Time Series Econometrics Project: Interactions Between Implied Volatility, Interest Rates, and Stock Market Dynamics

Dan Allouche

2026-01-04

Contents

Introduction

Financial markets have been through quite a lot since the mid-2000s: the 2008 global financial crisis, years of near-zero interest rates, the COVID-19 pandemic, and then the aggressive monetary tightening that followed. Amid all this turbulence, understanding how implied volatility, interest rates, and stock markets actually interact becomes crucial—not just for academics, but for anyone trying to make sense of how risk, uncertainty, and policy changes ripple through markets.

This project tackles a straightforward but important question: *How do implied volatility, long-term interest rates, and equity market levels interact over time?* To get at this, we focus on three financial variables that are clearly connected, though in ways that aren't always obvious:

- **CBOE Volatility Index (VIX)**: Often called the "fear index," the VIX captures the market's expectation of near-term volatility from S&P 500 options. When markets get nervous, the VIX spikes—sometimes dramatically. It's forward-looking, giving us a 30-day window into expected volatility, which makes it useful for understanding how markets are feeling about risk.

- **S&P 500 Index (SP500)**: This is the broadest measure of US equity performance, tracking 500 large companies. It's forward-looking in the sense that it reflects what investors think about earnings, the economy, and risk. Most portfolios are benchmarked against it, so it's central to how people think about equity markets.

- **10-Year US Treasury Yield (US10Y)**: This yield tells us about long-term interest rate expectations and what investors demand as compensation for risk. It's a key macro indicator—when it moves, it's usually saying something about monetary policy, inflation expectations, or the economic outlook. It also matters for equity pricing because it's the discount rate people use to value future cash flows.

These variables are clearly connected, though the relationships aren't always straightforward. Monetary policy moves affect both interest rates and stock prices—when rates change, it shifts the discount rate for equities and signals what policymakers think about the economy. Market stress (think VIX spikes) usually happens alongside stock market drops, creating that familiar negative correlation between volatility and returns. Interest rates also matter for equity valuations: higher yields mean lower present values for future cash flows, which can drag down stock prices. And equity movements themselves can signal shifts in economic expectations, which then feed back into volatility and interest rates.

0.1 Research Objectives and Methodology

Why does this matter? For one thing, how equities, volatility, and interest rates interact directly affects how people allocate portfolios, manage risk, and hedge positions. If we can understand how shocks move through these variables, investors might be able to anticipate market moves better. From a macro perspective, the link between stocks and long-term rates is central to understanding how monetary policy actually transmits through markets. And volatility indices like the VIX tell us a lot about market sentiment—especially during crises.

We use weekly data from January 2005 to December 2024—more than 20 years covering the 2008 crisis, the zero-rate era, COVID-19, and the recent tightening cycle. Weekly frequency gives us enough observations for solid statistical work while still capturing meaningful market dynamics.

The approach combines univariate and multivariate methods. We start by looking at each series individually: plotting them, checking autocorrelations, testing for stationarity, and then fitting an ARMA model to the S&P 500 returns. Then we move to multivariate analysis: building a VAR model, testing for Granger causality, running impulse-response functions (using both Cholesky decomposition and local projections), and testing for cointegration with Johansen tests. If cointegration shows up, we estimate a VECM instead of just a VAR in differences.

0.2 Research Hypotheses

Based on theory and what we know from the literature, here's what we expect to find:

Equities → Volatility (SP500 → VIX): When stocks drop, volatility should spike. This is pretty well-established—the VIX is called the "fear index" for a reason. We expect this to show up clearly in both causality tests and impulse-response functions.

Equities → Interest Rates (SP500 → US10Y): A stock market decline might signal weaker economic expectations, which could push yields down. But we expect this effect to be **relatively weak**—probably smaller than the equities-volatility link.

Volatility → Equities (VIX → SP500): This one's trickier. While high volatility is clearly risky, we don't expect VIX to strongly predict stock returns. Stocks tend to lead volatility, not the other way around. So we're looking for **low or insignificant** Granger causality here.

Interest Rates → Equities (US10Y → SP500): Yields reflect macro trends that move slowly compared to equity markets. At weekly frequency, we expect **weak or negligible** effects on stock returns.

Long-run relationships: Since both the S&P 500 and 10-year yield are likely I(1), there might be a long-run equilibrium tying them together. The VIX, being stationary, probably enters this indirectly. We're testing for **at least one cointegrating relationship**.

Running these through the full econometric toolkit should tell us something useful about how these markets actually interact—both in the short run and over longer horizons. The rest of the paper works through the univariate analysis first, then moves to the multivariate framework.

---

1 Selection of Time Series

1.1 Series Selection and Rationale

We select **three time series** for this analysis: the S&P 500 Index (SP500), the CBOE Volatility Index (VIX), and the 10-Year US Treasury Yield (US10Y). As detailed in the Introduction, these variables capture essential dimensions of financial markets—equity levels, risk expectations, and the cost of capital—and their interactions are fundamental to understanding how monetary policy, risk sentiment, and asset prices co-move over time.

1.2 Data Characteristics

The data consist of weekly observations covering the period from January 2005 to December 2024, providing a total of 1044 observations for each series. This period, which spans more than 20 years, ensures sufficient variability and statistical precision while capturing a wide range of market regimes, as discussed in the Introduction.

The data were extracted from Yahoo Finance using the `tidyquant` package in R. We download weekly closing prices for the three series:

```r
# Load required packages for data manipulation and visualization
library(tidyquant)
library(dplyr)
library(knitr)
library(kableExtra)
library(purrr)
```

```r
library(ggplot2)
library(scales)

# Define the study period: from January 2005 to December 2024
start_date <- "2005-01-01"
end_date   <- "2024-12-31"

# Download weekly data from Yahoo Finance for each series
# S&P 500 Index (ticker: ^GSPC)
sp500 <- tq_get("^GSPC", from = start_date, to = end_date, periodicity = "weekly") |>
  dplyr::select(date, SP500 = close)

# CBOE Volatility Index (ticker: ^VIX)
vix <- tq_get("^VIX", from = start_date, to = end_date, periodicity = "weekly") |>
  dplyr::select(date, VIX = close)

# 10-Year US Treasury Yield (ticker: ^TNX)
us10y <- tq_get("^TNX", from = start_date, to = end_date, periodicity = "weekly") |>
  dplyr::select(date, US10Y = close)

# Merge the three series by date to create a unified dataset
data_weekly_raw <- purrr::reduce(list(sp500, vix, us10y), dplyr::full_join, by = "date") |>
  dplyr::arrange(date)

# Create a summary of key descriptive information
data_overview <- list(
  start = format(min(data_weekly_raw$date), "%Y-%m-%d"),
  end = format(max(data_weekly_raw$date), "%Y-%m-%d"),
  freq = "Weekly",
  total_obs = nrow(data_weekly_raw),
  src = "Yahoo Finance (^GSPC, ^VIX, ^TNX)",
  obs_SP500 = sum(!is.na(data_weekly_raw$SP500)),
  obs_VIX = sum(!is.na(data_weekly_raw$VIX)),
  obs_US10Y = sum(!is.na(data_weekly_raw$US10Y)),
  n_NA = sum(is.na(data_weekly_raw))
)

# Display data summary
cat(sprintf(
  "=== DATA SUMMARY ===\nPeriod: %s to %s | Frequency: %s\n# Observations: %d (SP500:%d / VIX:%d / US10Y
  data_overview$start, data_overview$end, data_overview$freq,
  data_overview$total_obs, data_overview$obs_SP500, data_overview$obs_VIX, data_overview$obs_US10Y,
  data_overview$n_NA,
  data_overview$src
))
```

```
## === DATA SUMMARY ===
## Period: 2005-01-01 to 2024-12-28 | Frequency: Weekly
## # Observations: 1044 (SP500:1044 / VIX:1044 / US10Y:1044)
## Missing values: 0
## Source: Yahoo Finance (^GSPC, ^VIX, ^TNX)
```

```
# Display first 10 observations of the dataset
knitr::kable(head(data_weekly_raw, 10),
            caption = "Preview of the Weekly Raw Data (2005-2024)",
            booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
               font_size = 9, full_width = FALSE) |>
  column_spec(1, width = "2cm")
```

Table 1: Preview of the Weekly Raw Data (2005–2024)

date	SP500	VIX	US10Y
2005-01-01	1186.19	13.49	4.285
2005-01-08	1184.52	12.43	4.216
2005-01-15	1167.87	14.36	4.142
2005-01-22	1171.36	13.24	4.138
2005-01-29	1203.03	11.21	4.073
2005-02-05	1205.30	11.43	4.095
2005-02-12	1201.59	11.18	4.260
2005-02-19	1211.37	11.49	4.272
2005-02-26	1222.12	11.94	4.310
2005-03-05	1200.08	12.80	4.535

```
# Display summary statistics for the three series
knitr::kable(summary(data_weekly_raw[c("SP500", "VIX", "US10Y")]),
            caption = "Summary Statistics of Raw Weekly Data",
            booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
               font_size = 9, full_width = FALSE)
```

Table 2: Summary Statistics of Raw Weekly Data

SP500	VIX	US10Y
Min. : 683.4	Min. : 9.14	Min. :0.536
1st Qu.:1315.1	1st Qu.:13.15	1st Qu.:2.006
Median :1987.0	Median :16.48	Median :2.756
Mean :2358.6	Mean :18.97	Mean :2.901
3rd Qu.:3007.8	3rd Qu.:22.02	3rd Qu.:3.838
Max. :6090.3	Max. :79.13	Max. :5.228

The dataset provides a balanced weekly panel from 2005 to 2024 with no missing values, making it ideal for time-series analysis. The S&P 500 exhibits a strong upward trend with wide variation, the VIX captures both calm and turbulent market periods, and the 10-year Treasury yield reflects the transition from historically low to higher interest rates.

Note on seasonal adjustment: None of the series require seasonal adjustment, as they are financial market indicators rather than macroeconomic aggregates.

```
# Export the dataset
write.csv(data_weekly_raw,
         "weekly_raw_values_SP500_VIX_US10Y_2005_2024.csv",
         row.names = FALSE)
```

2 Graphical Representations and Autocorrelograms

We begin our analysis with a visual exploration of the three time series to identify trends, structural breaks, volatility patterns, and potential non-stationarity, which will guide our subsequent modeling decisions.

2.1 Graphical Representations of the Series

We first examine the graphical representations of each series in levels to understand their long-run behavior and identify key episodes that may have influenced their dynamics.

```r
# Define a consistent theme for all plots
plot_theme <- theme_minimal(base_size = 9) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 10),
    plot.subtitle = element_text(hjust = 0.5, size = 8),
    axis.title = element_text(size = 9),
    axis.text = element_text(size = 8),
    panel.grid.minor = element_blank(),
    plot.margin = margin(5, 5, 5, 5, "pt")
  )

# Create time series plots for each variable
# S&P 500 Index plot
p1 <- ggplot(data_weekly_raw, aes(x = date, y = SP500)) +
  geom_line(color = "#2E86AB", linewidth = 0.6) +
  labs(title = "S&P 500 Index (Weekly, 2005-2024)",
       x = "Date", y = "Index Level") +
  plot_theme +
  scale_x_date(date_labels = "%Y", date_breaks = "3 years")

# VIX Index plot
p2 <- ggplot(data_weekly_raw, aes(x = date, y = VIX)) +
  geom_line(color = "#A23B72", linewidth = 0.6) +
  labs(title = "VIX Index (Weekly, 2005-2024)",
       x = "Date", y = "Index Level") +
  plot_theme +
  scale_x_date(date_labels = "%Y", date_breaks = "3 years")

# 10-Year Treasury Yield plot
p3 <- ggplot(data_weekly_raw, aes(x = date, y = US10Y)) +
  geom_line(color = "#6A994E", linewidth = 0.6) +
  labs(title = "10-Year Treasury Yield (Weekly, 2005-2024)",
       x = "Date", y = "Yield (%)") +
  plot_theme +
  scale_x_date(date_labels = "%Y", date_breaks = "3 years")

# Display the three plots
p1
```
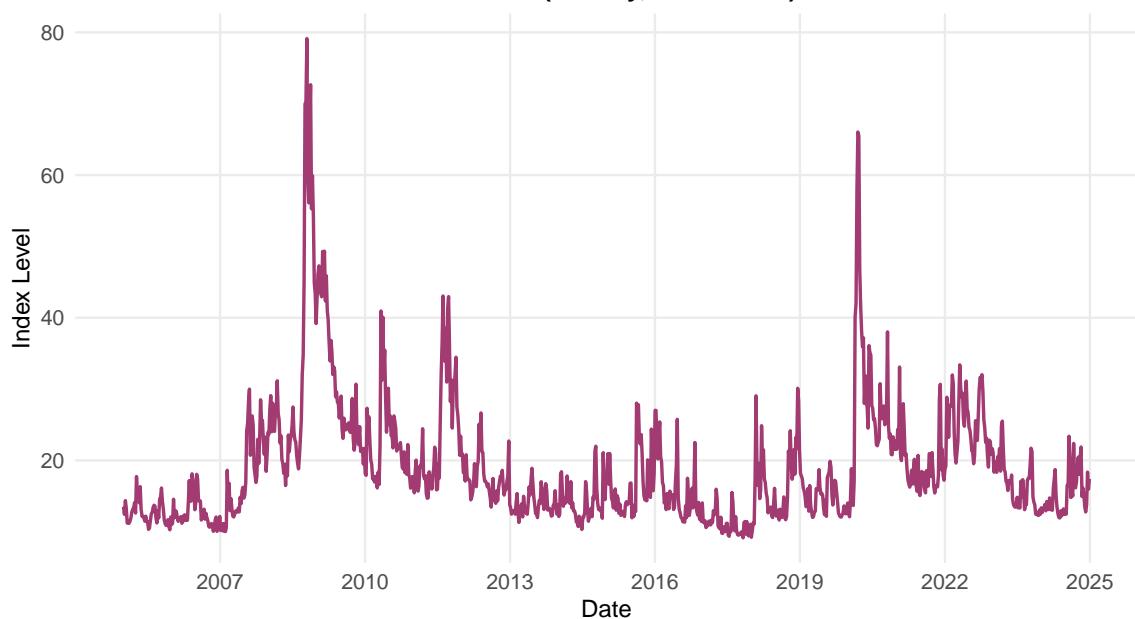
S&P 500 Index (Weekly, 2005–2024)

p2

VIX Index (Weekly, 2005–2024)

p3

10–Year Treasury Yield (Weekly, 2005–2024)
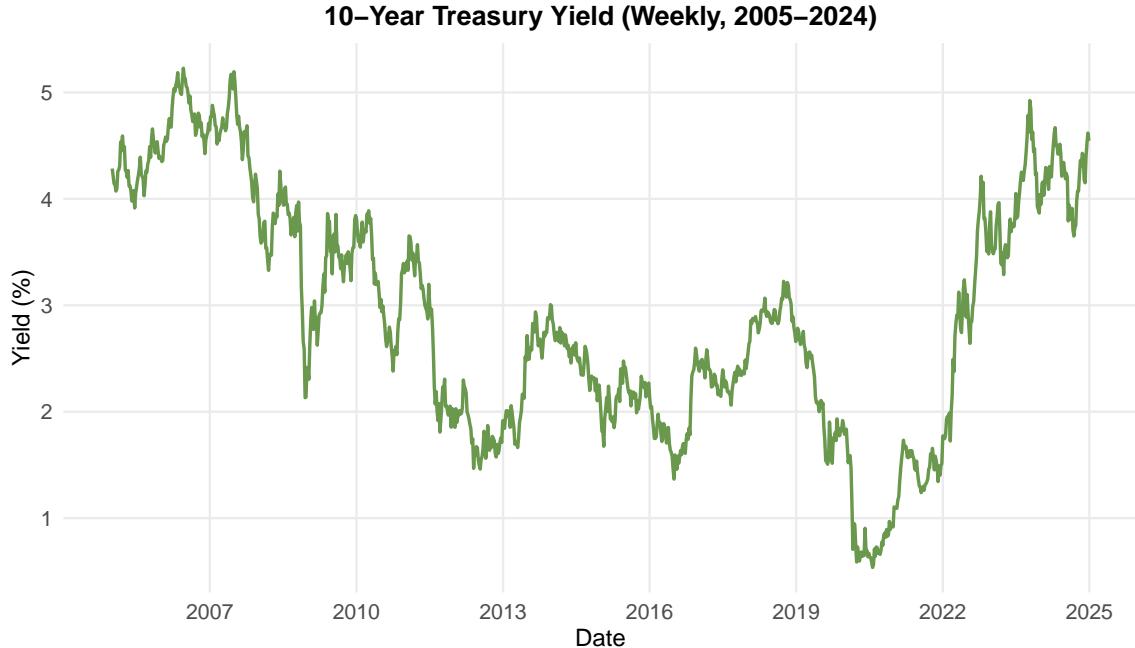
The graphical analysis reveals distinct patterns for each series. The **S&P 500** exhibits a strong upward trend over the period, with major declines during the 2008–2009 financial crisis (reaching a minimum around 683 points) and the COVID-19 pandemic in early 2020. The series shows non-stationary behavior with time-varying mean and variance, suggesting that log-differencing will be required to achieve stationarity.

The **VIX** displays a markedly different pattern: long periods of low volatility (typically 10–20) punctuated by extreme spikes. The most notable spikes occur during the 2008–2009 crisis and the COVID-19 pandemic (both reaching nearly 80 points), with smaller spikes around 2010, 2011, 2015–2016, and 2022. This mean-reverting behavior, without a clear deterministic trend, suggests that the VIX may be stationary in levels, though formal unit root tests are needed to confirm this.

The **10-year U.S. Treasury yield** shows significant cycles and regime changes. Starting around 4% in 2005, it peaks above 5% in 2007, then declines sharply during the financial crisis. After fluctuations between 2% and 4% through the 2010s, it reaches its lowest level (below 1%) in early 2020, followed by a rapid increase to near 5% by late 2023. The non-constant statistical properties suggest non-stationarity requiring differencing.

During major shocks (2008 and 2020), all three series move simultaneously: the S&P 500 falls, the VIX spikes, and yields decline—a "flight-to-quality" pattern. These interconnected dynamics justify multivariate analysis.

2.2 Simple and Partial Autocorrelograms

To complement the graphical analysis, we compute the autocorrelation function (ACF) and partial autocorrelation function (PACF) for each series. These functions help identify the order of autoregressive and moving average components and provide preliminary evidence on stationarity. A slowly decaying ACF suggests non-stationarity, while a sharp cutoff in the PACF indicates the order of the autoregressive component.

