University of Chicago MS in Applied Data Science Supplement
Time Series Modeling for Cocoa Price Forecasting

Xuanqi Wei

```r
# GitHub repository for this project:
# https://github.com/Henry-wxq/Time_Series_Modeling_for_Cocoa_Price_Forecasting/tree/main

# load required libraries
library(tidyverse)
library(lubridate)
library(rugarch)
```

Data Preprocessing

```r
# load and preprocess price data
cocoa_prices <- read.csv("Daily Prices_ICCO.csv", stringsAsFactors = FALSE)
cocoa_prices$Date <- as.Date(cocoa_prices$Date, format='%d/%m/%Y')
cocoa_prices$Price <- as.numeric(gsub(",", "", cocoa_prices$ICCO.daily.price..US..tonne.))
cocoa_prices <- cocoa_prices %>%
  mutate(YearMonth = floor_date(Date, "month")) %>%
  group_by(YearMonth) %>%
  summarise(Price = mean(Price, na.rm = TRUE)) %>%
  ungroup()
```

```r
# load and preprocess Ghana weather data
ghana_weather <- read.csv("Ghana_data.csv", stringsAsFactors = FALSE)
ghana_weather$DATE <- as.Date(ghana_weather$DATE)
ghana_weather <- ghana_weather %>%
  mutate(YearMonth = floor_date(DATE, "month")) %>%
  group_by(YearMonth) %>%
  summarise(across(c(PRCP, TAVG, TMAX, TMIN), mean, na.rm = TRUE))
```

```r
# Merge and Clean Monthly Data(log + diff)
cocoa_data <- left_join(cocoa_prices, ghana_weather, by = "YearMonth") %>%
  mutate(log_price = log(Price),
         diff_log_price = c(NA, diff(log_price))) %>%
  drop_na()
```

```r
# Split Data into Training and Testing Sets(7:3 ratio)
train_size <- floor(0.7 * nrow(cocoa_data))
train_data <- cocoa_data[1:train_size, ]
test_data <- cocoa_data[(train_size + 1):nrow(cocoa_data), ]
```

```r
# built GARCH model
log_returns <- diff(log(cocoa_data$Price))
log_returns <- na.omit(log_returns)
```

```r
train_size <- floor(0.7 * length(log_returns))
train_returns <- log_returns[1:train_size]
test_returns <- log_returns[(train_size + 1):length(log_returns)]
test_dates <- cocoa_data$YearMonth[(train_size + 2):(length(log_returns) + 1)]
garch_spec <- ugarchspec(
  variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(1, 0), include.mean = TRUE),
  distribution.model = "norm"
)
garch_fit <- ugarchfit(spec = garch_spec, data = train_returns)
garch_forecast <- ugarchforecast(garch_fit, n.ahead = length(test_returns))
predicted_returns <- as.numeric(fitted(garch_forecast))
last_train_price <- cocoa_data$Price[train_size + 1]

# back-transform forecasted values
forecast_prices <- last_train_price * exp(cumsum(predicted_returns))
garch_df <- tibble(Date = test_dates, Price = forecast_prices)

# plot GARCH Forecast
ggplot() +
  geom_line(data = cocoa_data, aes(x = YearMonth, y = Price), color = "black") +
  geom_line(data = garch_df, aes(x = Date, y = Price), color = "red") +
  labs(title = "GARCH Forecast vs Actual Prices (Monthly)", y = "Price", x = "Date") +
  theme_minimal()
```


GARCH Forecast vs Actual Prices (Monthly)