Rproject3

2024-03-13

Analyzing Invesco QQQ Trust (QQQ) Daily Closing Prices Over Five Years

```
library(xts)
```

```
## Warning: package 'xts' was built under R version 4.3.3
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.3.3
```

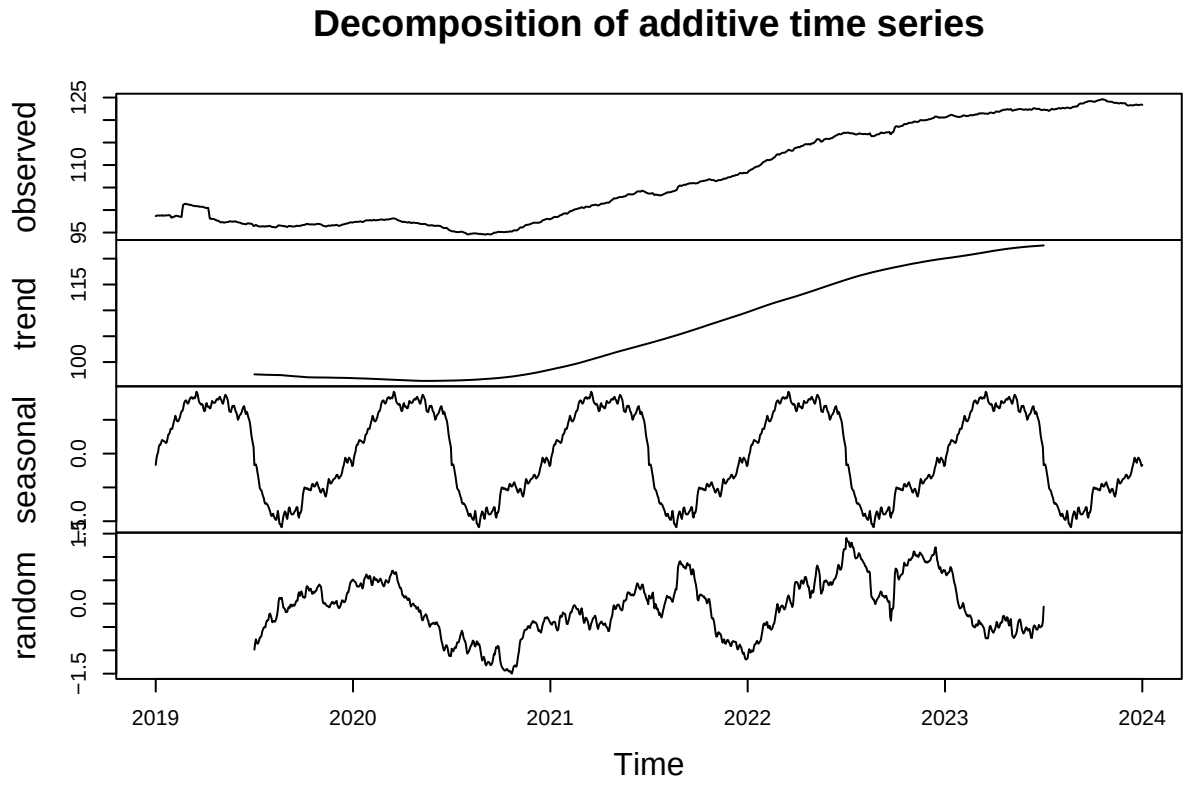
```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
# Read the CSV file and convert DATE column to Date format
df <- read.csv("Data.csv")
df$DATE <- as.Date(df$Date, format = '%Y-%m-%d')

# Convert "Close" column to numeric
df$Close <- as.numeric(as.character(df$Close))
```

```
## Warning: NAs introduced by coercion
```

```
# Check for non-numeric values in the "Close" column
non_numeric <- df$Close[!is.numeric(df$Close)]

#print(non_numeric)

# Create a time series object
df <- ts(df$Close, start = '2019', end='2024', frequency = 252)

# Decompose the time series
decomposed <- decompose(df)


# Plot the decomposed components
plot(decomposed)
```



Analyzing Invesco QQQ Trust (QQQ) Dataset with a 90-Day Moving Average

```
# task 2
# Load necessary libraries
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.3.3
```