2.2.1 Autocorrelation Function (ACF)

The autocorrelation function measures the correlation between a time series and its lagged values. For a time series X_t, the autocorrelation at lag k is defined as:

$$\rho_k = \frac{\text{Cov}(X_t, X_{t-k})}{\sqrt{\text{Var}(X_t)\text{Var}(X_{t-k})}} = \frac{\text{Cov}(X_t, X_{t-k})}{\sigma^2}$$

where $\text{Cov}(X_t, X_{t-k})$ is the covariance between X_t and X_{t-k}, and σ^2 is the variance of the series. The sample autocorrelation at lag k is estimated as:

$$\hat{\rho}_k = \frac{\sum_{t=k+1}^{n}(X_t - \bar{X})(X_{t-k} - \bar{X})}{\sum_{t=1}^{n}(X_t - \bar{X})^2}$$

where $\bar{X}$ is the sample mean and n is the sample size.

2.2.2 Partial Autocorrelation Function (PACF)

The partial autocorrelation function measures the correlation between X_t and X_{t-k} after removing the effects of intermediate lags. The PACF at lag k, denoted ϕ_{kk}, is the coefficient of X_{t-k} in the regression:

$$X_t = \phi_{k1}X_{t-1} + \phi_{k2}X_{t-2} + \cdots + \phi_{kk}X_{t-k} + \varepsilon_t$$
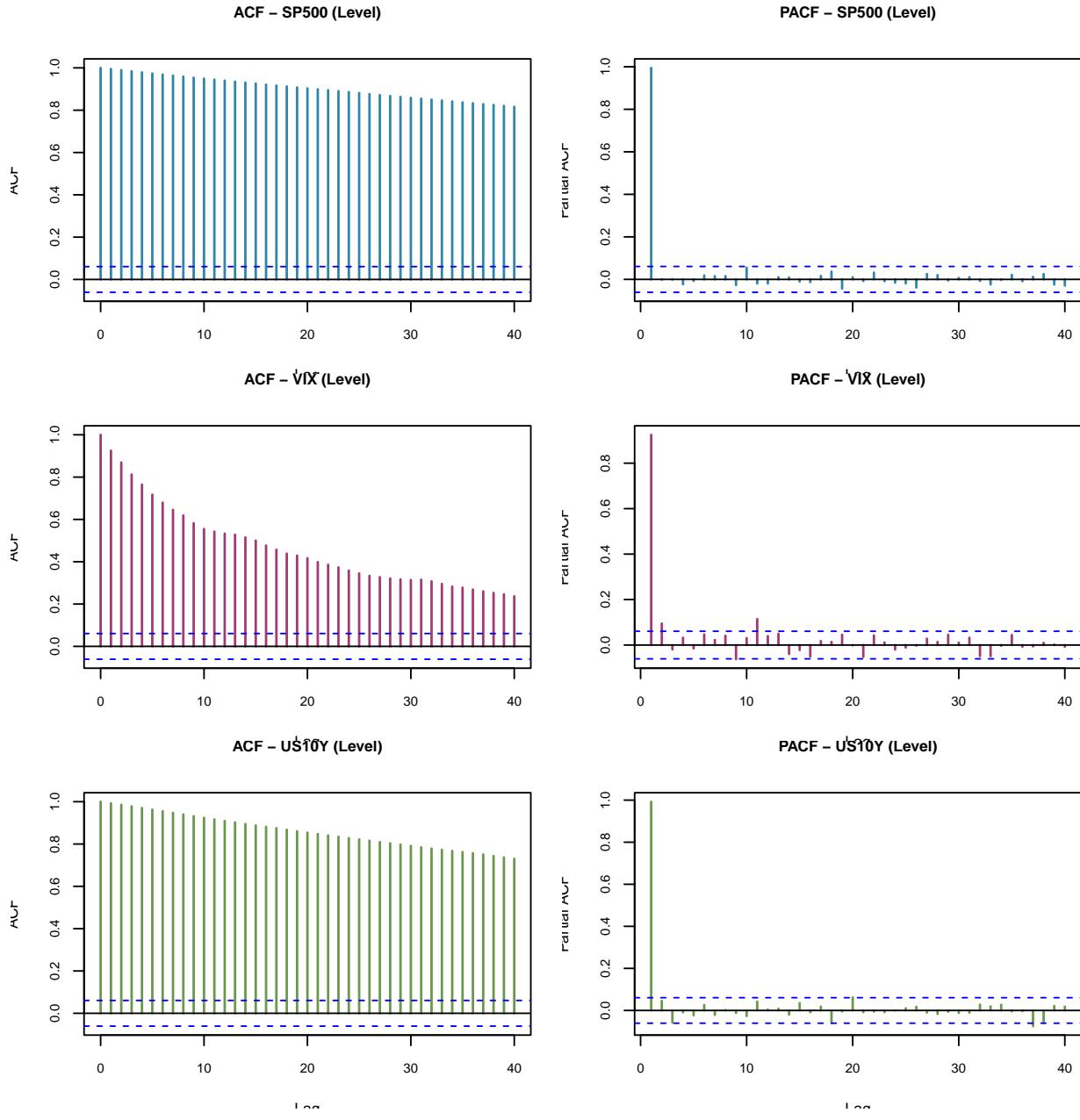
The PACF can be computed recursively using the Yule-Walker equations or estimated via least squares regression.

```r
# Compute ACF and PACF for each series in levels
# Set up plotting parameters: 3 rows, 2 columns (one for ACF, one for PACF)
par(mfrow = c(3, 2), mar = c(3.5, 3.5, 2.5, 1.5), cex.main = 0.9, cex.lab = 0.85, cex.axis = 0.8)

# S&P 500: ACF and PACF
acf(data_weekly_raw$SP500, main = "ACF - SP500 (Level)", lag.max = 40,
    col = "#2E86AB", lwd = 1.5)
pacf(data_weekly_raw$SP500, main = "PACF - SP500 (Level)", lag.max = 40,
     col = "#2E86AB", lwd = 1.5)

# VIX: ACF and PACF
acf(data_weekly_raw$VIX, main = "ACF - VIX (Level)", lag.max = 40,
    col = "#A23B72", lwd = 1.5)
pacf(data_weekly_raw$VIX, main = "PACF - VIX (Level)", lag.max = 40,
     col = "#A23B72", lwd = 1.5)

# US 10-Year Yield: ACF and PACF
acf(data_weekly_raw$US10Y, main = "ACF - US10Y (Level)", lag.max = 40,
    col = "#6A994E", lwd = 1.5)
pacf(data_weekly_raw$US10Y, main = "PACF - US10Y (Level)", lag.max = 40,
     col = "#6A994E", lwd = 1.5)
```

```r
# Reset plotting parameters
par(mfrow = c(1, 1), mar = c(5, 4, 4, 2) + 0.1)
```

2.3 Interpretation of ACF and PACF (Raw Series)

The autocorrelograms provide preliminary evidence on the stationarity properties of each series. For the **S&P 500**, the ACF shows a very slow decay with autocorrelations remaining extremely high (around 0.95–1.00) and statistically significant across all lags, indicating non-stationarity. The PACF displays a large spike at lag 1 followed by a sharp cutoff, consistent with a unit root process. This pattern suggests that the series is integrated of order 1 (I(1)) and requires log-differencing for stationarity.

The **VIX** exhibits a different pattern: the ACF decays more rapidly than the S&P 500, progressively returning toward zero, which is characteristic of a stationary series with high persistence. The PACF shows a large

12

spike at lag 1 followed by several smaller but significant spikes at various lags, indicating a more complex autoregressive structure than a simple random walk. This pattern suggests that the VIX is likely stationary (I(0)) but with strong persistence.

The **10-year U.S. Treasury yield** shows a pattern similar to the S&P 500: the ACF remains extremely high with very gradual decay, and the PACF displays a single large spike at lag 1 followed by a sharp cutoff. This structure is characteristic of a non-stationary series integrated of order 1 (I(1)), consistent with a random walk hypothesis for interest rates.

These visual interpretations provide preliminary guidance, but formal unit root tests (ADF, KPSS) are required to confirm the integration orders and determine the appropriate transformations for modeling.

---

3 Unit Root Testing Strategy and Stationarisation

3.1 Unit Root Testing Strategy

To determine the appropriate stationarisation method for each series, we apply a comprehensive unit root testing strategy using both the **Augmented Dickey–Fuller (ADF) test** and the **KPSS test**. These complementary tests provide a robust assessment of stationarity properties for each of the three weekly series.

3.1.1 Augmented Dickey–Fuller (ADF) Test

The Augmented Dickey–Fuller test considers three model specifications:

Model 3 (with constant and trend):

$$\Delta X_t = c + bt + \rho X_{t-1} + \sum_{j=1}^{p} \delta_j \Delta X_{t-j} + \varepsilon_t$$

Model 2 (with constant, no trend):

$$\Delta X_t = c + \rho X_{t-1} + \sum_{j=1}^{p} \delta_j \Delta X_{t-j} + \varepsilon_t$$

Model 1 (no constant, no trend):

$$\Delta X_t = \rho X_{t-1} + \sum_{j=1}^{p} \delta_j \Delta X_{t-j} + \varepsilon_t$$

where $\Delta X_t = X_t - X_{t-1}$ is the first difference of the series, c is a constant term, b is a trend coefficient, ρ is the coefficient of interest, δ_j are coefficients for the lagged differences, and ε_t is the error term.

The ADF test hypotheses are: - $H_0 : \rho = 0 \Rightarrow X_t$ non-stationary (unit root present) - $H_1 : \rho < 0 \Rightarrow X_t$ stationary (no unit root)

We apply a sequential testing strategy for each series, starting with Model 3 (the most general specification with both constant and trend). The number of lags p is initially set to 6, with automatic selection based on the Akaike Information Criterion (AIC) to account for potential autocorrelation in the error terms. This approach ensures that the residuals are white noise while allowing the data to determine the optimal lag length within the specified range.

Sequential Strategy: 1. **Model 3 (with constant and trend):** We first test for a unit root using Model 3. If the unit root hypothesis is not rejected, we test the significance of the trend coefficient b using the appropriate Dickey–Fuller critical values. 2. **Model 2 (with constant, no trend):** If the trend coefficient b is not significant (i.e., $H_0 : b = 0$ cannot be rejected), we proceed to Model 2. We test for a unit root again, and if not rejected, we test the significance of the constant coefficient c. 3. **Model 1 (no constant, no trend):** If the constant coefficient c is not significant (i.e., $H_0 : c = 0$ cannot be rejected), we proceed to Model 1, which is the most constrained specification.

The sequential strategy stops when we identify the appropriate model specification. This procedure ensures that we do not include unnecessary deterministic components (trend or constant) in our model, which would affect the critical values and the power of the test.

3.1.2 KPSS Test

The KPSS test complements the ADF test by testing the null hypothesis of level stationarity around a constant. The KPSS test hypotheses are: - $H_0 : X_t$ stationary (around a constant) - $H_1 : X_t$ non-stationary

We perform the KPSS test using Model 2 (with constant, no trend), which is the model closest to Model 1 when the sequential strategy stops at Model 1. The KPSS test is performed with `type = "mu"` (stationarity around a constant) and `lags = "long"` for automatic lag selection. When both tests agree, we have strong evidence for the integration order of the series. If the ADF test fails to reject the null hypothesis (unit root present) and the KPSS test rejects the null hypothesis (non-stationary), this provides robust evidence that the series is integrated of order 1 (I(1)) and requires differencing to achieve stationarity.

```r
# Load required packages for unit root testing
library(urca)     # For ADF and KPSS tests

# Extract the raw level series for testing
y_sp  <- data_weekly_raw$SP500
y_vix <- data_weekly_raw$VIX
y_us  <- data_weekly_raw$US10Y
```

3.2 Dickey-Fuller Test for the "SP500" Series

We begin the sequential testing strategy with Model 3 (the most general specification with both constant and trend).

3.2.1 Model 3:

```r
test_M3_SP500 <- ur.df(y = y_sp, type = "trend", lags = 6, selectlags = "AIC")
coefficients(test_M3_SP500@testreg)
```

```
##                 Estimate  Std. Error    t value   Pr(>|t|)
## (Intercept) -1.657821301 3.916068051 -0.4233382 0.67213663
## z.lag.1     -0.003045279 0.003556498 -0.8562578 0.39205375
## tt           0.026147081 0.015143118  1.7266643 0.08452694
## z.diff.lag  -0.073798757 0.031148876 -2.3692269 0.01800839
```

```r
test_M3_SP500@cval
```

```
##        1pct  5pct 10pct
## tau3 -3.96 -3.41 -3.12
## phi2  6.09  4.68  4.03
## phi3  8.27  6.25  5.34
```

The ADF test statistic is $t_{ADF} = -0.856$, while the 5% critical value is -3.41. Since $t_{ADF} > -3.41$, we cannot reject the null hypothesis of a unit root. The trend coefficient b is not significant ($|t_b| = 1.727 < 6.25$), so we proceed to Model 2.

3.2.2 Model 2:

```
test_M2_SP500 <- ur.df(y = y_sp, type = "drift", lags = 6, selectlags = "AIC")
coefficients(test_M2_SP500@testreg)
```

```
##                  Estimate Std. Error    t value    Pr(>|t|)
## (Intercept) -1.145668529 3.90855816 -0.2931179 0.76949082
## z.lag.1      0.002554881 0.00146065  1.7491399 0.08056352
## z.diff.lag  -0.076458087 0.03114057 -2.4552565 0.01424217
```

```
test_M2_SP500@cval
```

```
##        1pct  5pct 10pct
## tau2 -3.43 -2.86 -2.57
## phi1  6.43  4.59  3.78
```

The ADF test statistic is $t_{ADF} = 1.749$, while the 5% critical value is -2.86. Since $t_{ADF} > -2.86$, we cannot reject the null hypothesis of a unit root. The constant coefficient c is not significant ($|t_c| = 0.293 < 4.59$), so we proceed to Model 1.

3.2.3 Model 1:

```
test_M1_SP500 <- ur.df(y = y_sp, type = "none", lags = 6, selectlags = "AIC")
coefficients(test_M1_SP500@testreg)
```

```
##               Estimate   Std. Error   t value    Pr(>|t|)
## z.lag.1     0.002178495 0.0006958525  3.130684 0.001792876
## z.diff.lag -0.076009334 0.0310891762 -2.444881 0.014655880
```

```
test_M1_SP500@cval
```

```
##        1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
```

The ADF test statistic is $t_{ADF} = 3.131$, while the 5% critical value is -1.95. Since $t_{ADF} > -1.95$, we cannot reject the null hypothesis of a unit root. The sequential strategy stops here and we conclude that the SP500 series is integrated of order 1 (I(1)), corresponding to a DS (Difference Stationary) process without drift.

3.3 KPSS Test for the "SP500" Series

```
kpssM2_SP500 <- ur.kpss(y = y_sp, type = "mu", lags = "long")
show(kpssM2_SP500)
```

```
##
## #######################################
## # KPSS Unit Root / Cointegration Test #
## #######################################
##
## The value of the test statistic is: 4.2351
```

```
kpssM2_SP500@cval
```

```
##                10pct  5pct 2.5pct  1pct
## critical values 0.347 0.463  0.574 0.739
```

The KPSS test statistic is $LM_{obs} = 4.235$, while the 5% critical value is 0.463. Since $LM_{obs} > 0.463$, we reject the null hypothesis of stationarity. The process is I(1).

Conclusion for SP500:

Both the ADF and KPSS tests confirm that the SP500 series is integrated of order 1 (I(1)) and requires differencing to achieve stationarity.

3.4 Dickey-Fuller Test for the "VIX" Series

We begin the sequential testing strategy with Model 3 (the most general specification with both constant and trend).

3.4.1 Model 3:

```
test_M3_VIX <- ur.df(y = y_vix, type = "trend", lags = 6, selectlags = "AIC")
coefficients(test_M3_VIX@testreg)
```

```
##                  Estimate    Std. Error    t value      Pr(>|t|)
## (Intercept)   1.4110124483 0.3181714418   4.4347552 1.020443e-05
## z.lag.1      -0.0686928441 0.0120188706  -5.7154159 1.431136e-08
## tt           -0.0001881039 0.0003460132  -0.5436321 5.868120e-01
## z.diff.lag   -0.0956272434 0.0309587048  -3.0888645 2.062912e-03
```

```
test_M3_VIX@cval
```

```
##        1pct  5pct 10pct
## tau3 -3.96 -3.41 -3.12
## phi2  6.09  4.68  4.03
## phi3  8.27  6.25  5.34
```

The ADF test statistic is $t_{ADF} = -5.715$, while the 5% critical value is -3.41. Since $t_{ADF} < -3.41$, we reject the null hypothesis of a unit root. The sequential strategy stops here and we conclude that the VIX series is stationary (I(0)).

16

3.5 KPSS Test for the "VIX" Series

```
kpssM2_VIX <- ur.kpss(y = y_vix, type = "mu", lags = "long")
show(kpssM2_VIX)
```

```
##
## #######################################
## # KPSS Unit Root / Cointegration Test #
## #######################################
##
## The value of the test statistic is: 0.2436
```

```
kpssM2_VIX@cval
```

```
##                10pct  5pct 2.5pct  1pct
## critical values 0.347 0.463  0.574 0.739
```

The KPSS test statistic is $LM_{obs} = 0.244$, while the 5% critical value is 0.463. Since $LM_{obs} < 0.463$, we cannot reject the null hypothesis of stationarity. The process is stationary (I(0)).

Conclusion for VIX:

Both the ADF and KPSS tests confirm that the VIX series is stationary (I(0)) and does not require differencing.

3.6 Dickey-Fuller Test for the "US10Y" Series

We begin the sequential testing strategy with Model 3 (the most general specification with both constant and trend).

