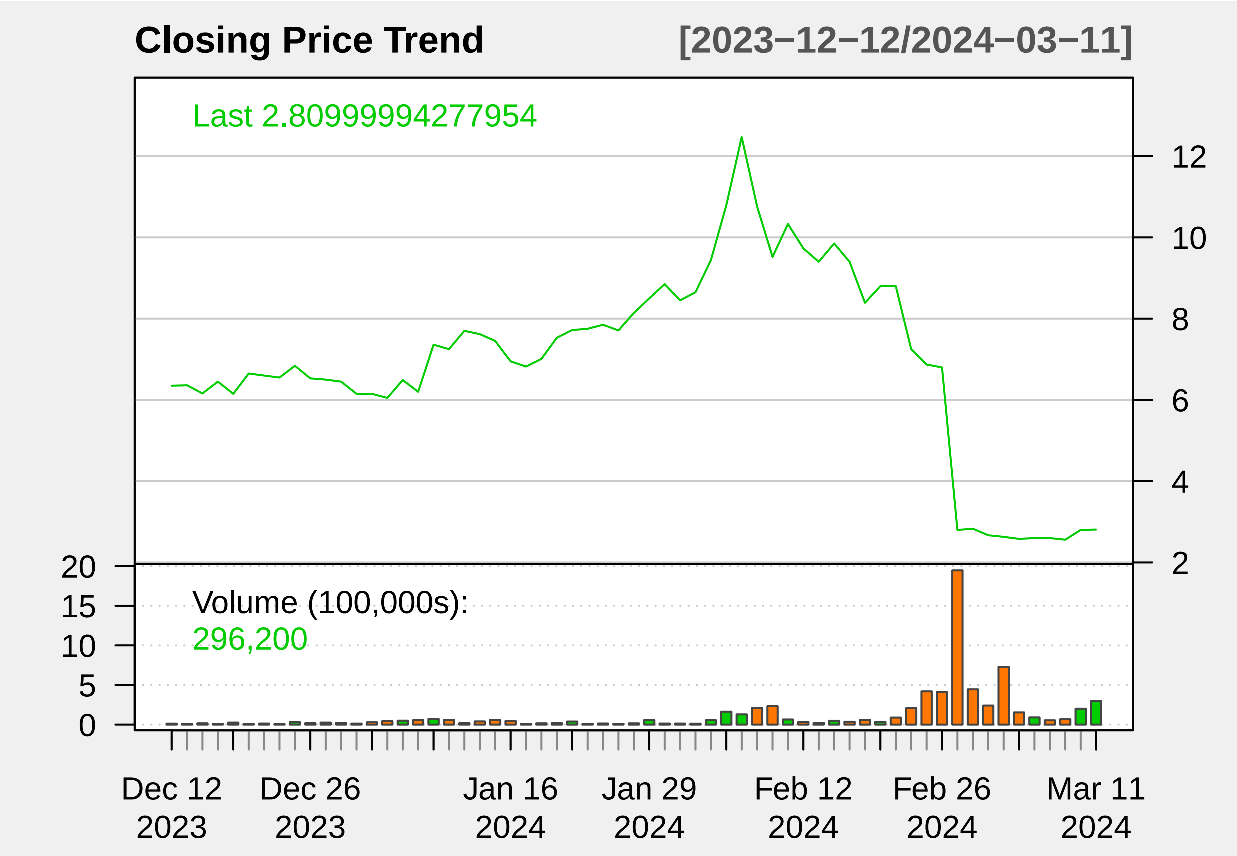
project2

2024-03-19

```r
# Install and load the quantmod package
library(quantmod)
```

```
## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

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

## Loading required package: TTR

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

```r
# Specify the stock symbol and the time period
stock_symbol <- "NERV"
start_date <- "2023-12-12"
end_date <- "2024-03-12"

# Fetch the stock data
NERV_cp_trend <- getSymbols(stock_symbol, from = start_date, to = end_date)

# Plot the closing price trend
chartSeries(NERV, type = "line", theme = "white", name = "Closing Price Trend")
```




```r
# Read the CSV file
NERV <- read.csv("Minerva_Stockinfo.csv")
```

Data Cleaning

```r
NERV$Date <- as.Date(NERV$Date, format = "%m/%d/%y")
# Sort the data by the date column
NERV <- NERV[order(NERV$Date), ]

