(annual_ts) {
  # Get the number of years in the time series
  years <- length(annual_ts)

  # Create a vector to store monthly values
```

```
  monthly_values <- numeric()

  # Loop over each year and repeat the annual value for 12 months
  for (i in 1:years) {
    # Repeat the value for all months in the year
    monthly_values <- c(monthly_values, rep(annual_ts[i], 12))
  }

  # Create the monthly time series with the expanded values
  monthly_ts <- ts(monthly_values, start = c(2015, 1), frequency = 12)
  return(monthly_ts)
}

# Expand the combined time series to monthly data
china_cocoa_ts_monthly <- expand_to_monthly(extended_china_cocoa_ts)
china_cocoa_ts_monthly <- window(china_cocoa_ts_monthly, start = c(2015,1), end = c(2025,
2))

# Trajectory plot (annual)
plot(extended_china_cocoa_ts, main = "China Consumption Rate Time Series (aggregated by
month)",
     ylab = "Consumption Rate", xlab = "Year", type = "l")

# Subset the time series up to the last date of 2024
china_cocoa_ts_2024 <- window(china_cocoa_ts_monthly, end = c(2024, 10))

#-----------------------Exchange rate data--------------------------#
# Load data
exchange_rate_data = read_csv("GHS_USD Historical Data.csv")

# Format the date column properly
exchange_rate_data$Date <- as.Date(exchange_rate_data$Date, format = "%d/%m/%Y")
exchange_rate_data = exchange_rate_data%>%select(Date, Price)

# Aggregate to averaged monthly data
monthly_exchange = exchange_rate_data%>%mutate(month = floor_date(Date,"month"))%>%
  group_by(month)%>%
  summarize(monthly_exchange = mean(Price))

# Convert to time series
exchange_ts <- ts(monthly_exchange$monthly_exchange,
             start = c(2015,1),
             frequency = 12)
```

```
# Trajectory plot
plot(exchange_ts, main = "GHS-USD Exchange Rate Time Series (aggregated by month)",
     ylab = "Exchange Rate", xlab = "Year", type = "l")

# Trim away 2024 Dec 31
exchange_ts_2024 <- window(exchange_ts,
                  end = c(2024,10),
                  frequency = 12)

#----------------------Numerical summary table-------------------------------#
fractional_to_date <- function(fractional_year) {
  # If the fractional year is exactly an integer, use the first day of the year
  if (fractional_year %% 1 == 0) {
    year <- floor(fractional_year)
    return(as.Date(paste(year, "01", "01", sep = "-")))  # January 1st of the year
  }

  # Ensure that the fractional year is numeric and within a valid range
  if (is.na(fractional_year) || fractional_year < 1900 || fractional_year > 2100) {
    return(NA)  # Return NA if invalid
  }

  year <- floor(fractional_year)  # Extract the year
  days_in_year <- 365.25  # Average days in a year considering leap years
  day_of_year <- round((fractional_year - year) * days_in_year)  # Calculate the day of the year

  # Make sure day_of_year is within the valid range (1-366)
  if (day_of_year < 1 || day_of_year > 366) {
    return(NA)  # Return NA if the day is out of bounds
  }

  # Create a Date object from the year and day of the year
  date <- as.Date(paste(year, day_of_year, sep = "-"), format = "%Y-%j")
  return(date)
}



# Create a summary function
summary_table <- function(ts_data) {
  # Get the index and time of the maximum value
  max_index <- which.max(ts_data)
  max_date <- time(ts_data)[max_index]
  max_date <- fractional_to_date(max_date)  # Convert to Date
```

```
  # Get the index and time of the minimum value
  min_index <- which.min(ts_data)
  min_date <- time(ts_data)[min_index]
  min_date <- fractional_to_date(min_date)  # Convert to Date