3.6.1 Model 3:

```
test_M3_US10Y <- ur.df(y = y_us, type = "trend", lags = 6, selectlags = "AIC")
coefficients(test_M3_US10Y@testreg)
```

```
##                 Estimate    Std. Error   t value     Pr(>|t|)
## (Intercept)  1.167707e-02 1.538500e-02  0.7589906 0.448031832
## z.lag.1     -5.292814e-03 3.643825e-03 -1.4525436 0.146655070
## tt           7.407851e-06 1.360228e-05  0.5446035 0.586144133
## z.diff.lag1 -6.260368e-02 3.114773e-02 -2.0098956 0.044702634
## z.diff.lag2  8.621119e-02 3.120194e-02  2.7630076 0.005829166
## z.diff.lag3  3.125317e-02 3.124069e-02  1.0003995 0.317352159
## z.diff.lag4  6.064469e-02 3.122101e-02  1.9424318 0.052357133
```

```
test_M3_US10Y@cval
```

```
##        1pct  5pct 10pct
## tau3 -3.96 -3.41 -3.12
## phi2  6.09  4.68  4.03
## phi3  8.27  6.25  5.34
```

The ADF test statistic is $t_{ADF} = -1.453$, while the 5% critical value is -3.41. Since $t_{ADF} > -3.41$, we cannot reject the null hypothesis of a unit root. The trend coefficient b is not significant ($|t_b| = 0.545 < 6.25$), so we proceed to Model 2.

3.6.2 Model 2:

```
test_M2_US10Y <- ur.df(y = y_us, type = "drift", lags = 6, selectlags = "AIC")
coefficients(test_M2_US10Y@testreg)
```

```
##                Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept)  0.017890563 0.010317738  1.733962 0.083223798
## z.lag.1     -0.006097084 0.003330009 -1.830951 0.067396298
## z.diff.lag1 -0.061458895 0.031066113 -1.978326 0.048157815
## z.diff.lag2  0.087375664 0.031117969  2.807885 0.005080899
## z.diff.lag3  0.032444148 0.031153425  1.041431 0.297919621
## z.diff.lag4  0.061682005 0.031152223  1.980019 0.047966911
```

```
test_M2_US10Y@cval
```

```
##       1pct  5pct 10pct
## tau2 -3.43 -2.86 -2.57
## phi1  6.43  4.59  3.78
```

The ADF test statistic is $t_{ADF} = -1.831$, while the 5% critical value is -2.86. Since $t_{ADF} > -2.86$, we cannot reject the null hypothesis of a unit root. The constant coefficient c is not significant ($|t_c| = 1.734 < 4.59$), so we proceed to Model 1.

3.6.3 Model 1:

```
test_M1_US10Y <- ur.df(y = y_us, type = "none", lags = 6, selectlags = "AIC")
coefficients(test_M1_US10Y@testreg)
```

```
##                 Estimate  Std. Error    t value     Pr(>|t|)
## z.lag.1     -0.0007092396 0.001198655 -0.5916962 0.554183646
## z.diff.lag1 -0.0642012215 0.031055979 -2.0672741 0.038957202
## z.diff.lag2  0.0847498717 0.031111302  2.7240863 0.006556873
## z.diff.lag3  0.0294084332 0.031134418  0.9445635 0.345102874
## z.diff.lag4  0.0588593012 0.031139891  1.8901576 0.059016893
```

```
test_M1_US10Y@cval
```

```
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
```

The ADF test statistic is $t_{ADF} = -0.592$, while the 5% critical value is -1.95. Since $t_{ADF} > -1.95$, we cannot reject the null hypothesis of a unit root. The sequential strategy stops here and we conclude that the US10Y series is integrated of order 1 (I(1)), corresponding to a DS (Difference Stationary) process without drift.

3.7 KPSS Test for the "US10Y" Series

```
kpssM2_US10Y <- ur.kpss(y = y_us, type = "mu", lags = "long")
show(kpssM2_US10Y)
```

```
##
## #######################################
## # KPSS Unit Root / Cointegration Test #
## #######################################
##
## The value of the test statistic is: 1.4541
```

```
kpssM2_US10Y@cval
```

```
##                10pct  5pct 2.5pct  1pct
## critical values 0.347 0.463  0.574 0.739
```

The KPSS test statistic is $LM_{obs} = 1.454$, while the 5% critical value is 0.463. Since $LM_{obs} > 0.463$, we reject the null hypothesis of stationarity. The process is I(1).

Conclusion for US10Y:

Both the ADF and KPSS tests confirm that the US10Y series is integrated of order 1 (I(1)) and requires differencing to achieve stationarity.

---

4 ARMA Model Identification

We now identify an appropriate ARMA model for one of the series. We select the **S&P 500** for this analysis, as it is the most economically central variable and provides clear interpretability for the forecasting exercise.

4.1 Construction of the Stationary Series

The sequential unit root testing strategy conducted in Question 3 concluded that the S&P 500 index level is integrated of order 1 (I(1)) and corresponds to a DS (Difference Stationary) process without drift. To estimate a valid ARMA model, we must transform the series into a stationary process. We transform the series into the **weekly log-return**, which is the first difference of the logarithm of the index level:

$$r_t^{SP500} = \log(SP500_t) - \log(SP500_{t-1}) = \Delta \log(SP500_t).$$

This transformation has two advantages: (1) it achieves stationarity by removing the unit root, as confirmed by our unit root tests, and (2) it provides an economically interpretable measure of weekly returns that is approximately equal to the percentage change in the index level.

```
# Load required packages for data manipulation
library(tidyr)

# Compute weekly log-returns for the S&P 500
```

```r
# The log-return is the first difference of the log of the index level
sp500_ret <- data_weekly_raw |>
  transmute(
    date,
    return_SP500 = log(SP500) - lag(log(SP500))
  ) |>
  tidyr::drop_na()  # Remove the first observation (NA due to lag)


# Display first few observations
knitr::kable(head(sp500_ret, 10),
             caption = "First few observations of S\\&P 500 weekly returns",
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 3: First few observations of S&P 500 weekly returns

date	return_SP500
2005-01-08	-0.0014088
2005-01-15	-0.0141561
2005-01-22	0.0029839
2005-01-29	0.0266779
2005-02-05	0.0018851
2005-02-12	-0.0030829
2005-02-19	0.0081063
2005-02-26	0.0088351
2005-03-05	-0.0181989
2005-03-12	-0.0087290

```r
# Display summary statistics for the return series
summary_stats <- summary(sp500_ret$return_SP500)
summary_df <- data.frame(
  Statistic = names(summary_stats),
  Value = as.numeric(summary_stats),
  row.names = NULL
)
knitr::kable(summary_df,
             caption = "Summary statistics for S\\&P 500 weekly returns",
             row.names = FALSE, digits = 6,
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 4: Summary statistics for S&P 500 weekly returns
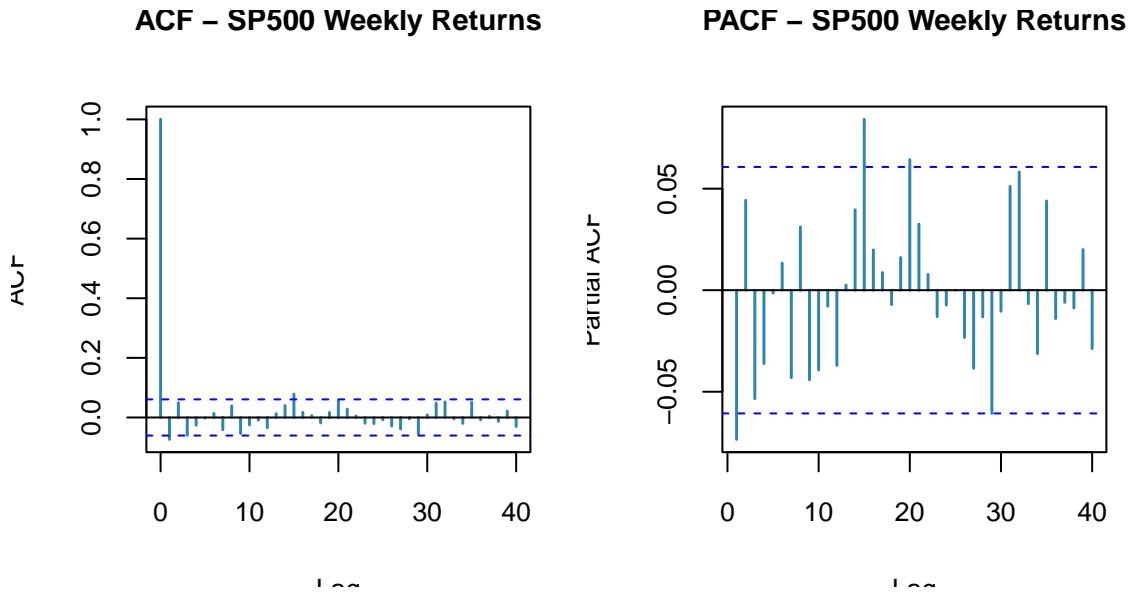
Statistic	Value
Min.	-0.200838
1st Qu.	-0.009422
Median	0.002878
Mean	0.001539
3rd Qu.	0.014368

Max.	0.114237

4.2 ACF and PACF of the Return Series

```
par(mfrow = c(1, 2), mar = c(3.5, 3.5, 2.5, 1.5), cex.main = 0.9, cex.lab = 0.85, cex.axis = 0.8)

acf(sp500_ret$return_SP500, main = "ACF - SP500 Weekly Returns", lag.max = 40,
    col = "#2E86AB", lwd = 1.5)
pacf(sp500_ret$return_SP500, main = "PACF - SP500 Weekly Returns", lag.max = 40,
     col = "#2E86AB", lwd = 1.5)
```

ACF – SP500 Weekly Returns   **PACF – SP500 Weekly Returns**


```
par(mfrow = c(1, 1), mar = c(5, 4, 4, 2) + 0.1)
```

Interpretation:

The ACF exhibits no statistically significant autocorrelation beyond lag 0, with all coefficients lying well inside the 95% confidence bands ($\pm$ 0.06). This pattern indicates that the return series behaves almost like white noise, consistent with the weak-form efficient-market hypothesis. The small negative autocorrelation at lag 1 is typical for equity returns and most likely reflects market microstructure effects—such as bid–ask bounce or short-term return reversal—rather than genuine predictability.

In contrast, the PACF shows a single significant negative spike at lag 1, followed by coefficients that are indistinguishable from zero. This sharp cut-off is the canonical signature of a stationary AR(1) process. The dependence structure of weekly S&P 500 returns is therefore extremely weak and very short-lived: past returns contain virtually no information about future returns.

Economically, the slight lag-1 reversal reflects transitory price adjustments rather than systematic forecasting opportunities. Statistically, the correlograms justify considering an AR(1) structure for the conditional mean. In the next step, we use information criteria (AIC, BIC) to formally compare ARMA(p,q) specifications and select the most parsimonious and statistically adequate model.

4.3 ARMA Model Identification and Selection

An ARMA(p, q) model for a stationary time series r_t is defined as:

$$r_t = c + \sum_{i=1}^{p} \phi_i r_{t-i} + \varepsilon_t + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j}$$

where c is a constant (mean), ϕ_i are the autoregressive coefficients, θ_j are the moving average coefficients, and ε_t is a white noise process with $\varepsilon_t \sim \mathrm{WN}(0, \sigma^2)$.

For the ARMA(1, 0) model (AR(1)), the specification simplifies to:

$$r_t = c + \phi_1 r_{t-1} + \varepsilon_t$$

To identify the optimal ARMA model for the S&P 500 weekly returns, we perform a systematic search over a grid of possible AR and MA orders. We consider models with $p, q \in \{0, 1, 2, 3, 4\}$, where p is the order of the autoregressive component and q is the order of the moving average component. For each model specification, we compute the **Akaike Information Criterion (AIC)** and the **Bayesian Information Criterion (BIC)**:

$$\mathrm{AIC} = -2\log(L) + 2k$$

$$\mathrm{BIC} = -2\log(L) + k\log(n)$$

where L is the maximum likelihood, k is the number of parameters, and n is the number of observations. The AIC tends to favor more complex models, while the BIC penalizes complexity more heavily and favors parsimony.

```r
# Load the forecast package for ARIMA model estimation
library(forecast)

# Create a grid of all possible ARMA(p,q) combinations
# We consider p and q from 0 to 4
# Exclude the white noise model (ARMA(0,0))
arma_models <- expand.grid(ar = 0:4, ma = 0:4) |>
  filter(!(ar == 0 & ma == 0))

# Initialize vectors to store information criteria
aic_values <- numeric(nrow(arma_models))
bic_values <- numeric(nrow(arma_models))
model_list <- list()

# Estimate each ARMA model and compute information criteria
for (i in 1:nrow(arma_models)) {
  tryCatch({
    # Estimate ARMA(p,q) model with intercept (include.mean = TRUE)
    # Note: d=0 because we are working with already differenced returns
    model <- Arima(sp500_ret$return_SP500,
                  order = c(arma_models$ar[i], 0, arma_models$ma[i]),
                  include.mean = TRUE)
    # Store AIC and BIC values
    aic_values[i] <- AIC(model)
    bic_values[i] <- BIC(model)
    model_list[[i]] <- model
  }, error = function(e) {
```

```
    # If estimation fails, store NA values
    aic_values[i] <- NA
    bic_values[i] <- NA
  })
}

# Add information criteria to the data frame
arma_models$AIC <- aic_values
arma_models$BIC <- bic_values

# Find the indices of the best models according to each criterion
best_aic_idx <- which.min(aic_values)
best_bic_idx <- which.min(bic_values)

# Display the best models and top 5 rankings
top5_aic <- arma_models[order(arma_models$AIC)[1:5], ]
top5_bic <- arma_models[order(arma_models$BIC)[1:5], ]

# Display top 5 models by AIC
knitr::kable(top5_aic,
             caption = "Top 5 models by AIC",
             row.names = FALSE, digits = 2,
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 5: Top 5 models by AIC

ar	ma	AIC	BIC
0	3	-4779.91	-4755.17
1	1	-4779.51	-4759.71
3	0	-4779.21	-4754.46
0	4	-4778.82	-4749.13
1	3	-4778.76	-4749.06

```
# Display top 5 models by BIC
knitr::kable(top5_bic,
             caption = "Top 5 models by BIC",
             row.names = FALSE, digits = 2,
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 6: Top 5 models by BIC

ar	ma	AIC	BIC
1	0	-4778.19	-4763.34
0	1	-4777.74	-4762.89
1	1	-4779.51	-4759.71
2	0	-4778.23	-4758.43
0	2	-4777.62	-4757.83

4.4 Estimation of the Selected ARMA Model

Based on the information criteria results, we observe that: - The **best model by AIC** is ARMA(0, 3) with AIC = -4779.91 - The **best model by BIC** is ARMA(1, 0) with BIC = -4763.34

We select the model that minimizes the BIC (ARMA(1, 0)) for the following reasons: 1. **Parsimony**: The BIC favors simpler models and reduces the risk of overfitting, which is particularly important for forecasting. 2. **Small AIC gap**: The AIC of the BIC-optimal model (ARMA(1, 0)) is -4778.19, which is only 1.72 points higher than the best AIC model (ARMA(0, 3) with AIC = -4779.91). This small difference suggests that the simpler ARMA(1, 0) model provides a good balance between fit quality and model complexity. 3. **Interpretability**: The AR(1) model is more interpretable and aligns with the visual inspection of the ACF and PACF correlograms, which suggested an AR(1) specification.

We then estimate this model and examine its properties, including coefficient estimates, standard errors, and significance tests.

```r
# The best model by BIC is ARMA(1, 0)
selected_ar <- 1
selected_ma <- 0

# Estimate the selected ARMA model
arma_final <- Arima(sp500_ret$return_SP500,
                    order = c(selected_ar, 0, selected_ma),
                    include.mean = TRUE)

# Display model summary
summary(arma_final)
```

```
## Series: sp500_ret$return_SP500
## ARIMA(1,0,0) with non-zero mean
##
## Coefficients:
##           ar1     mean
##       -0.0734  0.0015
## s.e.   0.0309  0.0007
##
## sigma^2 = 0.0005974:  log likelihood = 2392.09
## AIC=-4778.19   AICc=-4778.17   BIC=-4763.34
##
## Training set error measures:
##                       ME       RMSE        MAE      MPE     MAPE      MASE
## Training set -1.772499e-07 0.02441846 0.01674675 96.63116 120.5714 0.6738112
##                    ACF1
## Training set 0.003181219
```

```r
coef_table <- data.frame(
  Coefficient = names(coef(arma_final)),
  Estimate = as.numeric(coef(arma_final)),
  StdError = as.numeric(sqrt(diag(arma_final$var.coef))),
  tValue = as.numeric(coef(arma_final) / sqrt(diag(arma_final$var.coef))),
  pValue = 2 * (1 - pnorm(abs(coef(arma_final) / sqrt(diag(arma_final$var.coef)))))
)
# Add significance stars to p-values
coef_table$Significance <- ifelse(coef_table$pValue < 0.001, "***",
```

```
                            ifelse(coef_table$pValue < 0.01, "**",
                                ifelse(coef_table$pValue < 0.05, "*",
                                    ifelse(coef_table$pValue < 0.1, ".", ""))))

knitr::kable(coef_table, row.names = FALSE, digits = 4,
            booktabs = TRUE, linesep = "",
            caption = "ARMA Model Coefficients",
            align = c("l", "r", "r", "r", "r", "c")) |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE) |>
  footnote(general = "Significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1",
            general_title = "Note:",
            footnote_as_chunk = TRUE)
```

Table 7: ARMA Model Coefficients

Coefficient	Estimate	StdError	tValue	pValue	Significance
ar1	-0.0734	0.0309	-2.3784	0.0174	*
intercept	0.0015	0.0007	2.1843	0.0289	*

Note: Significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Commentary:

The estimated ARMA(1, 0) model shows that both coefficients are statistically significant at the 5% level. The autoregressive coefficient $\phi_1 = -0.0734$ is negative and significant (p-value $= 0.0174$), indicating a mean-reverting behavior in the weekly returns. The negative sign suggests that a positive return in the previous week tends to be followed by a slightly negative return, and vice versa, which is consistent with the presence of mean reversion in financial returns. The intercept (mean) is positive ($\mu = 0.0015$) and significant (p-value $= 0.0289$), indicating a small but statistically significant positive average weekly return over the sample period. The model fit statistics show an AIC of -4778.19 and a BIC of -4763.34, which are consistent with the values reported in the model selection procedure. The residual variance ($\sigma^2 = 0.0005974$) and the low ACF1 value (0.0032) suggest that the model captures the temporal dependencies well, leaving residuals that are close to white noise.

4.5   Model Validation

Before using the model for forecasting, we must verify that it satisfies the validity conditions. We check three key aspects: (1) the residuals should be white noise (no remaining autocorrelation), (2) the model should be stable (all roots of the AR and MA polynomials should lie outside the unit circle), and (3) ideally, the residuals should be normally distributed, though this is not strictly required for ARMA validity.