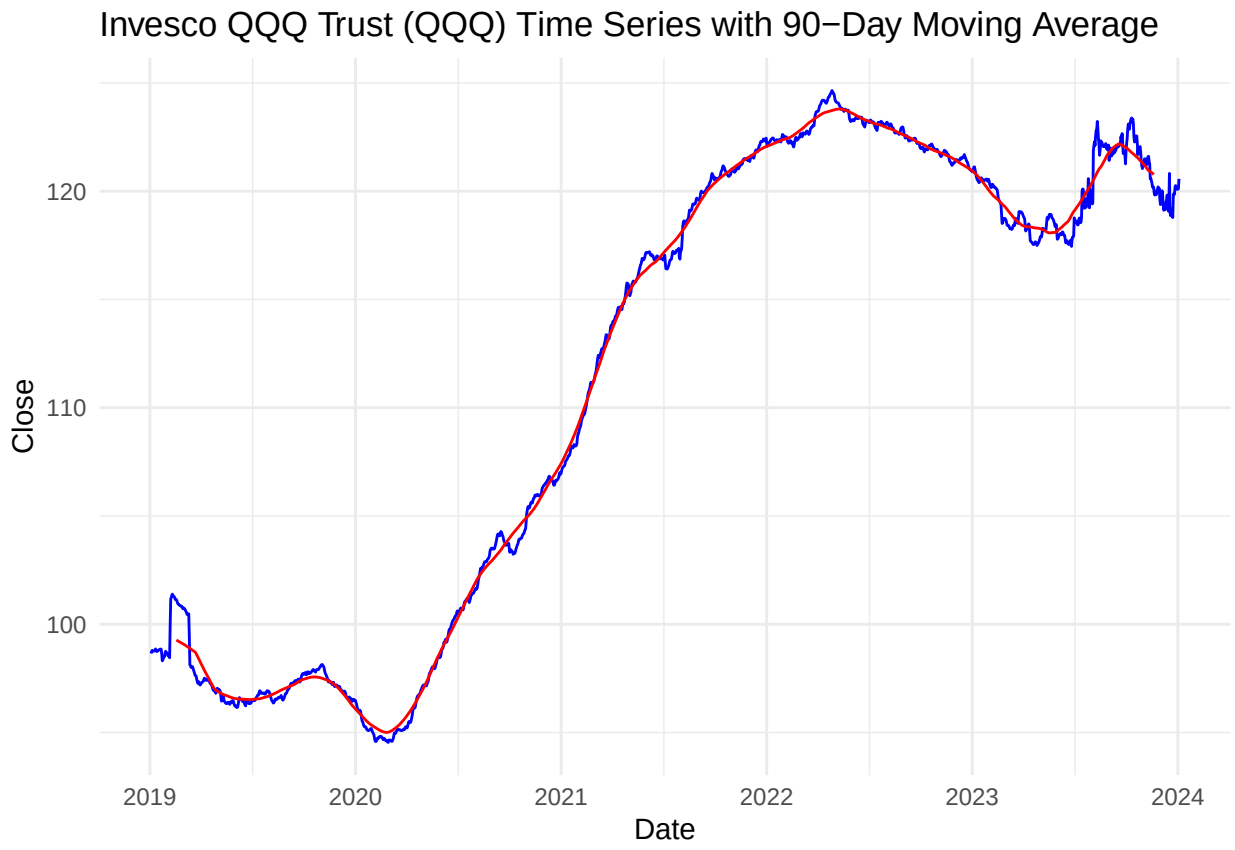
```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(ggplot2)

df <- read.csv('Data.csv')

# Convert df
ts_df <- ts(df$Close, start = c(2019, 3), end = c(2024, 3), frequency = 365)

# Calculate the 90-day moving average
ma_90 <- ma(ts_df, order = 90)

# Convert ts_df and ma_90 to data frames for plotting
df_plot <- data.frame(Date = time(ts_df), Close = as.numeric(ts_df))
ma_90_plot <- data.frame(Date = time(ma_90), MA_90 = as.numeric(ma_90))
```

```r
# Plot the original time series and the 90-day moving average
ggplot() +
  geom_line(data = df_plot, aes(x = Date, y = Close), color = "blue") +
  geom_line(data = ma_90_plot, aes(x = Date, y = MA_90), color = "red") +
  labs(title = "Invesco QQQ Trust (QQQ) Time Series with 90-Day Moving Average",
       x = "Date",
       y = "Close") +
  theme_minimal()
```

```
## Don't know how to automatically pick scale for object of type <ts>. Defaulting
## to continuous.
```