NERV <- na.omit(NERV)
```

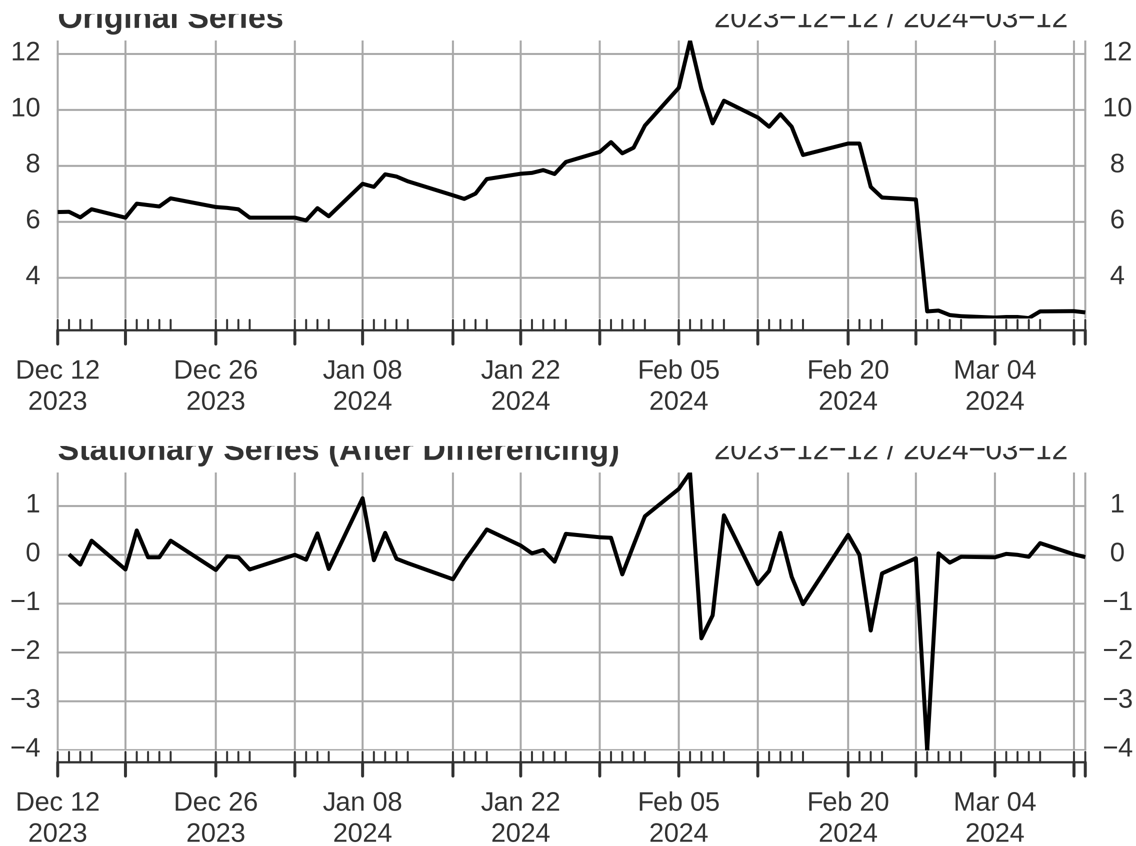
Training

```r
train_NERV <- subset(NERV, Date >= as.Date("2023-12-12") & Date <= as.Date("2024-02-12"))
test_NERV <- subset(NERV, !(Date >= as.Date("2023-12-12") & Date <= as.Date("2024-02-12")))
closing_prices <- xts(NERV$Close, order.by = NERV$Date)
library(tseries)

# Perform the Augmented Dickey-Fuller test
adf_result <- adf.test(closing_prices)
```

```r
# Print the test results
print(adf_result)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  closing_prices
## Dickey-Fuller = -1.2546, Lag order = 3, p-value = 0.8765
## alternative hypothesis: stationary
```

```r
stationary_series <- diff(closing_prices)

# Plot the original and differenced series
par(mfrow=c(2,1))
plot(closing_prices, main="Original Series", type='l')
plot(stationary_series, main="Stationary Series (After Differencing)", type='l')
```




```r
stationary_series <- na.omit(stationary_series)

adf_result_diff <- adf.test(stationary_series)

# Print the test results
print(adf_result_diff)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  stationary_series
## Dickey-Fuller = -3.438, Lag order = 3, p-value = 0.05823
## alternative hypothesis: stationary
```

```
CP_train<- xts(train_NERV$Close, order.by = train_NERV$Date)
CP_test <- xts(test_NERV$Close, order.by = test_NERV$Date)
library(forecast)

start_date <- as.Date(start_date)
time_series <- ts(data = CP_train, start = start_date, frequency = 1)

# Identify the best ARIMA model using auto.arima
arima_model <- auto.arima(time_series)

# Print the identified ARIMA model
print(arima_model)
```

```
## Series: time_series
## ARIMA(0,1,0)
##
## sigma^2 = 0.3573:  log likelihood = -37.08
## AIC=76.16   AICc=76.26   BIC=77.87
```

```
arima_model_1 <- arima(CP_train, order = c(0,1,0))
arima_model_2 <- arima(CP_train, order = c(2,1,2))
summary(arima_model_1)
```

```
##
## Call:
## arima(x = CP_train, order = c(0, 1, 0))
##
##
## sigma^2 estimated as 0.3573:  log likelihood = -37.08,  aic = 76.16
##
## Training set error measures:
##                      ME      RMSE       MAE      MPE     MAPE      MASE
## Training set 0.08062738 0.5905978 0.4025321 0.814106 4.738915 0.9765573
##                     ACF1
## Training set 0.009387741
```

```
summary(arima_model_2)
```

```
##
## Call:
## arima(x = CP_train, order = c(2, 1, 2))
##
## Coefficients:
##          ar1      ar2      ma1     ma2
##       0.7893  -0.4007  -0.7892  0.2398
```

```
## s.e.   0.8424   0.6920   0.9015  0.7741
##
## sigma^2 estimated as 0.3406:  log likelihood = -36.16,  aic = 82.31
##
## Training set error measures:
##                     ME      RMSE       MAE      MPE     MAPE      MASE
## Training set 0.1165558 0.5766574 0.3995788 1.202237 4.721362 0.9693924
##                     ACF1
## Training set -0.0501159
```

```
forecast_length <- nrow(test_NERV)  # Number of rows to forecast
forecasted_values <- forecast(arima_model, h = forecast_length+50)

# Plot the forecast with historical data
plot(forecasted_values)

# Overlay the actual data points on the forecast plot
lines(test_NERV$Date, test_NERV$Close, col = 'red')
```