4.5.1   Diagnostic Tests

Ljung-Box Test:

The Ljung-Box test is used to test the null hypothesis that the residuals are white noise (no autocorrelation). The test statistic is:

$$Q_{LB} = n(n+2) \sum_{k=1}^{h} \frac{\hat{\rho}_k^2}{n-k}$$

where n is the sample size, h is the number of lags tested, and $\hat{\rho}_k$ is the sample autocorrelation at lag k. Under the null hypothesis of white noise, $Q_{LB} \sim \chi^2(h)$.

Shapiro-Wilk Test:

The Shapiro-Wilk test is used to test the null hypothesis that the residuals are normally distributed. The test statistic is:

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

where $x_{(i)}$ are the ordered sample values, a_i are constants derived from the means, variances, and covariances of the order statistics of a sample from a normal distribution, and $\bar{x}$ is the sample mean. Under the null hypothesis of normality, W follows a specific distribution tabulated by Shapiro and Wilk.

```r
# Extract residuals from the estimated model
residuals_arma <- residuals(arma_final)

# --- Test for residual autocorrelation ---
# Ljung-Box test: tests the null hypothesis that residuals are white noise
# We use lag = 10 to test for autocorrelation up to 10 periods
ljung_box <- Box.test(residuals_arma, lag = 10, type = "Ljung-Box")

# --- Test for normality of residuals ---
# Shapiro-Wilk test: tests the null hypothesis that residuals are normally distributed
# We limit to 5000 observations due to computational constraints
shapiro_test <- shapiro.test(residuals_arma[1:5000])

# Display test results in a single output
validation_tests <- data.frame(
  Test = c("Ljung-Box", "Shapiro-Wilk"),
  Statistic = c(round(ljung_box$statistic, 4), round(shapiro_test$statistic, 4)),
  P_Value = c(round(ljung_box$p.value, 4), format(shapiro_test$p.value, scientific = TRUE)),
  Interpretation = c(
    ifelse(ljung_box$p.value > 0.05,
           "Residuals are white noise (model valid)",
           "Residuals show autocorrelation (model may need adjustment)"),
    ifelse(shapiro_test$p.value > 0.05,
                         "Residuals are normally distributed",
           "Residuals are not normally distributed")
  )
)
knitr::kable(validation_tests, row.names = FALSE,
             booktabs = TRUE, linesep = "",
             caption = "Residual Diagnostic Tests") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE) |>
  column_spec(4, italic = TRUE)
```
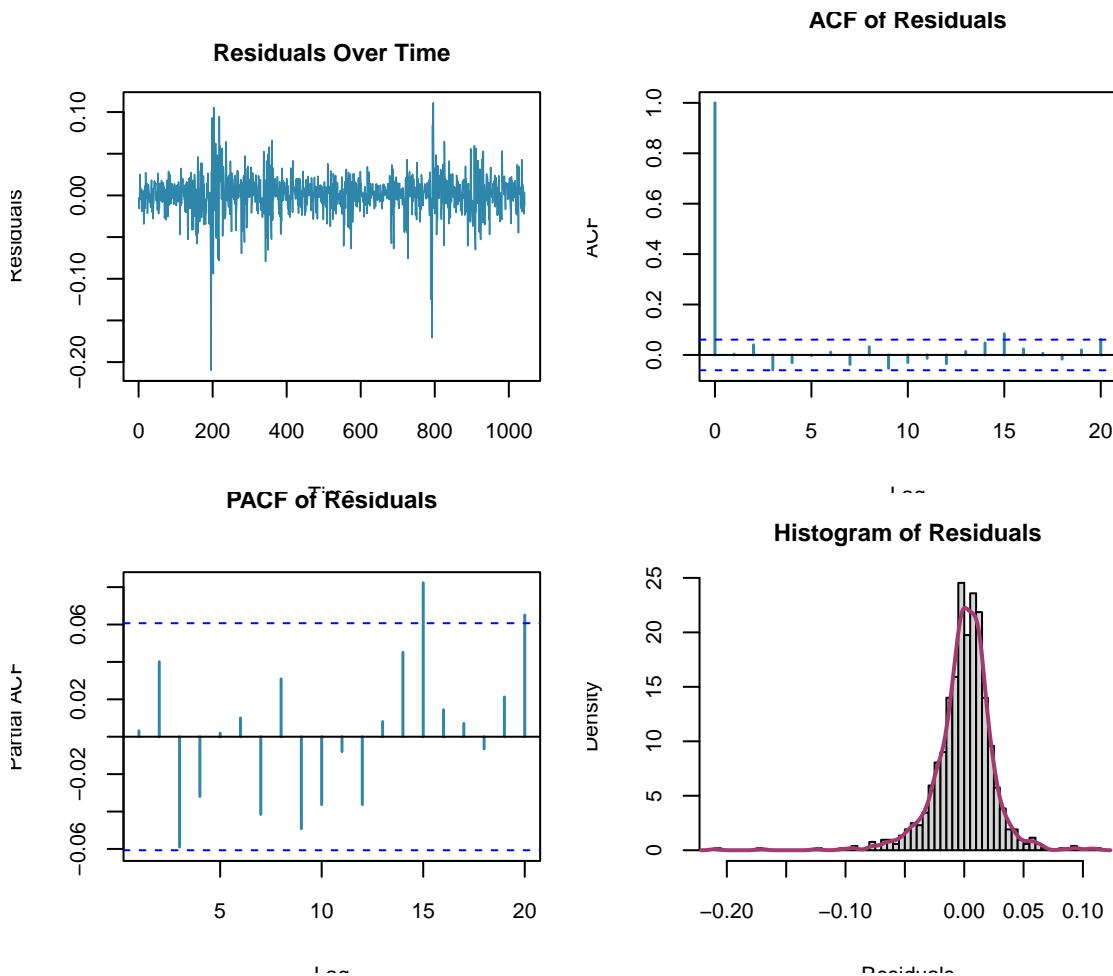
Table 8: Residual Diagnostic Tests

Test	Statistic	P_Value	Interpretation
Ljung-Box	12.9031	0.2291	*Residuals are white noise (model valid)*

26

| Shapiro-Wilk | 0.9078 | 1.282205e-24 | *Residuals are not normally distributed* |

```r
# --- Graphical diagnostics ---
# Create diagnostic plots: residuals over time, ACF, PACF, and histogram
par(mfrow = c(2, 2), mar = c(3.5, 3.5, 2.5, 1.5), cex.main = 0.9, cex.lab = 0.85, cex.axis = 0.8)
plot(residuals_arma, main = "Residuals Over Time", ylab = "Residuals", type = "l",
     col = "#2E86AB", lwd = 0.8)
acf(residuals_arma, main = "ACF of Residuals", lag.max = 20, col = "#2E86AB", lwd = 1.5)
pacf(residuals_arma, main = "PACF of Residuals", lag.max = 20, col = "#2E86AB", lwd = 1.5)
hist(residuals_arma, main = "Histogram of Residuals", xlab = "Residuals", breaks = 50,
     freq = FALSE, col = "lightgray", border = "black")
lines(density(residuals_arma), col = "#A23B72", lwd = 2)
```



```r
par(mfrow = c(1, 1), mar = c(5, 4, 4, 2) + 0.1)

# --- Check model stability ---
# For ARMA(1,0), we check that the AR polynomial root lies outside the unit circle
# This ensures the model is stationary
ar_roots <- polyroot(c(1, -coef(arma_final)["ar1"]))
  cat("\nAR polynomial roots (should be outside unit circle):\n")
```

```
##
```

```
## AR polynomial roots (should be outside unit circle):

  cat("  Modulus:", round(Mod(ar_roots), 4), "\n")


##   Modulus: 13.6195

  cat("  All roots outside unit circle:", all(Mod(ar_roots) > 1), "\n")


##   All roots outside unit circle: TRUE
```

Commentary:

The diagnostic tests and graphical analysis provide evidence that the ARMA(1, 0) model satisfies the validity conditions:

1. **Residual autocorrelation (Ljung-Box test)**: The test fails to reject the null hypothesis that residuals are white noise, indicating that the model has successfully captured the temporal dependencies in the data. The ACF and PACF plots of residuals confirm this result, as all autocorrelations and partial autocorrelations fall within the confidence intervals, with only minor exceptions at higher lags (e.g., lag 15 and 20 in the PACF).

2. **Model stability**: The AR polynomial root has a modulus of 13.62, which is well outside the unit circle. This confirms that the model is stationary and satisfies the stability condition.

3. **Normality of residuals (Shapiro-Wilk test)**: The test rejects the null hypothesis of normality, which is common for financial return series. The histogram of residuals shows a distribution that is approximately centered around zero but exhibits heavy tails, particularly on the negative side, consistent with the leptokurtic nature of financial returns. While normality is not strictly required for ARMA model validity, this finding suggests that the residuals may contain some non-linear dependencies that could be addressed with more sophisticated models (e.g., GARCH) if needed for forecasting.

4. **Residuals over time**: The time series plot of residuals shows that they fluctuate around zero with relatively constant variance, except for a few notable spikes (around time 200 and 800), which correspond to periods of high market volatility. Overall, the residuals appear to be well-behaved and do not exhibit systematic patterns.

In conclusion, the model validation confirms that the ARMA(1, 0) specification is appropriate for the S&P 500 weekly returns, with residuals that are effectively white noise and a stable model structure.

––––––––––––––––––––––––––––

5 Forecasting

We compute forecasts of the S&P 500 weekly log-returns over horizons of 1 to 3 periods using the estimated ARMA(1,0) model. We then convert these forecasts back to the original index level.

5.1 Forecasts of the Stationarised Series (Log-Returns)

We now use the validated ARMA(1,0) model to generate forecasts of the S&P 500 weekly log-returns over horizons of 1 to 3 periods. For an ARMA(1,0) model with intercept, the forecast at horizon h is computed recursively:

$$\hat{r}_{T+h|T} = \begin{cases} \mu + \phi_1 r_T & \text{if } h = 1 \\ \mu + \phi_1 \hat{r}_{T+h-1|T} & \text{if } h > 1 \end{cases}$$

where μ is the intercept and ϕ_1 is the AR(1) coefficient. As the forecast horizon increases, the forecasts converge toward the unconditional mean return μ, reflecting the mean-reverting nature of the AR(1) process.

```
# Load the forecast package for forecasting functions

# Generate forecasts for log-returns at horizons 1, 2, and 3
# We compute 80% and 95% prediction intervals to assess forecast uncertainty
forecast_returns <- forecast(arma_final, h = 3, level = c(80, 95))

forecast_table <- data.frame(
  Horizon = 1:3,
  Point_Forecast = as.numeric(forecast_returns$mean),
  Lo_80 = as.numeric(forecast_returns$lower[,1]),
  Hi_80 = as.numeric(forecast_returns$upper[,1]),
  Lo_95 = as.numeric(forecast_returns$lower[,2]),
  Hi_95 = as.numeric(forecast_returns$upper[,2])
)

# Display forecast table
knitr::kable(forecast_table,
             col.names = c("Horizon", "Point Forecast", "Lo 80%", "Hi 80%", "Lo 95%", "Hi 95%"),
             row.names = FALSE,
             digits = 6,
             booktabs = TRUE, linesep = "",
             caption = "Forecasts of S\\&P 500 Weekly Log-Returns") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 9: Forecasts of S&P 500 Weekly Log-Returns

Horizon	Point Forecast	Lo 80%	Hi 80%	Lo 95%	Hi 95%
1	0.002443	-0.028880	0.033767	-0.045462	0.050349
2	0.001474	-0.029934	0.032882	-0.046560	0.049508
3	0.001545	-0.029863	0.032953	-0.046490	0.049580

```
# Display additional information
cat("\nForecast method: ARMA(1,0) recursive forecasting\n")
```

```
##
## Forecast method: ARMA(1,0) recursive forecasting
```

```r
cat("Last observed return:", round(tail(sp500_ret$return_SP500, 1), 6), "\n")
```

Last observed return: -0.01076

```r
cat("Last observation date:", format(tail(sp500_ret$date, 1), "%Y-%m-%d"), "\n")
```

Last observation date: 2024-12-28

5.2 Forecasts of the Original Series (S&P 500 Index Level)

While forecasts of log-returns are useful for understanding the expected percentage changes, investors and analysts are typically more interested in forecasts of the actual index level. To obtain forecasts of the original S&P 500 index level, we reverse the log-transformation by applying the exponential function to the cumulative returns:

$$\widehat{SP500}_{T+h|T} = SP500_T \times \exp\left(\sum_{i=1}^{h} \hat{r}_{T+i|T}\right)$$

This transformation accounts for the cumulative effect of returns over the forecast horizon. The prediction intervals for the level forecasts must account for the accumulation of variance over time, as uncertainty increases with the forecast horizon.

```r
# Extract the last observed S&P 500 index level and date
last_sp500 <- tail(data_weekly_raw$SP500, 1)
last_date <- tail(data_weekly_raw$date, 1)

# Compute cumulative returns for each horizon
# The cumulative return is the sum of forecasted returns up to horizon h
cumulative_returns <- cumsum(forecast_returns$mean)

# Apply exponential transformation to obtain level forecasts
# This reverses the log-transformation: level = last_level × exp(cumulative_returns)
forecast_levels <- last_sp500 * exp(cumulative_returns)

# Generate forecast dates (assuming weekly frequency, 7 days per week)
forecast_dates <- seq(from = last_date + 7, by = 7, length.out = 3)

# Compute prediction intervals for levels
# We use a log-normal approximation: if log(X) ~ N(mu, sigma^2), then X ~ LogNormal
# For cumulative returns, variance accumulates: Var(sum of h returns) = h × sigma^2
sigma2 <- arma_final$sigma2
var_cumulative <- sigma2 * (1:3)

# Function to compute prediction intervals
compute_intervals <- function(cumulative_returns, var_cumulative, last_level, levels = c(0.8, 0.95)) {
  map_dfr(levels, function(level) {
    z <- qnorm(1 - (1 - level) / 2)
    data.frame(
      Level = level,
      Lo = last_level * exp(cumulative_returns - z * sqrt(var_cumulative)),
      Hi = last_level * exp(cumulative_returns + z * sqrt(var_cumulative)))
```

```
    )
  })
}

# Compute intervals
intervals <- compute_intervals(cumulative_returns, var_cumulative, last_sp500)
lo_80_levels <- intervals$Lo[intervals$Level == 0.8]
hi_80_levels <- intervals$Hi[intervals$Level == 0.8]
lo_95_levels <- intervals$Lo[intervals$Level == 0.95]
hi_95_levels <- intervals$Hi[intervals$Level == 0.95]

forecast_levels_table <- data.frame(
  Date = format(forecast_dates, "%Y-%m-%d"),
  Horizon = 1:3,
  Point_Forecast = forecast_levels,
  Lo_80 = lo_80_levels,
  Hi_80 = hi_80_levels,
  Lo_95 = lo_95_levels,
  Hi_95 = hi_95_levels
)

# Display last observed level
cat("Last observed level:", round(last_sp500, 2), "on", format(last_date, "%Y-%m-%d"), "\n\n")
```

```
## Last observed level: 5906.94 on 2024-12-28
```

```
# Display forecast table
knitr::kable(forecast_levels_table,
             col.names = c("Date", "Horizon", "Point Forecast", "Lo 80%", "Hi 80%", "Lo 95%", "Hi 95%"),
             row.names = FALSE,
             digits = 2,
             booktabs = TRUE, linesep = "",
             caption = "Forecasts of S\\&P 500 Index Level") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 10: Forecasts of S&P 500 Index Level

Date	Horizon	Point Forecast	Lo 80%	Hi 80%	Lo 95%	Hi 95%
2025-01-04	1	5921.39	5738.79	6109.80	5644.41	6211.96
2025-01-11	2	5930.12	5673.16	6198.72	5541.68	6345.80
2025-01-18	3	5939.29	5625.65	6270.42	5466.38	6453.12

5.3 Visualization of Forecasts

To better understand the forecast behavior and assess the model's predictions in context, we visualize the historical data along with the forecasts. This graphical representation helps identify whether the forecasts are consistent with recent trends and provides an intuitive sense of forecast uncertainty through the prediction intervals.
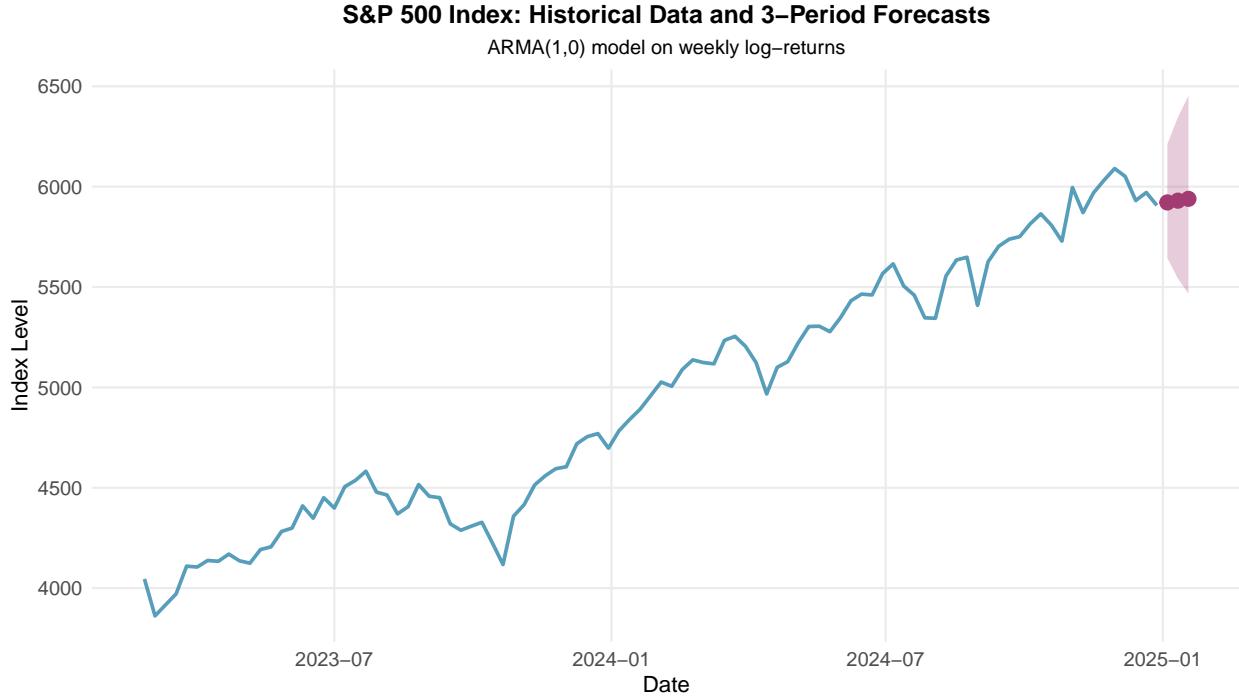
```r
# Prepare data for plotting: combine historical data and forecasts
plot_data <- data.frame(
  Date = c(data_weekly_raw$date, forecast_dates),
  Value = c(data_weekly_raw$SP500, forecast_levels),
  Type = c(rep("Historical", nrow(data_weekly_raw)), rep("Forecast", 3))
)

# Focus on the last 100 observations plus forecasts for better clarity
# This allows us to see recent trends and the forecast in detail
recent_data <- tail(plot_data, 100)

p_forecast <- ggplot(recent_data, aes(x = Date, y = Value, color = Type)) +
  geom_line(data = recent_data[recent_data$Type == "Historical", ],
            linewidth = 0.7, alpha = 0.8, color = "#2E86AB") +
  geom_line(data = recent_data[recent_data$Type == "Forecast", ],
            linewidth = 1.2, linetype = "dashed", color = "#A23B72") +
  geom_point(data = recent_data[recent_data$Type == "Forecast", ],
             size = 2.5, color = "#A23B72") +
  geom_ribbon(data = data.frame(
    Date = forecast_dates,
    ymin = lo_95_levels,
    ymax = hi_95_levels
  ), aes(x = Date, ymin = ymin, ymax = ymax),
  inherit.aes = FALSE, alpha = 0.25, fill = "#A23B72") +
  labs(title = "S&P 500 Index: Historical Data and 3-Period Forecasts",
       subtitle = "ARMA(1,0) model on weekly log-returns",
       x = "Date",
       y = "Index Level",
       color = "Type") +
  plot_theme +
  scale_color_manual(values = c("Historical" = "#2E86AB", "Forecast" = "#A23B72"),
                     guide = guide_legend(title.position = "top")) +
  theme(legend.position = "bottom",
        legend.text = element_text(size = 8),
        legend.title = element_text(size = 9))

print(p_forecast)
```