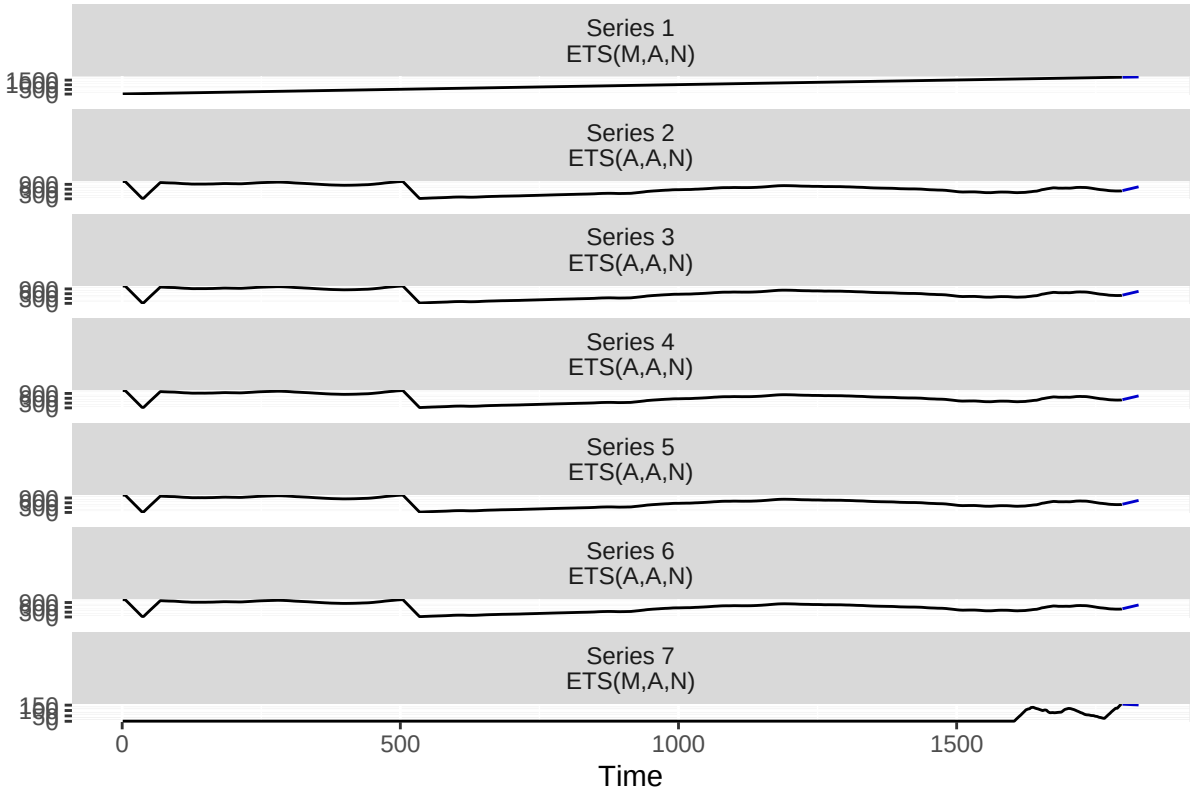
```
## Warning: Removed 90 rows containing missing values (`geom_line()`).
```