  # Create the summary table
  data.frame(
    Mean = mean(ts_data, na.rm = TRUE),
    Median = median(ts_data, na.rm = TRUE),
    Min = min(ts_data, na.rm = TRUE),
    Min_Date = min_date,  # Date when the minimum value occurred
    Max = max(ts_data, na.rm = TRUE),
    Max_Date = max_date,  # Date when the maximum value occurred
    SD = sd(ts_data, na.rm = TRUE)
  )
}

# Combine the summaries for all time series into a table
summary_results <- bind_rows(
  price_ts = summary_table(price_ts),
  extended_prep_ts = summary_table(extended_prep_ts),
  extended_temp_ts = summary_table(extended_temp_ts),
  extended_china_cocoa_ts = summary_table(extended_china_cocoa_ts),
  exchange_ts = summary_table(exchange_ts),
  .id = "Time_Series"
)

# Print the summary table
print(summary_results)

#-----------------------Auto-ARIMA with predictors--------------------#

predictors = cbind(prep_ts_2024,temp_ts_2024,china_cocoa_ts_2024,exchange_ts_2024)

summary(fit <- lm(price_ts_2024~prep_ts_2024 + temp_ts_2024 + china_cocoa_ts_2024+
exchange_ts_2024, na.action=NULL))
plot(resid(fit),
     main = "Trajectory plot of residuals",
     xlab = "Year",
     ylab= "Residuals")
acf(resid(fit),
     main = "ACF plot of residuals")
# Residuals of regression model not stationary, need to take difference
```

```
plot(diff(resid(fit)),
     main = "Trajectory plot of first order difference residuals",
     xlab = "Year",
     ylab= "Residuals")
acf2(diff(resid(fit)),
     main = "ACF and PACF plots of first order difference residuals")
# First diff Residuals seems stationary

# Investigate potential models
sarima(price_ts_2024, 2,1,2, xreg = predictors)
sarima(price_ts_2024, 2,1,0, xreg = predictors)
sarima(price_ts_2024, 0,1,2, xreg = predictors)

# ARIMA Forcasting
price_future_actual <- window(price_ts, start = c(2024,11))
prep_future = window(extended_prep_ts, start = c(2024,11))
temp_future = window(extended_temp_ts, start = c(2024,11))
cocoa_future = window(china_cocoa_ts_monthly, start = c(2024,11))
exchange_future = window(exchange_ts, start = c(2024,11))

X_future = cbind(prep_future,temp_future,cocoa_future,exchange_future)

forecast_values = sarima.for(price_ts_2024, n.ahead = 4, p =2,d=1,q=0, xreg = predictors,
newxreg=X_future)

arima_forecast_values <- data.frame(
  Date = as.Date(c("2024-11-01", "2024-12-01", "2025-01-01", "2025-02-01")),
  Forecast = forecast_values$pred,
  Actual = price_future_actual
)


# Plot ARIMA forcasted values
ggplot(arima_forecast_values, aes(x = Date)) +
  geom_line(aes(y = Actual, color = "Actual"), size = 0.7) +
  geom_line(aes(y = Forecast, color = "Predicted"), size = 0.7) +
  labs(title = "ARIMA: Actual vs Predicted Prices (Nov 2024 - Feb 2025)",
       y = "Price (US$/tonne)",
       x = "Date") +
  scale_x_date(date_labels = "%b %Y") +  # Formats x-axis labels
  scale_color_manual("", values = c("Actual" = "royalblue4", "Predicted" = "orangered3")) +
  theme_minimal()

# Extract the forecasted mean (predicted values)
```

```
y_pred <- forecast_values$pred

# Calculate RMSE
sqrt(mean((price_future_actual - y_pred)^2))
# 3225.57

#----------Random forest----------#
# Create a date vector for the training period (monthly)
# We assume that price_ts_2024 is the monthly price time series from Jan 2015 to Oct 2024
train_dates <- seq.Date(from = as.Date("2015-01-01"), to = as.Date("2024-10-01"), by =
"month")

# Build the training data frame using the monthly time series
df_train <- data.frame(
  Date = train_dates,
  Price = as.numeric(price_ts_2024),
  Prep = as.numeric(prep_ts_2024),
  Temp = as.numeric(temp_ts_2024),
  Consumption = as.numeric(china_cocoa_ts_2024),
  Exchange = as.numeric(exchange_ts_2024)
)