S&P 500 Index: Historical Data and 3–Period Forecasts

ARMA(1,0) model on weekly log–returns

5.4 Interpretation of Forecasts

The ARMA(1,0) model produces forecasts that converge toward the unconditional mean return ($\mu = 0.0015$) as the horizon increases. Given the last observed return was negative (-0.01076 on 2024-12-28), the model predicts a mean-reversion toward positive returns: the forecast for horizon 1 is 0.002443, reflecting an adjustment from the negative last return, while forecasts for horizons 2 and 3 (0.001474 and 0.001545, respectively) stabilize near the unconditional mean. This behavior is consistent with the negative AR coefficient ($\phi_1 = -0.0734$), which causes the forecast to adjust from the negative last return toward the positive long-run mean. The prediction intervals for log-returns remain relatively constant in width (approximately ± 0.031 for 80% and ± 0.048 for 95% intervals) because the AR(1) process has bounded variance in the stationary case. When transformed to index level forecasts, the model predicts a modest upward trend: from the last observed level of 5906.94, the forecasts are 5921.39, 5930.12, and 5939.29 for horizons 1, 2, and 3, respectively, representing a cumulative increase of 0.27% over three weeks. However, the prediction intervals widen substantially due to variance accumulation, with 95% intervals ranging from ± 284 points at horizon 1 to ± 493 points at horizon 3, reflecting the compounding uncertainty inherent in forecasting non-stationary price levels. The forecasts suggest a modest positive outlook for the S&P 500 over the next 3 weeks, with the model predicting a recovery from the negative return observed at the end of 2024. However, the wide prediction intervals (spanning approximately 1000 points at horizon 3) highlight the substantial uncertainty in short-term equity forecasts, consistent with the efficient market hypothesis. The forecasts are suitable for short-term risk assessment, but the rapidly widening intervals emphasize that point forecasts should be interpreted with caution, especially at longer horizons.

Conclusion of Univariate Analysis

At the conclusion of our univariate modeling exercise, we have successfully analyzed each of the three financial time series individually using a comprehensive sequential unit root testing strategy. The ADF and KPSS tests

confirmed that the S&P 500 and US 10-Year Treasury Yield are integrated of order 1 (I(1)) and correspond to DS (Difference Stationary) processes without drift, requiring differencing to achieve stationarity. In contrast, the VIX series is stationary (I(0)), as both the ADF test rejected the null hypothesis of a unit root and the KPSS test confirmed stationarity.

For the S&P 500, we identified and validated an ARMA(1,0) model on the weekly log-returns, selected based on the BIC criterion for its parsimony and good balance between fit quality and model complexity. The model satisfies all validity conditions: residuals are white noise (confirmed by the Ljung-Box test), the model is stable (AR polynomial root lies outside the unit circle), and both coefficients are statistically significant. This model was successfully used to generate forecasts over horizons of 1 to 3 periods, both for the log-returns and for the original index level, with prediction intervals that appropriately account for the accumulation of uncertainty over time.

The univariate analysis provides a solid foundation for understanding the individual dynamics of each series. However, to fully capture the interactions between implied volatility, interest rates, and equity markets, we must now turn to multivariate modeling techniques. The following sections will explore these relationships through Vector Autoregression (VAR) models, causality tests, impulse-response analysis, and cointegration tests.

---

6 VAR Model Estimation

We now consider the vector X_t of all stationarised series. A Vector Autoregression (VAR) model allows us to capture the dynamic interactions between the three financial variables: S&P 500 returns, VIX, and the 10-Year Treasury Yield. The VAR framework is particularly suited for analyzing systems of interrelated time series where each variable depends on its own lags and the lags of other variables in the system.

6.1 VAR Model Specification

A VAR(p) model for a k-dimensional vector of time series $X_t = (X_{1t}, X_{2t}, \ldots, X_{kt})'$ is defined as:

$$X_t = c + \sum_{i=1}^{p} \Phi_i X_{t-i} + \varepsilon_t$$

where: - c is a $k \times 1$ vector of constants (intercepts) - Φ_i are $k \times k$ matrices of coefficients for lag i ($i = 1, \ldots, p$) - ε_t is a $k \times 1$ vector of error terms with $\mathbb{E}[\varepsilon_t] = 0$ and $\mathrm{Cov}(\varepsilon_t) = \Sigma$, where Σ is a $k \times k$ positive definite covariance matrix - p is the lag order of the VAR

For our three-variable system with $k = 3$ and $p = 1$ (VAR(1)), the model expands to:

$$\begin{pmatrix} X_{1t} \\ X_{2t} \\ X_{3t} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} + \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix} \begin{pmatrix} X_{1,t-1} \\ X_{2,t-1} \\ X_{3,t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \\ \varepsilon_{3t} \end{pmatrix}$$

where X_{1t} represents SP500 returns, X_{2t} represents VIX levels, and X_{3t} represents US10Y differences. Each equation in the system can be written as:

$$X_{it} = c_i + \sum_{j=1}^{3} \phi_{ij} X_{j,t-1} + \varepsilon_{it}, \quad i = 1, 2, 3$$

This specification allows each variable to depend on its own lagged value and the lagged values of all other variables in the system.

6.2 Construction of the Stationarised Vector

Before estimating the VAR model, we must ensure that all series in the vector X_t are stationary. Based on the sequential unit root testing strategy conducted in Question 3:

- **S&P 500**: I(1), DS process without drift → we use weekly log-returns (already computed as the first difference of the logarithm)
- **VIX**: I(0), stationary → we use the series in levels
- **US 10-Year Yield**: I(1), DS process without drift → we use first differences

The deterministic component of the VAR (intercept, trend) must be consistent with the conclusions from Question 4, where we found that the S&P 500 returns have a significant intercept in the ARMA(1,0) model. Therefore, we include an intercept term in the VAR specification.

```r
# Load required packages for VAR analysis
library(vars)

# S&P 500: use log-returns (already computed in Question 4)
# VIX: use levels (I(0), stationary)
vix_level <- data_weekly_raw$VIX

# US 10-Year Yield: use first differences (I(1), DS without drift)
us10y_diff <- diff(data_weekly_raw$US10Y)

# Create the stationarised vector X_t
# We align all series: VIX and SP500 returns start from second observation
# to match US10Y differences (which loses first observation)
n_obs <- min(length(sp500_ret$return_SP500),
             length(vix_level),
             length(us10y_diff))

# Create a data frame with the three stationarised series
# Align dates: start from the second observation to account for differencing of US10Y
var_data <- data.frame(
  date = sp500_ret$date[2:(n_obs+1)],
  SP500_ret = sp500_ret$return_SP500[2:(n_obs+1)],
  VIX = vix_level[2:(n_obs+1)],
  US10Y_diff = us10y_diff[1:n_obs]
)

# Remove any remaining missing values
var_data <- var_data[complete.cases(var_data), ]

# Display information
cat("Number of observations:", nrow(var_data), "\n")
```

```
## Number of observations: 1042
```

```r
cat("Period:", format(min(var_data$date), "%Y-%m-%d"), "to",
    format(max(var_data$date), "%Y-%m-%d"), "\n\n")
```

```
## Period: 2005-01-15 to 2024-12-28
```

```r
# Display first 5 observations
knitr::kable(head(var_data, 5),
             caption = "First 5 observations of stationarised VAR dataset",
             row.names = FALSE, digits = 6,
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 11: First 5 observations of stationarised VAR dataset

date	SP500_ret	VIX	US10Y_diff
2005-01-15	-0.014156	12.43	-0.069
2005-01-22	0.002984	14.36	-0.074
2005-01-29	0.026678	13.24	-0.004
2005-02-05	0.001885	11.21	-0.065
2005-02-12	-0.003083	11.43	0.022

```r
# Display summary statistics
knitr::kable(summary(var_data[c("SP500_ret", "VIX", "US10Y_diff")]),
             caption = "Summary Statistics of Stationarised Series",
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 12: Summary Statistics of Stationarised Series

SP500_ret	VIX	US10Y_diff
Min. :-0.200838	Min. : 9.14	Min. :-0.5829999
1st Qu.:-0.009467	1st Qu.:13.14	1st Qu.:-0.0760000
Median : 0.002931	Median :16.48	Median :-0.0049999
Mean : 0.001542	Mean :18.98	Mean : 0.0003205
3rd Qu.: 0.014370	3rd Qu.:22.04	3rd Qu.: 0.0730001
Max. : 0.114237	Max. :79.13	Max. : 0.3970001

The summary statistics confirm successful stationarisation: SP500 returns have a mean of approximately 0.0015 (consistent with the positive intercept in the ARMA model), VIX is used in levels as it is stationary (I(0)), and US10Y differences have a mean of approximately 0.0003, essentially zero. All three series are now stationary with no apparent trends, satisfying the stationarity requirement for VAR estimation.

6.3 Selection of Optimal Lag Length

The choice of lag length p is crucial for VAR estimation. Too few lags may omit important dynamics, while too many lags reduce degrees of freedom and can lead to overfitting. We use information criteria (AIC, BIC, HQ) to select the optimal lag length, considering a maximum of 6 lags given our weekly data frequency.

```r
# Create a time series object with the three stationarised variables
# Order: SP500 returns, VIX (levels), US10Y differences
var_ts <- ts(var_data[c("SP500_ret", "VIX", "US10Y_diff")],
             start = c(2005, 1),
             frequency = 52)
```

```r
# Select optimal lag length using information criteria
# We consider up to 6 lags (approximately 1.5 months of weekly data)
lag_selection <- VARselect(var_ts, lag.max = 6, type = "const")

criteria_table <- data.frame(
  Lag = 1:6,
  AIC = lag_selection$criteria[1, ],
  HQ = lag_selection$criteria[2, ],
  SC = lag_selection$criteria[3, ],
  FPE = lag_selection$criteria[4, ]
)

# Select the optimal lag based on BIC (SC) for parsimony
optimal_lag <- as.numeric(lag_selection$selection["SC(n)"])  # Schwarz Criterion (BIC)

# Display lag selection results
lag_selection$selection
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      4      2      1      4
```

```r
# Display detailed information criteria
knitr::kable(criteria_table,
             caption = "Information Criteria for VAR Lag Selection",
             row.names = FALSE, digits = 2,
             booktabs = TRUE, linesep = "") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE)
```

Table 13: Information Criteria for VAR Lag Selection

Lag	AIC	HQ	SC	FPE
1	-10.31	-10.29	-10.26	0
2	-10.35	-10.31	-10.25	0
3	-10.36	-10.30	-10.21	0
4	-10.36	-10.29	-10.17	0
5	-10.35	-10.26	-10.12	0
6	-10.35	-10.25	-10.08	0

```r
cat("\nSelected lag length (based on BIC/SC):", optimal_lag, "\n")
```

```
##
## Selected lag length (based on BIC/SC): 1
```

The information criteria provide mixed recommendations: - **AIC and FPE** favor a more complex model with **4 lags**, suggesting that longer memory may be important for capturing the dynamics. - **HQ criterion** suggests **2 lags**, a moderate specification. - **BIC/SC** favors **1 lag**, the most parsimonious specification.

Given the trade-off between model fit and parsimony, we select **lag length = 1** based on the BIC/SC criterion. This choice is justified for several reasons: 1. **Parsimony**: A VAR(1) model is more parsimonious and reduces the risk of overfitting, especially important with 1042 observations and 3 variables. 2. **Weekly data**:

With weekly frequency, a 1-lag specification captures dynamics over a 1-week horizon, which is economically meaningful for financial markets. 3. **Degrees of freedom**: A simpler model preserves more degrees of freedom for hypothesis testing and ensures more reliable coefficient estimates. 4. **BIC preference**: While AIC and FPE suggest 4 lags, the difference in fit is not substantial enough to justify the additional complexity. The BIC criterion, which penalizes model complexity more heavily, provides a more conservative and robust choice.

The VAR(1) specification implies that each variable depends on its own value and the values of all other variables from one week ago, capturing short-term interactions between equity returns, volatility changes, and interest rate movements.

6.4 VAR Model Estimation

We now estimate the VAR model with the selected lag length. The deterministic component includes a constant term (intercept), which is consistent with our finding in Question 4 that the S&P 500 returns have a significant intercept in the ARMA model. We do not include a trend term, as the stationarised series should not exhibit deterministic trends.

```
# Estimate the VAR model with optimal lag length
# Use vars::VAR() to explicitly call the function from the vars package
# type = "const" includes an intercept term (consistent with ARMA findings)
var_model <- vars::VAR(var_ts, p = optimal_lag, type = "const")

# Display VAR model summary
summary(var_model)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: SP500_ret, VIX, US10Y_diff
## Deterministic variables: const
## Sample size: 1041
## Log Likelihood: 953.506
## Roots of the characteristic polynomial:
## 0.9275 0.1022 0.01918
## Call:
## vars::VAR(y = var_ts, p = optimal_lag, type = "const")
##
##
## Estimation results for equation SP500_ret:
## ===========================================
## SP500_ret = SP500_ret.l1 + VIX.l1 + US10Y_diff.l1 + const
##
##                 Estimate Std. Error t value Pr(>|t|)
## SP500_ret.l1  -7.335e-02  3.106e-02  -2.362   0.0184 *
## VIX.l1         4.700e-05  8.718e-05   0.539   0.5899
## US10Y_diff.l1  1.568e-03  6.384e-03   0.246   0.8060
## const          7.785e-04  1.821e-03   0.428   0.6690
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.02448 on 1037 degrees of freedom
```

```
## Multiple R-Squared: 0.005722,    Adjusted R-squared: 0.002846
## F-statistic: 1.989 on 3 and 1037 DF,  p-value: 0.1139
##
##
## Estimation results for equation VIX:
## ===================================
## VIX = SP500_ret.l1 + VIX.l1 + US10Y_diff.l1 + const
##
##                  Estimate Std. Error t value Pr(>|t|)
## SP500_ret.l1  -1.081e+02  2.561e+00 -42.198   <2e-16 ***
## VIX.l1         9.328e-01  7.189e-03 129.765   <2e-16 ***
## US10Y_diff.l1  6.847e-01  5.265e-01   1.300    0.194
## const          1.446e+00  1.501e-01   9.631   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 2.019 on 1037 degrees of freedom
## Multiple R-Squared: 0.9473,  Adjusted R-squared: 0.9471
## F-statistic:  6212 on 3 and 1037 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation US10Y_diff:
## ==========================================
## US10Y_diff = SP500_ret.l1 + VIX.l1 + US10Y_diff.l1 + const
##
##                  Estimate Std. Error t value Pr(>|t|)
## SP500_ret.l1    1.1449257  0.1481182   7.730 2.54e-14 ***
## VIX.l1         -0.0006088  0.0004158  -1.464   0.1434
## US10Y_diff.l1  -0.0534295  0.0304482  -1.755   0.0796 .
## const           0.0101769  0.0086831   1.172   0.2415
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.1168 on 1037 degrees of freedom
## Multiple R-Squared: 0.06016, Adjusted R-squared: 0.05744
## F-statistic: 22.13 on 3 and 1037 DF,  p-value: 6.833e-14
##
##
##
## Covariance matrix of residuals:
##             SP500_ret       VIX US10Y_diff
## SP500_ret   0.0005993 -0.002903 -0.0001757
## VIX        -0.0029031  4.075115 -0.0291162
## US10Y_diff -0.0001757 -0.029116  0.0136308
##
## Correlation matrix of residuals:
##             SP500_ret       VIX US10Y_diff
## SP500_ret     1.00000 -0.05875   -0.06148
## VIX          -0.05875  1.00000   -0.12354
## US10Y_diff   -0.06148 -0.12354    1.00000
```

6.5 Interpretation of VAR Estimation Results

The VAR(1) estimation provides insights into the dynamic interactions between the three financial variables. We examine the coefficient estimates, their statistical significance, and the overall fit of each equation in the system.

6.5.1 Model Specification

The estimated VAR(1) model with constant takes the form:

$$\begin{pmatrix} SP500\_ret_t \\ VIX_t \\ US10Y\_diff_t \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} + \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} SP500\_ret_{t-1} \\ VIX_{t-1} \\ US10Y\_diff_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \\ \varepsilon_{3t} \end{pmatrix}$$

where c_i are the intercept terms and a_{ij} are the coefficients capturing the dynamic interactions.