Forecasting Invesco QQQ Trust (QQQ) Closing Prices for the Next 30 Days

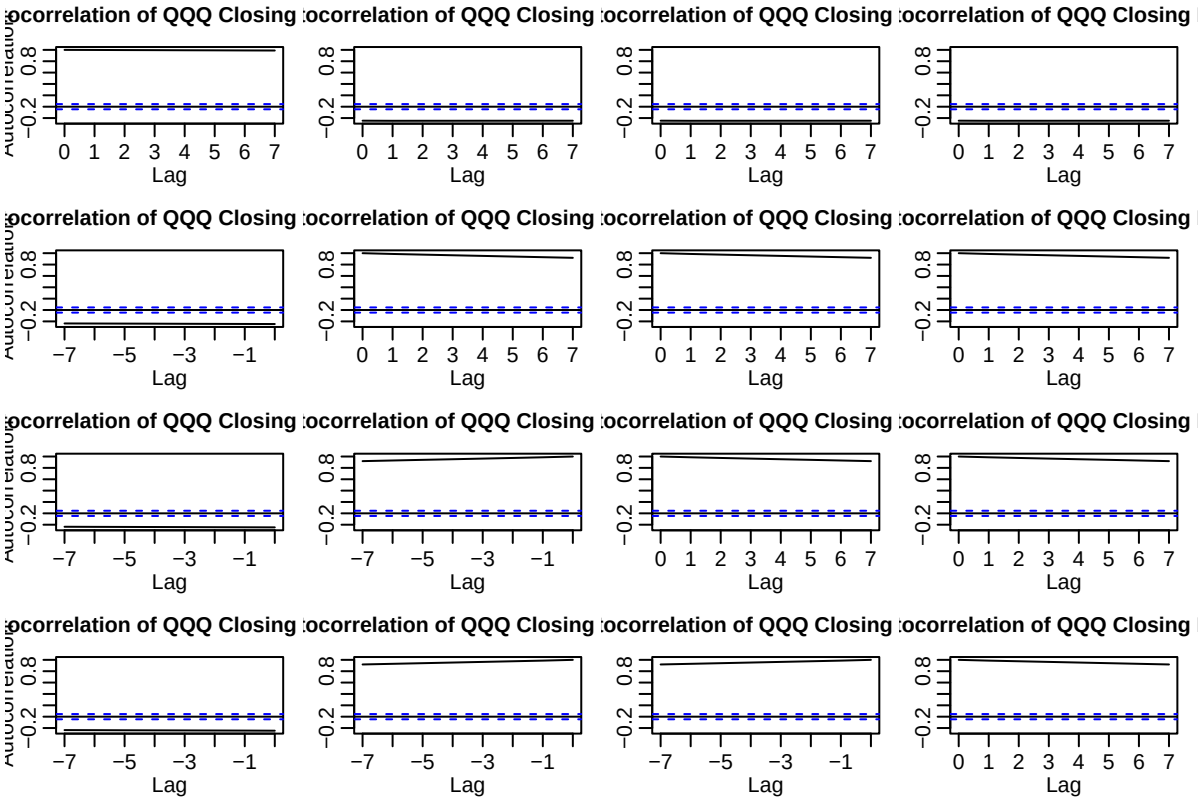
```r
# Calculate the 30-day moving average
df_ma <- ma(df, order = 30)

# Exclude any NA values
df_ma <- ts(na.exclude(df_ma))

# Forecast the next 30 days
myforecast <- forecast(df_ma, level = c(0), h = 30)
```

```r
# Plot the forecast
autoplot(myforecast)
```