# Create lag features (1-month and 7-month lags)
df_train <- df_train %>%
  arrange(Date) %>%
  mutate(
    Price_lag1 = lag(Price, 1),
    Price_lag7 = lag(Price, 7)
  ) %>%
  na.omit()  # Remove rows with NA values due to lagging

# Fit Random Forest Model on training data
set.seed(109)
rf_model <- randomForest(Price ~ Price_lag1 + Price_lag7 + Prep + Temp + Consumption +
Exchange,
                         data = df_train,
                         importance = TRUE)
print(rf_model)
print(importance(rf_model))

# In-sample predictions for training period
train_predictions <- predict(rf_model, newdata = df_train)
df_train$Predicted <- train_predictions
```

```
## Prepare Test Data
# For testing, we now predict for the period November 2024 to February 2025
test_dates <- seq.Date(from = as.Date("2024-11-01"), to = as.Date("2025-02-01"), by = "month")
df_test <- data.frame(
  Date = test_dates,
  Price = as.numeric(window(price_ts, start = c(2024, 11), end = c(2025, 2))),
  Prep = as.numeric(window(extended_prep_ts, start = c(2024, 11), end = c(2025, 2))),
  Temp = as.numeric(window(extended_temp_ts, start = c(2024, 11), end = c(2025, 2))),
  Consumption = as.numeric(window(china_cocoa_ts_monthly, start = c(2024, 11), end =
c(2025, 2))),
  Exchange = as.numeric(window(exchange_ts, start = c(2024, 11), end = c(2025, 2)))
)

# To create lag features in the test set, combine the last 7 months of training with the test set
last_train <- tail(df_train, 7)
df_test_full <- bind_rows(last_train, df_test) %>%
  arrange(Date) %>%
  mutate(
    Price_lag1 = lag(Price, 1),
    Price_lag7 = lag(Price, 7)
  )

# Keep only rows from the test period (from November 2024 onward)
df_test_final <- df_test_full %>% filter(Date >= as.Date("2024-11-01"))

# Make predictions on the test data
rf_test_predictions <- predict(rf_model, newdata = df_test_final)
df_test_final$Predicted <- rf_test_predictions

# Calculate error metrics for the test period
rmse_val <- rmse(df_test_final$Price, rf_test_predictions)
mape_val <- mape(df_test_final$Price, rf_test_predictions)
print(paste("Test RMSE:", rmse_val))
print(paste("Test MAPE:", mape_val))

# Combine training and test results for plotting
df_combined <- bind_rows(
  df_train %>% select(Date, Price, Predicted),
  df_test_final %>% select(Date, Price, Predicted)
)

# Plot Actual vs Predicted over the Entire Period
ggplot(df_combined, aes(x = Date)) +
  geom_line(aes(y = Price, color = "Actual"), size = 0.7) +
```

```
  geom_line(aes(y = Predicted, color = "Predicted"), size = 0.7) +
  labs(title = "Random Forest: Actual vs Predicted Prices (2015-2025)",
       y = "Price (US$/tonne)",
       x = "Date") +
  scale_color_manual("", values = c("Actual" = "royalblue4", "Predicted" = "orangered3"))+
  theme_minimal()

# Plot Actual vs Predicted for the Test Period (Nov 2024 - Feb 2025)
ggplot(df_test_final, aes(x = Date)) +
  geom_line(aes(y = Price, color = "Actual"), size = 0.7) +
  geom_line(aes(y = Predicted, color = "Predicted"), size = 0.7) +
  labs(title = "Random Forest: Actual vs Predicted Prices (Test Period: Nov 2024 - Feb 2025)",
       y = "Price (US$/tonne)",
       x = "Date") +
  scale_color_manual("", values = c("Actual" = "royalblue4", "Predicted" = "orangered3"))+
  theme_minimal()
```