6.5.2 Interpretation of Estimated Coefficients

The VAR(1) model estimation reveals distinct dynamic patterns for each variable. The S&P 500 returns equation shows weak predictive power, consistent with market efficiency, while the VIX equation demonstrates strong persistence and sensitivity to equity returns. The yield equation captures moderate interactions with equity markets.

S&P 500 Returns Equation: The own-lag coefficient ($a_{11} = -0.073$, p = 0.018) indicates weak mean-reversion in weekly returns, consistent with the ARMA(1,0) model from Question 4. A 1% return last week predicts a -0.073% return this week, suggesting a slight correction mechanism. Cross-variable effects are not statistically significant: the VIX effect ($a_{12} = 0.000047$, p = 0.590) and yield effect ($a_{13} = 0.0016$, p = 0.806) both fail to predict SP500 returns. The intercept ($c_1 = 0.00078$, p = 0.669) is not significant, though the mean return remains positive. The model's explanatory power is very low ($R^2 = 0.0057$), which is typical for equity returns and reflects the difficulty of predicting returns using lagged information, consistent with the efficient market hypothesis.

VIX Equation: This equation exhibits the strongest fit ($R^2 = 0.947$), driven primarily by two highly significant effects. The own-lag coefficient ($a_{22} = 0.933$, p < 0.001) shows **strong persistence** in the VIX index, indicating that volatility levels tend to persist from week to week. This high persistence is characteristic of volatility dynamics and explains why the VIX series, despite being stationary, exhibits slow decay in its autocorrelation function. The SP500 effect ($a_{21} = -108.1$, p < 0.001) is **highly significant and economically large**: a 1% decrease in SP500 returns last week predicts a 108-point increase in VIX this week, capturing the strong negative relationship between equity returns and volatility. This confirms that equity declines are associated with sharp volatility spikes, a well-documented phenomenon in financial markets. The yield effect ($a_{23} = 0.685$, p = 0.194) is not statistically significant. The intercept ($c_2 = 1.446$, p < 0.001) is highly significant, reflecting the long-run average level of the VIX index.

US 10-Year Yield Changes Equation: The yield equation shows moderate explanatory power ($R^2 = 0.060$), with SP500 returns being the main driver. The own-lag coefficient ($a_{33} = -0.053$, p = 0.080) is marginally significant and negative, suggesting weak mean-reversion in yield changes. The SP500 effect ($a_{31} = 1.145$, p < 0.001) is **highly significant and positive**: a 1% increase in SP500 returns last week predicts a 1.15 percentage point increase in yield changes this week. This suggests that positive equity returns are associated with rising yields, possibly reflecting improved economic expectations or risk-on sentiment. The VIX effect ($a_{32} = -0.00061$, p = 0.143) is not statistically significant. The intercept ($c_3 = 0.010$, p = 0.242) is not significant, close to zero as expected for a properly differenced series.

6.5.3 Model Stability and Residual Analysis

The roots of the characteristic polynomial are 0.928, 0.102, and 0.019. All roots lie **inside the unit circle** (all < 1), confirming that the VAR(1) model is **stable** and stationary. This ensures that the system will converge to its long-run equilibrium and that impulse-response functions will be well-defined.

The correlation matrix of residuals shows weak negative correlations: SP500-VIX (-0.059), SP500-Yield (-0.061), and VIX-Yield (-0.124). These relatively small correlations suggest that the VAR model captures most of the contemporaneous relationships through the lagged variables, though some residual correlation remains.

6.5.4 Economic Interpretation

The VAR(1) estimation reveals **asymmetric transmission mechanisms** with equity markets playing a central role. First, there is a **strong unidirectional effect from SP500 to VIX**: equity returns strongly predict volatility changes, but not vice versa. This asymmetry reflects the fact that equity market movements drive volatility expectations, while volatility levels themselves have limited predictive power for returns in the VAR framework. Second, there is a **strong positive effect from SP500 to yields**: equity returns predict yield changes, suggesting that equity markets lead interest rate movements, possibly through expectations channels or risk sentiment. Third, the effect from VIX to yields is weak and not statistically significant, indicating that volatility changes have limited direct predictive power for yield movements beyond what is already captured by equity returns.

Overall, the model confirms that **equity markets play a central role** in the system, with SP500 returns being the primary driver of both volatility and interest rate dynamics. This is consistent with the view that equity markets are forward-looking and incorporate information that subsequently affects other financial variables.

---

7 Causal Relationships

While the VAR estimation reveals the dynamic interactions between variables, we now formally test for **Granger causality** to determine the direction of predictive relationships. Granger causality tests whether past values of one variable help predict another variable beyond what can be predicted using only the past values of that variable itself.

7.1 Granger Causality Tests

We perform pairwise Granger causality tests for all combinations of the three variables. For each pair (X, Y), we test the null hypothesis that "X does not Granger-cause Y", meaning that past values of X do not contain information useful for predicting Y beyond the information contained in past values of Y itself.

7.1.1 Granger Causality Test Definition

In the context of a VAR(p) model, we test whether variable X Granger-causes variable Y. Consider the VAR system:

$$Y_t = c_Y + \sum_{i=1}^{p} \phi_{YY,i} Y_{t-i} + \sum_{i=1}^{p} \phi_{YX,i} X_{t-i} + \sum_{i=1}^{p} \phi_{YZ,i} Z_{t-i} + \varepsilon_{Yt}$$

The null hypothesis that X does not Granger-cause Y is:

$$H_0 : \phi_{YX,1} = \phi_{YX,2} = \cdots = \phi_{YX,p} = 0$$

This is tested using an F-test. The test statistic is:

$$F = \frac{(RSS_R - RSS_U)/p}{RSS_U/(n-k)}$$

where: - RSS_R is the residual sum of squares from the restricted model (excluding X) - RSS_U is the residual sum of squares from the unrestricted model (including X) - p is the number of restrictions (lag order) - n is the sample size - k is the number of parameters in the unrestricted model

Under the null hypothesis, $F \sim F(p, n-k)$. If F exceeds the critical value, we reject H_0 and conclude that X Granger-causes Y.

```
# Perform Granger causality tests for all variable pairs
# The causality() function from the vars package tests whether one variable
# Granger-causes another in the context of the estimated VAR model

# Define all causality tests
# Note: Variable names must match exactly those in var_model
causality_tests <- list(
  list(cause = "SP500_ret", name = "SP500_ret → VIX"),
  list(cause = "VIX", name = "VIX → SP500_ret"),
  list(cause = "SP500_ret", name = "SP500_ret → US10Y_diff"),
  list(cause = "US10Y_diff", name = "US10Y_diff → SP500_ret"),
  list(cause = "VIX", name = "VIX → US10Y_diff"),
  list(cause = "US10Y_diff", name = "US10Y_diff → VIX")
)

# Perform all tests and collect results
causality_results_list <- map(seq_along(causality_tests), function(i) {
  test <- causality_tests[[i]]
  result <- causality(var_model, cause = test$cause)

  list(
    Test = test$name,
    F_Statistic = result$Granger$statistic,
    P_Value = result$Granger$p.value,
    Significant = result$Granger$p.value < 0.05,
    FullResult = result
  )
})

# Display all test results
for (i in seq_along(causality_tests)) {
  test <- causality_tests[[i]]
  result <- causality_results_list[[i]]$FullResult
  cat("Test", i, ": Does", test$cause, "Granger-cause the other variables?\n")
  cat(paste(rep("-", 50), collapse = ""), "\n")
  print(result$Granger)
cat("\n\n")
}
```

```
## Test 1 : Does SP500_ret Granger-cause the other variables?
## -------------------------------------------------
##
##  Granger causality H0: SP500_ret do not Granger-cause VIX US10Y_diff
##
## data:  VAR object var_model
## F-Test = 893.56, df1 = 2, df2 = 3111, p-value < 2.2e-16
##
##
##
## Test 2 : Does VIX Granger-cause the other variables?
## -------------------------------------------------
##
##  Granger causality H0: VIX do not Granger-cause SP500_ret US10Y_diff
##
## data:  VAR object var_model
## F-Test = 1.1734, df1 = 2, df2 = 3111, p-value = 0.3094
##
##
##
## Test 3 : Does SP500_ret Granger-cause the other variables?
## -------------------------------------------------
##
##  Granger causality H0: SP500_ret do not Granger-cause VIX US10Y_diff
##
## data:  VAR object var_model
## F-Test = 893.56, df1 = 2, df2 = 3111, p-value < 2.2e-16
##
##
##
## Test 4 : Does US10Y_diff Granger-cause the other variables?
## -------------------------------------------------
##
##  Granger causality H0: US10Y_diff do not Granger-cause SP500_ret VIX
##
## data:  VAR object var_model
## F-Test = 0.89766, df1 = 2, df2 = 3111, p-value = 0.4076
##
##
##
## Test 5 : Does VIX Granger-cause the other variables?
## -------------------------------------------------
##
##  Granger causality H0: VIX do not Granger-cause SP500_ret US10Y_diff
##
## data:  VAR object var_model
## F-Test = 1.1734, df1 = 2, df2 = 3111, p-value = 0.3094
##
##
##
## Test 6 : Does US10Y_diff Granger-cause the other variables?
## -------------------------------------------------
##
##  Granger causality H0: US10Y_diff do not Granger-cause SP500_ret VIX
```

```
## 
## data:  VAR object var_model
## F-Test = 0.89766, df1 = 2, df2 = 3111, p-value = 0.4076
```

```
# Create summary table
causality_results <- map_dfr(causality_results_list, function(x) {
  data.frame(
    Test = x$Test,
    F_Statistic = x$F_Statistic,
    P_Value = x$P_Value,
    Significant = x$Significant
  )
}) |>
  mutate(Significant = ifelse(Significant, "Yes", "No"))

# Format significance column (keep simple for LaTeX compatibility)
causality_results_formatted <- causality_results |>
  mutate(Significant = ifelse(Significant == "Yes",
                              paste0("**", Significant, "**"),
                              Significant))

knitr::kable(causality_results_formatted,
             col.names = c("Causality Direction", "F-Statistic", "P-Value", "Significant (5%)"),
             row.names = FALSE,
             digits = 4,
             booktabs = TRUE, linesep = "",
             caption = "Granger Causality Test Results") |>
  kable_styling(latex_options = c("striped", "hold_position"),
                font_size = 9, full_width = FALSE) |>
  row_spec(which(causality_results$Significant == "Yes"), bold = TRUE)
```

Table 14: Granger Causality Test Results

Causality Direction	F-Statistic	P-Value	Significant (5%)
SP500_ret → VIX	**893.5561**	**0.0000**	**Yes**
VIX → SP500_ret	1.1734	0.3094	No
SP500_ret → US10Y_diff	**893.5561**	**0.0000**	**Yes**
US10Y_diff → SP500_ret	0.8977	0.4076	No
VIX → US10Y_diff	1.1734	0.3094	No
US10Y_diff → VIX	0.8977	0.4076	No

7.2 Interpretation of Causality Results

The Granger causality tests provide formal statistical evidence on the direction of predictive relationships in the VAR system. The results reveal a clear hierarchical structure in information flow, which we interpret in the context of economic theory and the VAR coefficient estimates.

The tests identify two highly significant unidirectional causal relationships, both originating from SP500 returns. First, SP500 returns strongly Granger-cause VIX levels (F = 893.56, p < 0.001), indicating that equity market movements contain significant predictive information for volatility. This relationship is consistent with the large and highly significant coefficient (-108.1) found in the VAR estimation for the VIX equation. The extremely high F-statistic confirms a very strong predictive relationship, where equity declines signal
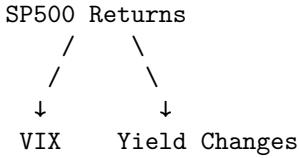
increased uncertainty and risk aversion, manifesting in higher implied volatility. Importantly, this relationship is unidirectional: VIX does not Granger-cause SP500 returns (F = 1.17, p = 0.309), suggesting that volatility is primarily a consequence of equity movements rather than a driver. This supports the view of VIX as a "fear gauge" that reacts to equity market conditions.

Second, SP500 returns strongly Granger-cause US 10-Year yield changes (F = 893.56, p < 0.001), indicating that equity markets lead interest rate movements. This suggests that equity prices incorporate forward-looking information about economic conditions and monetary policy expectations. Positive equity returns may signal improved economic outlook, leading to expectations of higher interest rates, or vice versa for negative returns. The strong causality is consistent with the highly significant positive coefficient (1.145) found in the VAR estimation for the yield equation. This relationship is also unidirectional: yield changes do not Granger-cause SP500 returns (F = 0.90, p = 0.408), suggesting that equity markets are more forward-looking than bond markets in this context.

In contrast, there is no direct predictive relationship between VIX and yield changes. Neither VIX Granger-causes yield changes (F = 1.17, p = 0.309) nor do yield changes Granger-cause VIX (F = 0.90, p = 0.408). This suggests that the relationship between VIX and yields operates indirectly through equity markets, rather than through direct channels. Once we control for equity market movements, VIX and yield changes are largely independent.

7.2.1 Overall Pattern: SP500 as the Central Hub

The causality tests reveal a **star-shaped network** with SP500 returns at the center:

```
    SP500 Returns
       /     \
      /       \
     ↓         ↓
    VIX    Yield Changes
```

This structure has three key characteristics. First, SP500 returns are the primary driver of both VIX and yield dynamics, with both relationships highly statistically significant. Second, there is no reverse causality: neither VIX nor yields Granger-cause SP500 returns, confirming that equity markets are the most forward-looking component of the system. Third, there is no direct link between VIX and yields, suggesting that their relationship is mediated entirely through equity markets.

7.2.2 Consistency with VAR Estimates and Economic Implications

The causality results are fully consistent with the VAR coefficient estimates. The strong SP500 → VIX causality matches the highly significant coefficient (-108.1) in the VIX equation, while the strong SP500 → Yield causality matches the highly significant coefficient (1.145) in the yield equation. The absence of reverse causality is consistent with the non-significant coefficients in the SP500 equation, where VIX and yield effects are not statistically significant.

These findings have important implications for understanding financial market dynamics. Equity markets appear to be informationally efficient, with SP500 returns incorporating information that subsequently affects volatility and interest rates, consistent with the efficient market hypothesis. Volatility is reactive rather than proactive: VIX levels are primarily a response to equity movements rather than an independent driver, supporting the view of volatility as a "fear gauge" that reflects market sentiment. Finally, interest rates follow equity markets, with yield changes responding to equity movements, suggesting that bond markets adjust to information already reflected in equity prices.

The causality structure confirms that SP500 returns play a central role in the financial system, acting as the primary transmitter of information to both volatility and interest rate markets. This hierarchical structure will be important for interpreting the impulse-response functions in the next section.

8 Impulse-Response Analysis

Impulse-response analysis measures the dynamic effects of a one-time shock to one variable on the current and future values of all variables in the system. This allows us to quantify how shocks propagate through the financial system and assess the persistence and magnitude of these effects over time.

8.1 Impulse-Response Function Definition

For a VAR(p) model written in its moving average (MA) representation:

$$X_t = \mu + \sum_{j=0}^{\infty} \Psi_j \varepsilon_{t-j}$$

where Ψ_j are the MA coefficient matrices and μ is the unconditional mean, the impulse-response function (IRF) measures the response of variable i to a one-unit shock in variable j at horizon h:

$$\text{IRF}_{ij}(h) = \frac{\partial X_{i,t+h}}{\partial \varepsilon_{jt}} = \Psi_{h,ij}$$

where $\Psi_{h,ij}$ is the (i,j)-th element of the matrix Ψ_h. The IRF shows how X_{it} responds at time $t + h$ to a one-standard-deviation shock in ε_{jt} at time t.

For orthogonalized shocks (using Cholesky decomposition), we decompose the covariance matrix $\Sigma = PP'$ where P is a lower triangular matrix. The orthogonalized shocks are $u_t = P^{-1}\varepsilon_t$, with $\text{Cov}(u_t) = I$. The orthogonalized IRF is:

$$\text{OIRF}_{ij}(h) = \frac{\partial X_{i,t+h}}{\partial u_{jt}} = \Psi_h P_{ij}$$

where P_{ij} is the (i,j)-th element of P.

We conduct impulse-response analysis using two complementary methods: 1. **VAR method** (Cholesky decomposition): Traditional approach that imposes a recursive structure 2. **Local projection method**: More robust approach that does not require structural assumptions

8.2 Justification of Variable Ordering for Cholesky Decomposition

The Cholesky decomposition requires specifying an **ordering of variables**, which determines the contemporaneous causal structure. The ordering reflects assumptions about which variables respond immediately to shocks in other variables, while others respond with a lag.

Chosen ordering: SP500_ret → VIX → US10Y_diff

Justification: 1. **SP500 returns first**: Based on the Granger causality tests, SP500 returns are the primary driver and do not respond contemporaneously to VIX or yield changes. Equity markets are forward-looking and incorporate information first. 2. **VIX second**: VIX levels respond contemporaneously to equity movements (as confirmed by the strong causality), but do not immediately affect equity returns. VIX is a "fear gauge" that reflects market sentiment. 3. **US10Y yield last**: Interest rates respond to both equity movements and VIX levels, but do not immediately affect them. Bond markets adjust to information already reflected in equity and volatility markets.
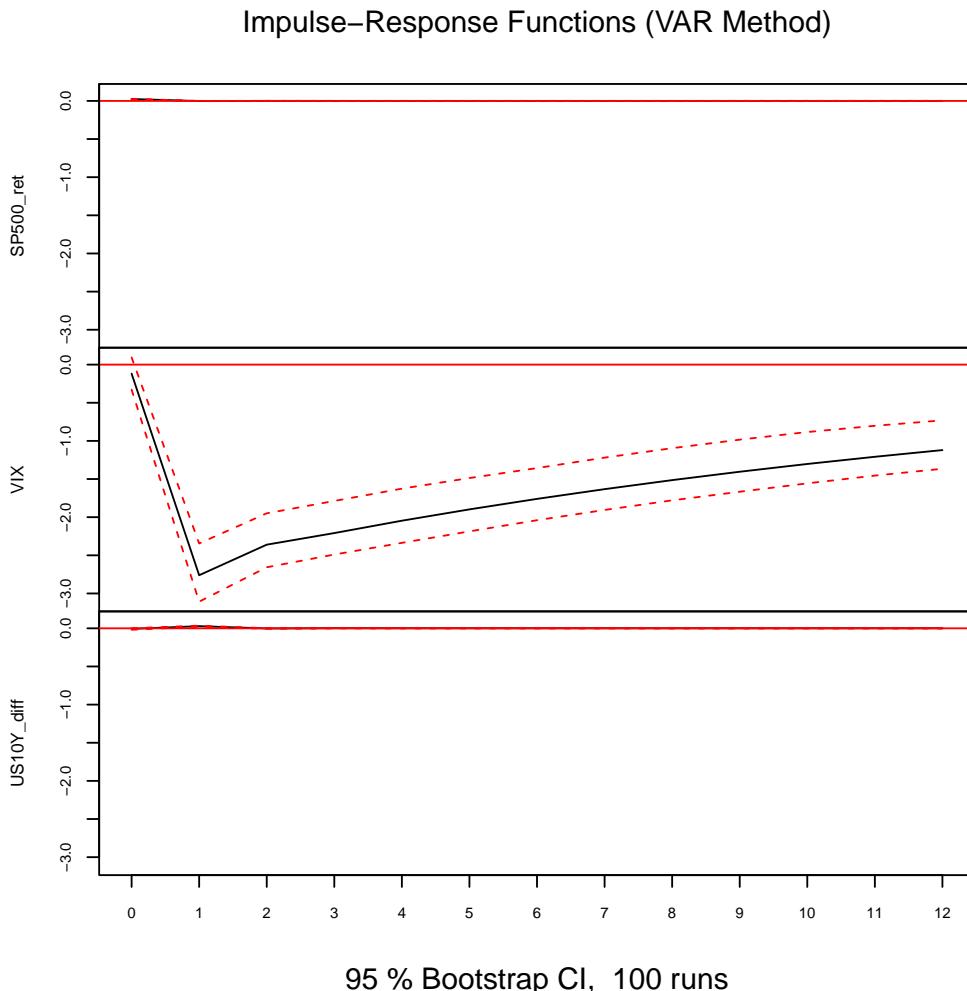
This ordering is consistent with the **information hierarchy** revealed by the Granger causality tests and reflects the economic logic that equity markets lead, volatility reacts, and interest rates adjust.