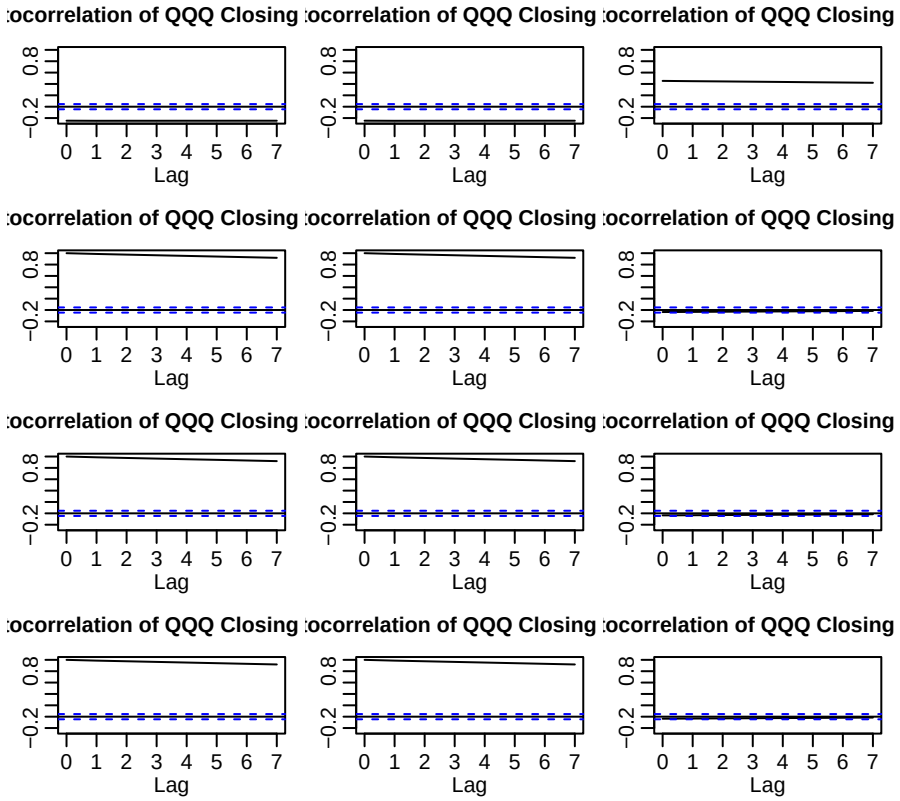


Computing and Plotting Autocorrelation of Invesco QQQ Trust (QQQ) Closing Prices

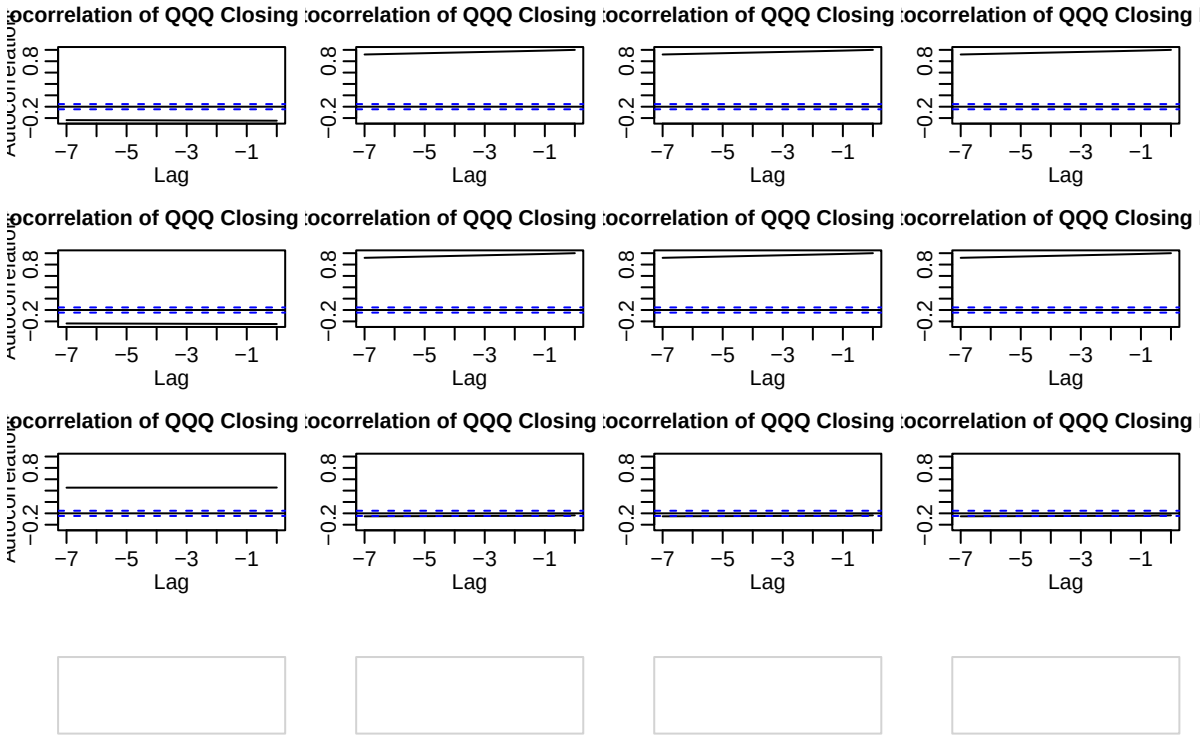
```r
# Compute autocorrelation
acf_obs <- acf(df, lag.max = length(df), plot = FALSE)
# Plot autocorrelation
plot(acf_obs, type = "l", main = "Autocorrelation of QQQ Closing Prices", xlab = "Lag", ylab = "Autocor
```


ocorrelation of QQQ Closing
Autocorrelation
−0.2
0.8
−7 −5 −3 −1
0 1 2 3 4 5 6 7
Lag