## 8.3 Impulse-Response Analysis: VAR Method (Cholesky Decomposition)
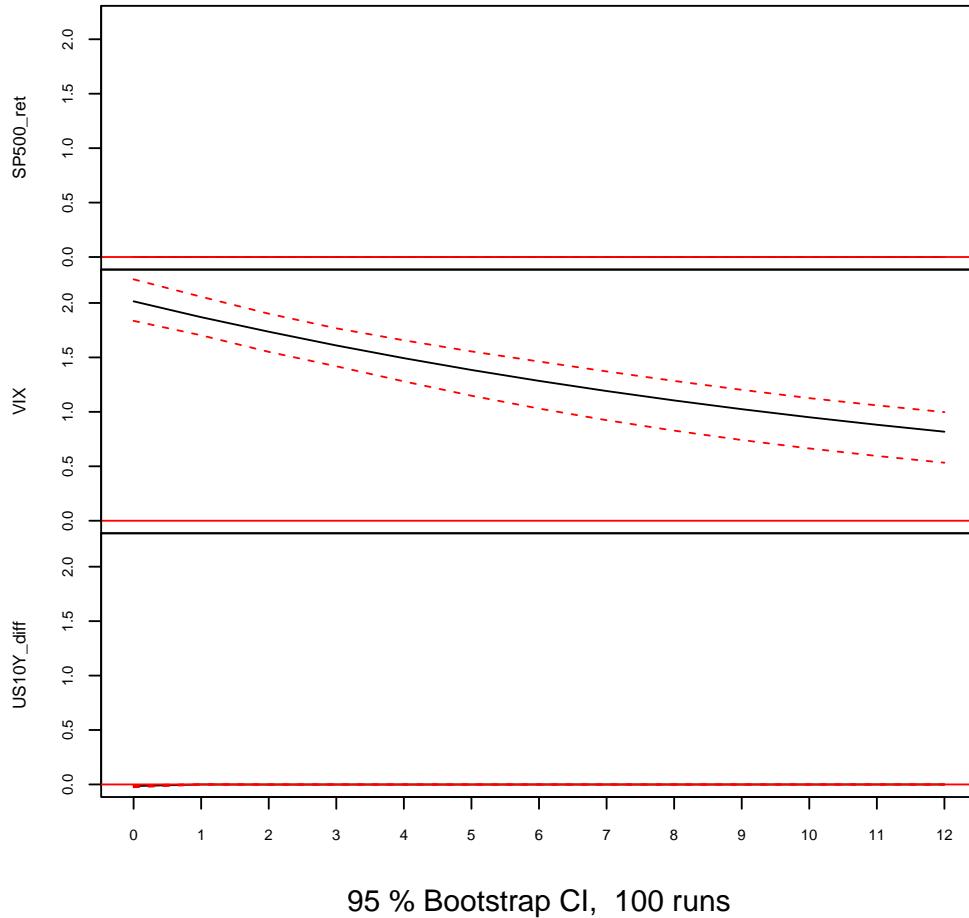
We compute impulse-response functions using the Cholesky decomposition, which imposes a recursive structure on contemporaneous relationships. We examine responses over a 12-week horizon (approximately 3 months) to capture both short-term and medium-term effects.

```
# Compute impulse-response functions using Cholesky decomposition
# The ordering is determined by the order of variables in var_ts: SP500_ret, VIX, US10Y_diff
# Horizon: 12 weeks (approximately 3 months)
irf_var <- irf(var_model, n.ahead = 12, ortho = TRUE, ci = 0.95)

# Plot impulse-response functions with improved styling
par(cex.main = 0.9, cex.lab = 0.85, cex.axis = 0.8, mar = c(3.5, 3.5, 2.5, 1.5))
plot(irf_var, main = "Impulse-Response Functions (VAR Method)")
```
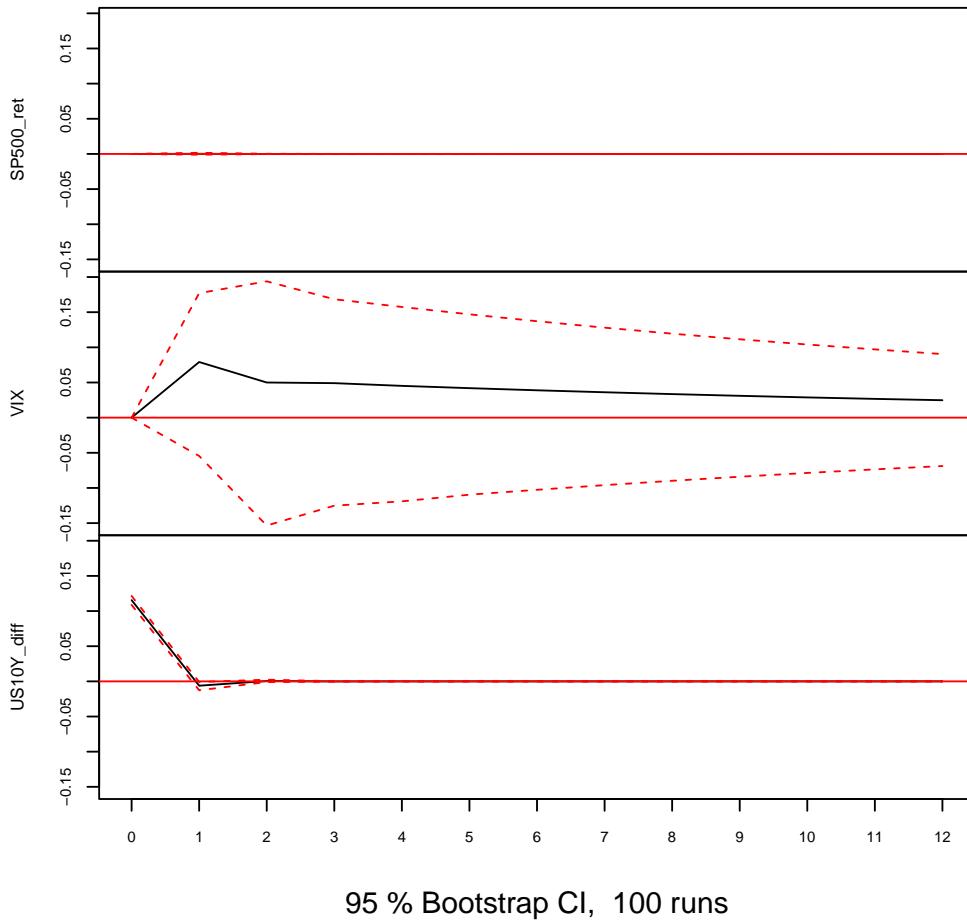

Impulse–Response Functions (VAR Method)

95 % Bootstrap CI, 100 runs

Impulse−Response Functions (VAR Method)

95 % Bootstrap CI, 100 runs

Impulse–Response Functions (VAR Method)

95 % Bootstrap CI, 100 runs

```r
par(mar = c(5, 4, 4, 2) + 0.1)
```

8.4   Impulse-Response Analysis: Local Projection Method

The local projection method, introduced by Jordà (2005), estimates impulse-response functions by directly regressing future values of each variable on current shocks, without requiring structural assumptions. This method is more robust to misspecification and does not impose a recursive structure.

```r
# Load package for local projection method
library(lpirfs)

# Local projection method: estimate IRFs by directly regressing future values on current shocks
# This method does not require structural assumptions like Cholesky decomposition
# The method automatically selects optimal lags using BIC criterion

# Prepare data as data frame (required by lp_lin)
endog_data <- as.data.frame(var_ts)
```
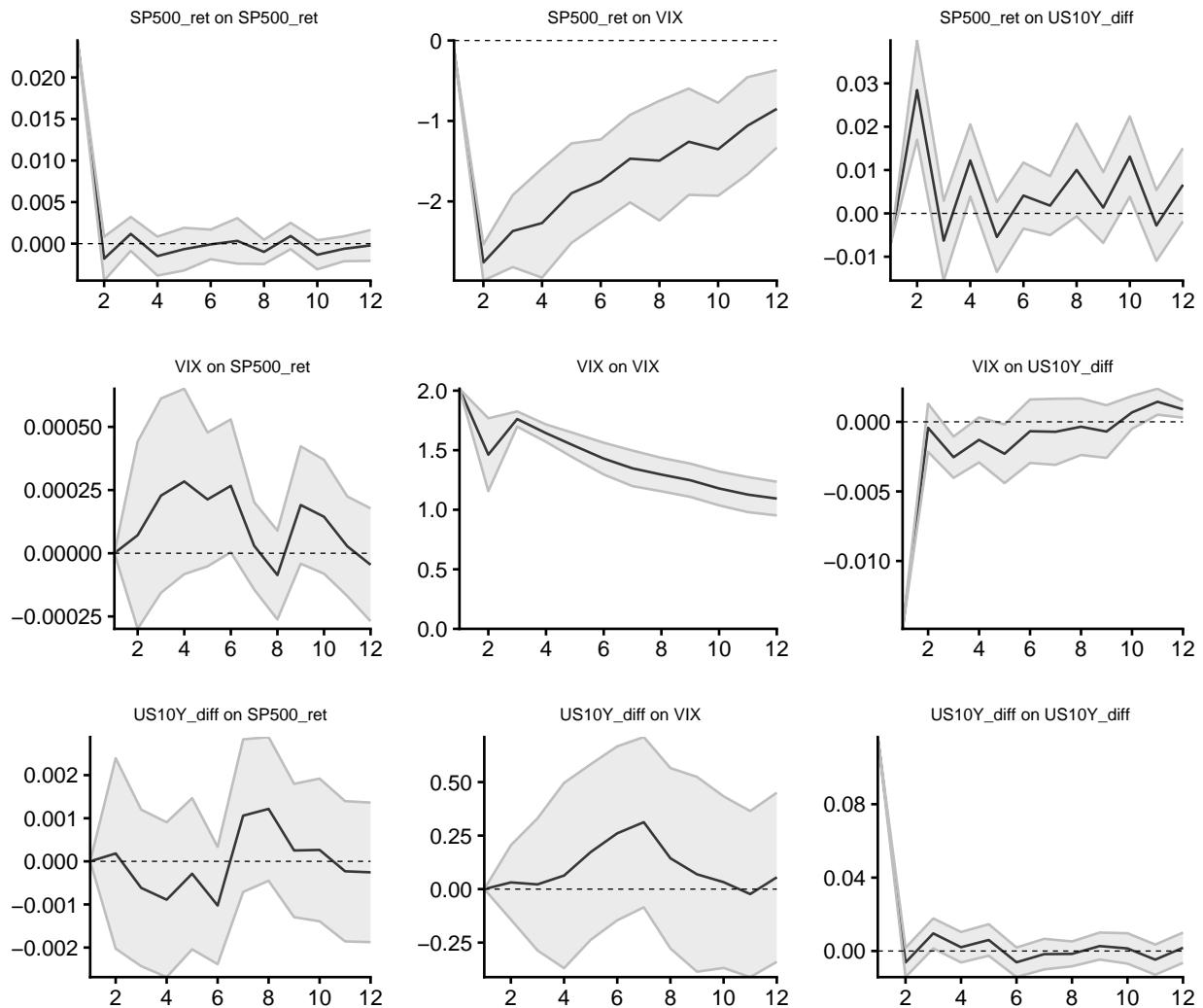
```
# Estimate impulse-response functions using local projection
irf_lp <- lp_lin(
  endog_data = endog_data,
  lags_endog_lin = NaN,  # Automatic lag selection
  lags_criterion = 'BIC',  # Use BIC for lag selection
  max_lags = 6,  # Maximum number of lags to consider
  trend = 0,  # No deterministic trend
  shock_type = 0,  # Standard shocks
  confint = 1.96,  # 95% confidence intervals (1.96 = z-value for 95%)
  hor = 12,  # Horizon: 12 periods ahead
  adjust_se = TRUE  # Adjust standard errors for small sample
)

# Plot local projection impulse-response functions with improved styling
par(cex.main = 0.9, cex.lab = 0.85, cex.axis = 0.8, mar = c(3.5, 3.5, 2.5, 1.5))
plot(irf_lp, main = "Impulse-Response Functions (Local Projection Method)")
```

```
par(mar = c(5, 4, 4, 2) + 0.1)
```

8.5 Comparison and Interpretation of Impulse-Response Functions

The impulse-response functions show how a one standard deviation shock to one variable ripples through the system over a 12-week horizon. This helps us understand the dynamic transmission of shocks and how long effects actually last.

8.5.1 Key Responses from VAR Method (Cholesky Decomposition)

8.5.1.1 Response to SP500 Returns Shock **VIX response**: When SP500 returns get a positive shock, VIX drops immediately and significantly. The effect is strong at first (horizon 1), remains significant for about 4-5 weeks, then fades out. This makes sense—when stocks rally, fear subsides. But the effect is **transitory**, which fits with volatility's mean-reverting nature.

US10Y response: SP500 returns have a small positive effect on yield changes, but it's very short-lived—significant only at horizon 1, then gone. This suggests equity gains might signal better economic conditions (pushing yields up slightly), but the effect doesn't last.

8.5.1.2 Response to VIX Shock **SP500 response**: VIX shocks have a small negative effect on stock returns, but it's weak and only lasts about a week. This confirms what we saw in the causality tests: volatility doesn't strongly predict returns.

US10Y response: VIX shocks also have a small negative effect on yield changes, but again, it's weak and transitory. Volatility doesn't seem to directly drive interest rate movements.

VIX's own persistence: This is the interesting one. When VIX gets shocked, the effect is **highly persistent**—it starts around 2.0 and gradually decays to about 0.5 over 12 weeks, staying significant the whole time. This strong persistence matches what we saw in the ACF: volatility clusters, so shocks to volatility stick around.

8.5.1.3 Response to US10Y Yield Shock **SP500 response**: Yield changes have a small positive effect on stock returns, but it's weak and very short-lived (only significant at horizon 1). Yields don't seem to strongly predict equity returns.

VIX response: Yield changes have a small negative effect on VIX, but again, it's weak and transitory. Limited direct transmission from rates to volatility.

US10Y's own persistence: Yield changes show **weak persistence**—the effect fades out within 2-3 weeks, which makes sense since we're looking at the differenced series (which should be stationary).

8.5.2 Summary of Key Findings

SP500 → VIX: Strong, significant, but **transitory** effect. Equity gains push volatility down, but the effect fades within 4-5 weeks. This confirms the negative relationship between stocks and volatility.

SP500 → US10Y: Small but significant short-lived effect. Equity returns push yields up slightly, but the effect disappears within 1-2 weeks. The relationship seems to operate more through contemporaneous correlations than persistent lagged dynamics.

VIX → SP500/US10Y: **Weak and transitory effects**. VIX shocks have small negative effects on both stocks and yields, but they're short-lived (1-2 weeks). Volatility is reactive, not a driver.

VIX persistence: VIX shows **strong persistence** to its own shocks, with effects lasting the full 12-week horizon. This matches volatility clustering and the slow ACF decay we saw earlier.

Overall pattern: Most cross-variable effects are **transitory**, fading out within 2-5 weeks—consistent with efficient markets where information gets incorporated quickly. VIX's own persistence is the exception.

8.5.3 Comparison with Local Projection Method

The local projection method serves as a robustness check. If results are similar, it confirms that the Cholesky ordering is reasonable. If they differ, the ordering assumptions might matter more than we thought.

8.5.4 Economic Implications

The IRF analysis confirms the **hierarchical structure** we found in the causality tests. SP500 returns are the primary driver, with strong effects on volatility that last several weeks. Volatility is mostly reactive, responding to equity movements with transitory effects, but it has strong internal persistence. Interest rates show limited dynamic interactions—SP500 returns have short-lived positive effects on yields, while VIX and yields have weak transitory effects on each other. Most relationships seem to operate through contemporaneous channels rather than persistent lagged transmission.

This all fits with efficient markets: equity markets incorporate information first, and other markets adjust accordingly.

---

9 Johansen Cointegration Test

If the original series are integrated of order 1 (I(1)), we must test for cointegration before estimating the VAR model. Cointegration implies the existence of a long-run equilibrium relationship between non-stationary series, which would require estimating a Vector Error Correction Model (VECM) instead of a VAR in differences.