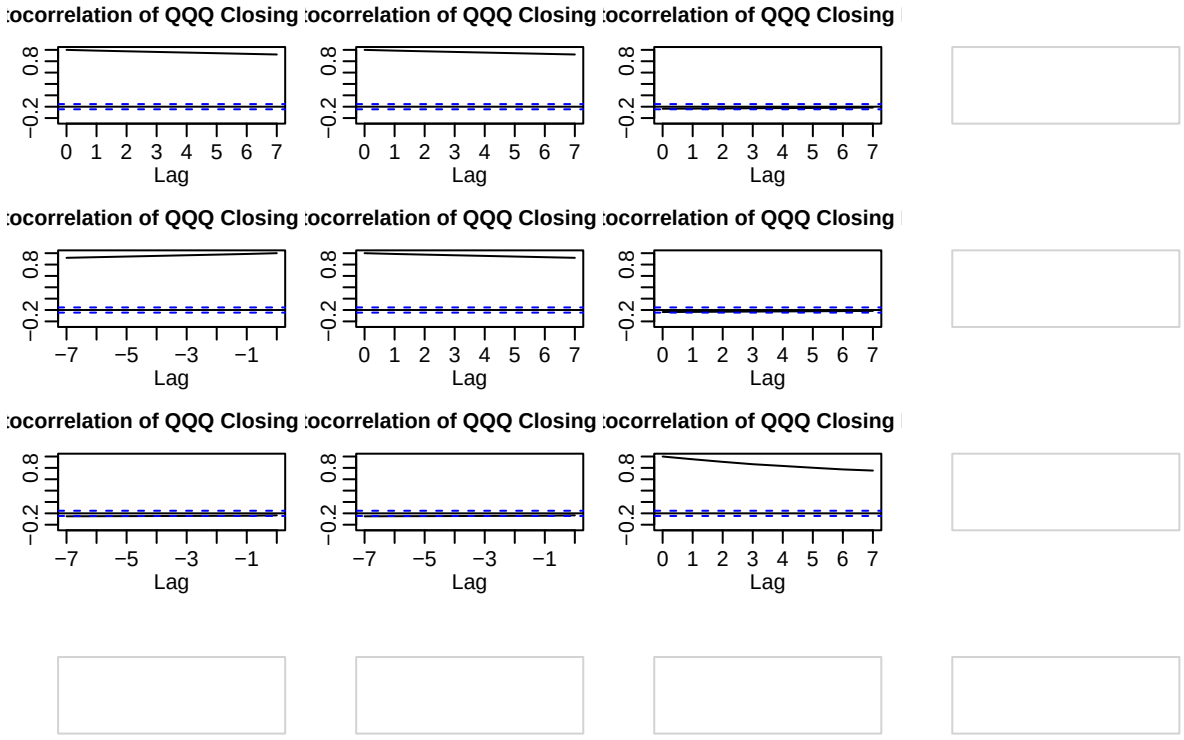
[1 , 1]


:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
:ocorrelation of QQQ Closing
−0.2
0.8
0 1 2 3 4 5 6 7
Lag

[1 , 2]


of QQQ Closing
Autocorrelation
0.8
−0.2
−7
−5
−3
−1
Lag

[2 , 1]



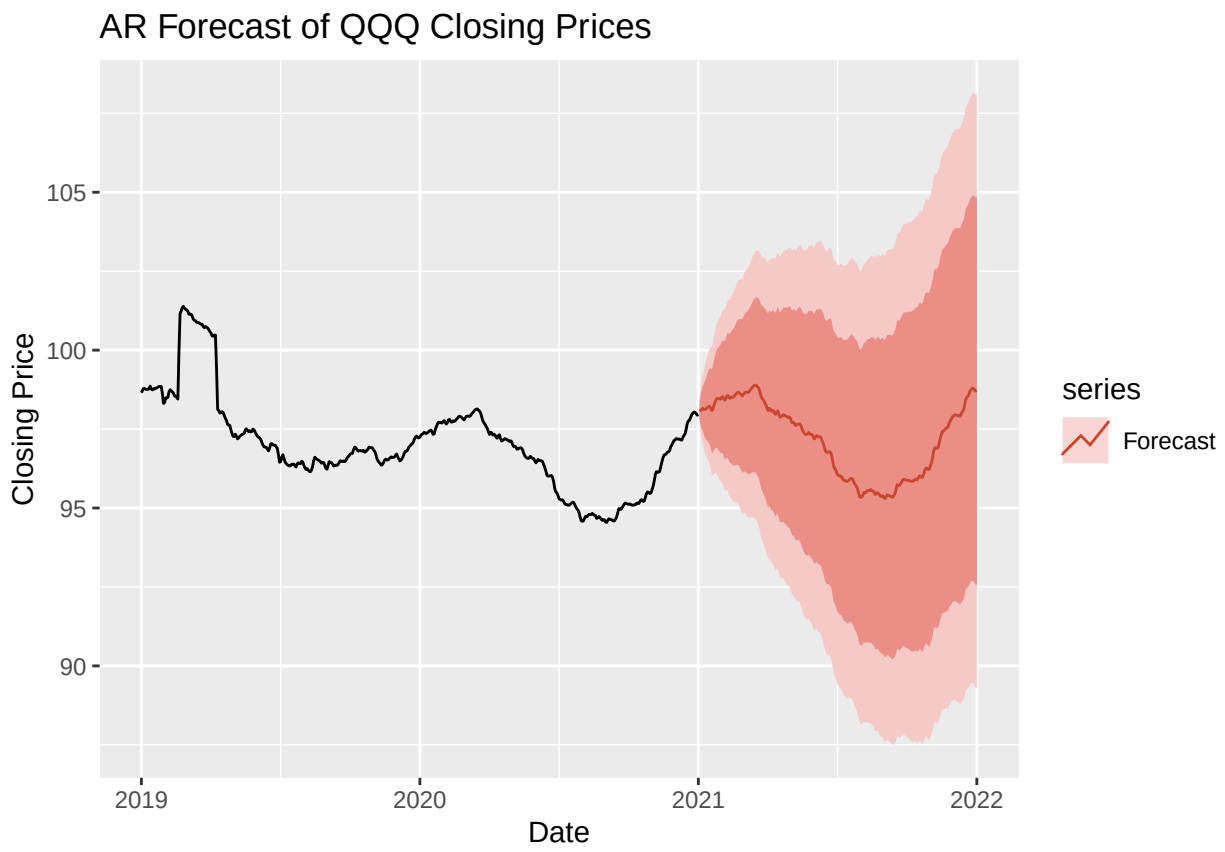
[2 , 2]

Autoregressive (AR) Forecasting of QQQ Closing Prices

```
library(xts)

# Read the CSV file and convert DATE column to Date format
df <- read.csv("Data.csv")
df$DATE <- as.Date(df$Date, format ='%Y-%m-%d')

# Convert "Close" column to numeric
df$Close <- as.numeric(as.character(df$Close))
```

```
## Warning: NAs introduced by coercion
```

```
# Create a time series object
df <- ts(df$Close, start = '2019', end='2021', frequency = 252)

ar_model <- auto.arima(df)

# Forecast one year into the future
forecast_result <- forecast(ar_model, h = 252)

# Plot the original data and the forecast
autoplot(df) +
  autolayer(forecast_result, series = "Forecast") +
  xlab("Date") +
  ylab("Closing Price") +
  ggtitle("AR Forecast of QQQ Closing Prices")
```



ARIMA Forecast of QQQ Closing Price Changes

```
# Load required libraries
library(forecast)
library(ggplot2)

# Read the CSV file and convert DATE column to Date format
df <- read.csv("Data.csv")
df$DATE <- as.Date(df$Date, format ='%Y-%m-%d')


# Convert "Close" column to numeric
df$Close <- as.numeric(as.character(df$Close))
```

```
## Warning: NAs introduced by coercion
```

```
# Create a time series object
df <- ts(df$Close, start = '2019', end='2024', frequency = 252)

# Detrend the time series data using differencing
qqq_detrended <- diff(df)

# Fit an ARIMA model
arima_model <- auto.arima(qqq_detrended)
```

```r
# Print the ARIMA model summary
print(arima_model)
```

```
## Series: qqq_detrended
## ARIMA(0,1,0)
##
## sigma^2 = 0.01089:  log likelihood = 1058.74
## AIC=-2115.47   AICc=-2115.47   BIC=-2110.34
```

```r
# Forecast future changes in the data
forecast_result <- forecast(arima_model, h = 252)

# Plot the forecasted changes
autoplot(forecast_result) +
  xlab("Date") +
  ylab("Change in Closing Price") +
  ggtitle("ARIMA Forecast of QQQ Closing Price Changes")
```