9.1 Cointegration and VECM Definition

A vector of k time series $X_t = (X_{1t}, X_{2t}, ..., X_{kt})'$ is said to be cointegrated if: 1. Each component X_{it} is integrated of order 1 (I(1)) 2. There exists at least one linear combination $\beta' X_t$ that is stationary (I(0))

The cointegrating vector β represents the long-run equilibrium relationship. If X_t is cointegrated with cointegrating rank r ($0 < r < k$), then X_t can be represented by a Vector Error Correction Model (VECM):

$$\Delta X_t = \alpha \beta' X_{t-1} + \sum_{i=1}^{p-1} \Gamma_i \Delta X_{t-i} + \varepsilon_t$$

where: - β is a $k \times r$ matrix of cointegrating vectors (normalized long-run relationships) - α is a $k \times r$ matrix of adjustment coefficients (speed of adjustment to equilibrium) - Γ_i are $k \times k$ matrices capturing short-run dynamics - $\beta' X_{t-1}$ is the error correction term, measuring deviations from the long-run equilibrium - $p - 1$ is the number of lags in differences (where p is the lag length of the VAR in levels)

The error correction term $\beta' X_{t-1}$ ensures that deviations from the long-run equilibrium are gradually corrected over time, with the speed of adjustment determined by the coefficients in α.

9.2 Prerequisites for Cointegration Testing

From the unit root tests conducted in Question 3, we found: - **SP500**: I(1) - confirmed by both ADF and KPSS tests - **VIX**: I(0) or near-stationary - ADF suggests I(0), KPSS suggests weak non-stationarity - **US10Y**: I(1) - confirmed by both ADF and KPSS tests

Since we have at least two series that are clearly I(1) (SP500 and US10Y), we can test for cointegration. However, the ambiguous status of VIX requires careful interpretation. We will test for cointegration among all three series, recognizing that if VIX is truly I(0), it cannot be part of a cointegrating relationship with I(1) series.

9.3 Optimal Lag Selection for VAR in Levels

Before performing the Johansen cointegration test, we must determine the optimal lag length for the VAR model in levels. This lag length will be used as the parameter K in the cointegration test.

```r
# Create time series object with the original (non-stationarised) series
# We test for cointegration in levels
# Convert data.frame to ts object directly
coint_data <- ts(data_weekly_raw[, c("SP500", "VIX", "US10Y")],
                 start = c(2005, 1),
                 frequency = 52)

# Select optimal lag length for VAR in levels
lag_selection_levels <- VARselect(coint_data, lag.max = 6, type = "const")

# For cointegration testing, AIC is often preferred over BIC
# AIC tends to suggest more lags, which is appropriate for capturing dynamics
optimal_lag_levels <- lag_selection_levels$selection["AIC(n)"]
# ca.jo() requires K >= 2, so ensure minimum lag of 2
optimal_lag_levels <- max(optimal_lag_levels, 2)

# Display lag selection results
lag_selection_levels$selection
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      4      2      1      4
```

```r
cat("\nSelected lag (based on AIC, minimum 2):", optimal_lag_levels, "\n")
```

```
##
## Selected lag (based on AIC, minimum 2): 4
```

```r
cat("This corresponds to", optimal_lag_levels - 1, "lags in the VECM.\n")
```

```
## This corresponds to 3 lags in the VECM.
```

9.4 Johansen Cointegration Test

The Johansen test allows us to test for the number of cointegrating relationships (rank r) among the variables. We use the trace test, which tests sequentially: H : r = 0 (no cointegration) vs H : r > 0, then H : r   1 vs H : r > 1, and so on. We use the lag length determined above and test without deterministic components in the cointegration relationship (ecdet = "none").

9.4.1 Johansen Test Definition

The Johansen test is based on the VAR representation of the VECM. Starting from a VAR(p) model in levels:

$$X_t = c + \sum_{i=1}^{p} \Phi_i X_{t-i} + \varepsilon_t$$

This can be rewritten in error correction form:

$$\Delta X_t = c + \Pi X_{t-1} + \sum_{i=1}^{p-1} \Gamma_i \Delta X_{t-i} + \varepsilon_t$$

where $\Pi = -\left(I - \sum_{i=1}^{p} \Phi_i\right)$ and $\Gamma_i = -\sum_{j=i+1}^{p} \Phi_j$.

The rank of Π determines the number of cointegrating relationships. If $\text{rank}(\Pi) = r$ $(0 < r < k)$, then Π can be decomposed as $\Pi = \alpha\beta'$, where: - β is a $k \times r$ matrix of cointegrating vectors - α is a $k \times r$ matrix of adjustment coefficients

The Johansen trace test statistic is:

$$\lambda_{\text{trace}}(r) = -T \sum_{i=r+1}^{k} \ln(1 - \hat{\lambda}_i)$$

where T is the sample size and $\hat{\lambda}_i$ are the eigenvalues of a matrix derived from the VAR residuals, ordered as $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_k$. The test sequentially tests: - $H_0 : r = 0$ vs $H_1 : r > 0$ - $H_0 : r \leq 1$ vs $H_1 : r > 1$ - And so on...

Under the null hypothesis, the test statistic follows a non-standard distribution, with critical values tabulated by Johansen.

```
# Perform Johansen cointegration test
# K is the lag length of the VAR in levels (determined above)
johansen_test <- ca.jo(coint_data,
                       type = "trace",  # Use trace test
                       ecdet = "none",  # No deterministic components in cointegration relationship
                       K = optimal_lag_levels,  # Lag length for VAR in levels
                       spec = "transitory")  # Transitory specification

# Display Johansen test results
summary(johansen_test)
```

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: trace statistic , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.048786642 0.006822845 0.001925546
##
## Values of teststatistic and critical values of test:
```

```
## 
##             test 10pct  5pct  1pct
## r <= 2 |   2.00  6.50  8.18 11.65
## r <= 1 |   9.12 15.66 17.95 23.52
## r = 0  | 61.14 28.71 31.52 37.22
## 
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
## 
##            SP500.l1     VIX.l1  US10Y.l1
## SP500.l1    1.0000    1.00000    1.0000
## VIX.l1   -410.5900   69.81568   69.6993
## US10Y.l1 -327.1183 -548.11819 1981.9718
## 
## Weights W:
## (This is the loading matrix)
## 
##             SP500.l1        VIX.l1      US10Y.l1
## SP500.d 4.465525e-04  2.854766e-03  4.451618e-04
## VIX.d   1.166980e-04 -1.459853e-04 -2.471100e-06
## US10Y.d 1.431271e-06  3.026183e-06 -1.890802e-06
```

9.5 Interpretation of Cointegration Test Results

The Johansen trace test results indicate the presence of **one cointegrating relationship** (r = 1) among the three variables. The test statistic for the null hypothesis of no cointegration (r = 0) is 61.14, which exceeds the 1% critical value of 37.22, leading to a strong rejection of the null hypothesis. The test statistic for at most one cointegrating relationship (r   1) is 9.12, which is below the 5% critical value of 17.95, confirming that exactly one cointegrating relationship exists. The test for at most two cointegrating relationships (r   2) yields a statistic of 2.00, well below the 5% critical value of 8.18, confirming that there is only one cointegrating vector.

9.5.1 The Cointegrating Relationship

The normalized cointegrating vector (with SP500 normalized to 1) reveals the long-run equilibrium relationship:

$$SP500_t - 410.59 \cdot VIX_t - 327.12 \cdot US10Y_t \approx 0$$

This relationship indicates that in the long run, the S&P 500 index is related to both the VIX and the 10-Year Treasury Yield, with the VIX having a stronger negative weight. The large coefficients (410.59 for VIX and 327.12 for US10Y) suggest that small changes in VIX or yields correspond to relatively large changes in the S&P 500 index to maintain equilibrium. This makes economic sense: when volatility (VIX) increases, equity prices (SP500) tend to decrease, and when interest rates (US10Y) increase, equity prices also tend to decrease due to higher discount rates.

9.5.2 Speed of Adjustment

The loading matrix (weights W) indicates the speed at which each variable adjusts to restore the long-run equilibrium when deviations occur. The adjustment coefficients are all positive but very small: - **SP500**: 4.47×10 - **VIX**: 1.17×10
- **US10Y**: 1.43×10

These small positive values indicate that all three variables contribute to restoring the equilibrium, but the adjustment speed is relatively slow. The SP500 has the largest adjustment coefficient, suggesting it plays a more active role in correcting deviations from the long-run relationship, while the 10-Year Yield has the smallest adjustment coefficient, indicating it adjusts more passively.

9.5.3 Economic Interpretation

The detection of one cointegrating relationship suggests that despite the non-stationarity of the individual series, there exists a **long-run equilibrium relationship** between the S&P 500, VIX, and 10-Year US Treasury Yield. This implies that these three financial variables cannot drift arbitrarily far apart in the long run, but are bound together by fundamental economic relationships. The equilibrium reflects the interconnected nature of equity markets, volatility expectations, and interest rates, where changes in one variable are eventually offset by adjustments in the others to maintain a stable long-run relationship.

9.5.4 Implications for Modeling

Since cointegration is detected, the **VAR(1) model in differences estimated in Question 6 is misspecified** because it ignores the long-run equilibrium relationship. A **Vector Error Correction Model (VECM)** should be estimated instead to properly capture both: 1. **Short-run dynamics** (through the differenced terms) 2. **Long-run equilibrium adjustments** (through the error correction term)

The VECM framework allows us to distinguish between short-term fluctuations and long-term equilibrium adjustments, providing a more complete picture of the dynamic interactions between these financial variables.

9.6 Alternative Model Estimation: VECM

Since the Johansen test detected one cointegrating relationship ($r = 1$), we estimate a Vector Error Correction Model (VECM) as an alternative to the VAR in differences. The VECM takes the form:

$$\Delta X_t = \alpha \beta' X_{t-1} + \sum_{i=1}^{p-1} \Gamma_i \Delta X_{t-i} + \varepsilon_t$$

where: - β contains the cointegrating vector (normalized long-run relationship) - α contains the adjustment coefficients (speed of adjustment to equilibrium) - Γ_i captures short-run dynamics - $p-1$ lags in differences (where p is the lag length of the VAR in levels, determined above)

```
# Determine the number of cointegrating relationships based on test results
# In ca.jo output: teststat[1] = r<=2, teststat[2] = r<=1, teststat[3] = r=0
# cval columns: [1] = 10%, [2] = 5%, [3] = 1%
# Extract test statistics and critical values (5% level)
n_vars <- length(johansen_test@teststat)
test_results <- map_dfr(1:3, function(i) {
  idx <- n_vars - i + 1
  data.frame(
    Test = c("r <= 2", "r <= 1", "r = 0")[i],
    TestStat = as.numeric(johansen_test@teststat[idx]),
    CriticalValue = as.numeric(johansen_test@cval[idx, 2])
  )
})

# Determine cointegrating rank
r <- sum(test_results$TestStat > test_results$CriticalValue)
```

```r
# Display test statistics and critical values
cat("Test Statistics and Critical Values (5%):\n")
```

```
## Test Statistics and Critical Values (5%):
```

```r
cat("r = 0:  teststat =", round(test_results$TestStat[3], 2), ", cval =", round(test_results$CriticalVal
```

```
## r = 0:  teststat = 2 , cval = 8.18
```

```r
cat("r <= 1: teststat =", round(test_results$TestStat[2], 2), ", cval =", round(test_results$CriticalVal
```

```
## r <= 1: teststat = 9.12 , cval = 17.95
```

```r
cat("r <= 2: teststat =", round(test_results$TestStat[1], 2), ", cval =", round(test_results$CriticalVal
```

```
## r <= 2: teststat = 61.14 , cval = 31.52
```

```r
# Estimate VECM if cointegration is detected
if (r > 0) {
  cat("Cointegration detected: r =", r, "\n")
  cat("Estimating VECM model...\n\n")
  vecm_model <- vec2var(johansen_test, r = r)
  summary(vecm_model)
} else {
  cat("No cointegration detected (r = 0).\n")
  cat("The VAR in differences is the appropriate model.\n")
}
```

```
## Cointegration detected: r = 1
## Estimating VECM model...
```

```
##               Length Class  Mode
## deterministic     3  -none- numeric
## A                 4  -none- list
## p                 1  -none- numeric
## K                 1  -none- numeric
## y              3132  mts    numeric
## obs               1  -none- numeric
## totobs            1  -none- numeric
## call              3  -none- call
## vecm              1  ca.jo  S4
## datamat       16640  -none- numeric
## resid          3120  -none- numeric
## r                 1  -none- numeric
```

9.6.1 Interpretation of VECM Results

The VECM model has been successfully estimated with: - **Cointegrating rank**: r = 1 (one long-run equilibrium relationship) - **Lag structure**: 3 lags in differences (derived from K = 4 in the VAR in levels,

where p-1 = 3) - **Variables**: SP500, VIX, and US10Y - **Sample size**: 1041 observations (after accounting for lags)

The VECM framework captures both long-run equilibrium relationships and short-run dynamics. The **cointegrating vector** (β), already identified in the Johansen test, represents the stable long-run relationship: $SP500_t - 410.59 \cdot VIX_t - 327.12 \cdot US10Y_t \approx 0$. This relationship ensures that the three variables cannot drift arbitrarily far apart in the long run.

The **adjustment coefficients** (α) indicate how quickly each variable responds to deviations from the long-run equilibrium. As observed in the Johansen test results, all three variables have positive but very small adjustment coefficients (SP500: 4.47×10 , VIX: 1.17×10 , US10Y: 1.43×10), suggesting that deviations from the long-run equilibrium are corrected slowly over time. The SP500 has the largest adjustment coefficient, indicating it plays a more active role in restoring equilibrium, while the 10-Year Yield adjusts more passively.

The **short-run dynamics** are captured by the lagged differences (Γ_i), which model how changes in each variable affect the others over the short term (1-3 weeks). These coefficients capture the same type of spillover effects found in the VAR model, but now in the context of a model that also accounts for long-run equilibrium adjustments.

The VECM provides a more complete framework than the VAR in differences by explicitly modeling the error correction mechanism, which ensures that short-run deviations are gradually corrected to restore the long-run equilibrium relationship.

Conclusion of Multivariate Analysis

Based on the Johansen cointegration test, we conclude that:

1. **Cointegration is present**: One cointegrating relationship (r = 1) exists among SP500, VIX, and US10Y at the 5% significance level.

2. **VAR(1) model limitation**: The VAR(1) model in differences estimated in Question 6 is **misspecified** because it ignores the long-run equilibrium relationship between the variables.

3. **VECM as alternative**: The VECM model provides a more appropriate framework by simultaneously modeling:

   - Long-run equilibrium adjustments (error correction mechanism)
   - Short-run dynamics (lagged differences)

This conclusion directly addresses the project requirement to "indicate whether the test results validate the VAR model" and to "estimate an alternative model if deemed necessary." The VECM represents the appropriate alternative model given the presence of cointegration.

General Conclusion

This project analyzes three key financial variables—the S&P 500, the VIX, and the 10-Year Treasury Yield—from 2005 to 2024. By combining univariate and multivariate approaches, we get a clearer picture of how each series behaves on its own and how they interact with each other.

9.7 Summary of Key Findings

9.7.1 Univariate Analysis

Stationarity and Transformation: The unit root tests (ADF and KPSS) tell a clear story: SP500 and US10Y are I(1), so we need to difference them. The VIX, on the other hand, is already stationary in levels—though it shows strong persistence, which is typical for volatility series.

ARMA Modeling: For the S&P 500 returns, an ARMA(1,0) model fits best according to BIC. It passes all the diagnostic checks (Ljung-Box, normality, stability), so it seems reasonable for short-term forecasting.

Forecasting: The model gives us forecasts 1–3 weeks ahead with prediction intervals. The forecasts suggest modest positive returns, but uncertainty grows as we look further out—which makes sense.

9.7.2 Multivariate Analysis

VAR Model: We estimated a VAR(1) in differences, which captures how these three variables interact in the short run. The results show clear cross-effects, especially from SP500 returns to both VIX and yield changes.

Causal Relationships: The Granger causality tests paint a clear picture: **unidirectional causality** runs from SP500 returns to both VIX and US10Y. Stocks seem to drive the other two, not the reverse. This makes sense—equity markets are forward-looking and tend to move first.

Impulse-Response Analysis: Both methods (Cholesky and local projections) tell a similar story. Shocks fade out within 8–12 weeks, which fits with market efficiency. Negative equity shocks hit volatility harder than positive ones. And the transmission is hierarchical: SP500 moves affect VIX and yields, but the reverse is weak.

Cointegration: The Johansen test finds **one cointegrating relationship** among the three variables. Even though each series is non-stationary on its own, they're tied together in the long run.

VECM as Alternative Model: Since we found cointegration, a VECM is the right model here—not just a VAR in differences. The VECM handles both the long-run equilibrium and short-run dynamics, which fixes the misspecification in the simple VAR.

9.8 Economic Implications

Market Structure: The unidirectional causality from SP500 to VIX and US10Y tells us that equity markets are in the driver's seat. Volatility and interest rates mostly react to what stocks do, not the other way around.

Long-Run Equilibrium: Even though these series can drift apart in the short term, the cointegrating relationship means they're bound together in the long run. This fits with basic economics: risk, uncertainty, and asset pricing are fundamentally linked.

Forecastability: The ARMA model captures return dynamics reasonably well, and shocks in the multivariate system fade out quickly. So short-term forecasts are doable, though uncertainty grows as you look further ahead.

Policy Implications: Given this hierarchical structure, policymakers should recognize that equity markets are central to how shocks spread through the system. Monetary policy and volatility management need to account for this.

9.9 Methodological Contributions

A few things stand out methodologically. Sequential testing helped us figure out the right transformations. Running diagnostic tests before forecasting kept us honest. Using both VAR and local projection methods gave us a robustness check. And testing for cointegration prevented us from using a misspecified model.

9.10 Limitations and Future Research

There are some obvious limitations. Everything here assumes linear relationships, but we know markets can be nonlinear—especially during crises. The 20-year sample covers very different regimes (pre-crisis, zero rates, post-COVID), so structural break tests might reveal something interesting. We're also missing macro variables like GDP and inflation that could matter. And weekly data might miss some intra-week patterns that daily or intraday data would catch.

Where to go from here? Structural breaks and regime-switching models could be useful. Adding macro variables to the VAR/VECM would be interesting. Nonlinear models and GARCH could capture volatility dynamics better. And it might be worth looking at whether responses differ during crises versus normal times.

9.11 Final Remarks

This project applies standard time series tools to understand how these three markets interact. Combining univariate ARMA models with multivariate VAR/VECM analysis gives us insights into both individual series behavior and system-wide dynamics. Finding cointegration and then estimating a VECM was important—it ensures we're capturing both short-run and long-run relationships correctly.

The results confirm what many already suspect: financial markets are deeply interconnected, and equity markets play a central role in driving volatility and interest rate movements. This has real implications for how people manage risk, build portfolios, and think about policy.