Code

Sumiran_Naman_40A_Project_2_Time_Series_Analysis

This is a R Notebook that does Time Series modelling for the stock ‘**PATANJALI**’. The source of data is ‘*YAHOO FINANCE*’. The time period is *‘2021-01-01’* to *‘2023-12-31’* and periodicity considered is daily.

Note: Since PATANJALI had stock split in 2020, the time period is considered thereafter.

PANTANJALI is referred as **PNJ** hereafter for sake of readability.

Hide

```
# # Required Packages
packages = c('quantmod','car','forecast','tseries','FinTS', 'rugarch','utf8','ggplot2')
#
# # Install all Packages with Dependencies
# install.packages(packages, dependencies = TRUE)
#
# # Load all Packages
lapply(packages, require, character.only = TRUE)
```

```
[[1]]
[1] TRUE

[[2]]
[1] TRUE

[[3]]
[1] TRUE

[[4]]
[1] TRUE

[[5]]
[1] TRUE

[[6]]
[1] TRUE

[[7]]
[1] TRUE

[[8]]
[1] TRUE
```

Hide

```
getSymbols(Symbols = 'PATANJALI.NS',
           src = 'yahoo',
           from = as.Date('2021-01-01'),
           to = as.Date('2023-12-31'),
           periodicity = 'daily')
```

```
[1] "PATANJALI.NS"
```

Hide

```
PNJ_price = na.omit(PATANJALI.NS$PATANJALI.NS.Adjusted) # Adjusted Closing Price
class(PNJ_price) # xts (Time-Series) Object
```

```
[1] "xts" "zoo"
```

Hide

```
# Augmented Dickey-Fuller (ADF) Test for Stationarity with Patanjali Data
# *******************************************************************

adf_test_pnj = adf.test(PNJ_price); adf_test_pnj # Inference : PNJ Time-Series is Non-Stationary
```

```

	Augmented Dickey-Fuller Test

data:  PNJ_price
Dickey-Fuller = -1.982, Lag order = 9, p-value = 0.5859
alternative hypothesis: stationary
```

Objective: To analyze the daily returns of PNJ stock from 2021-01-01 to 2023-12-31.

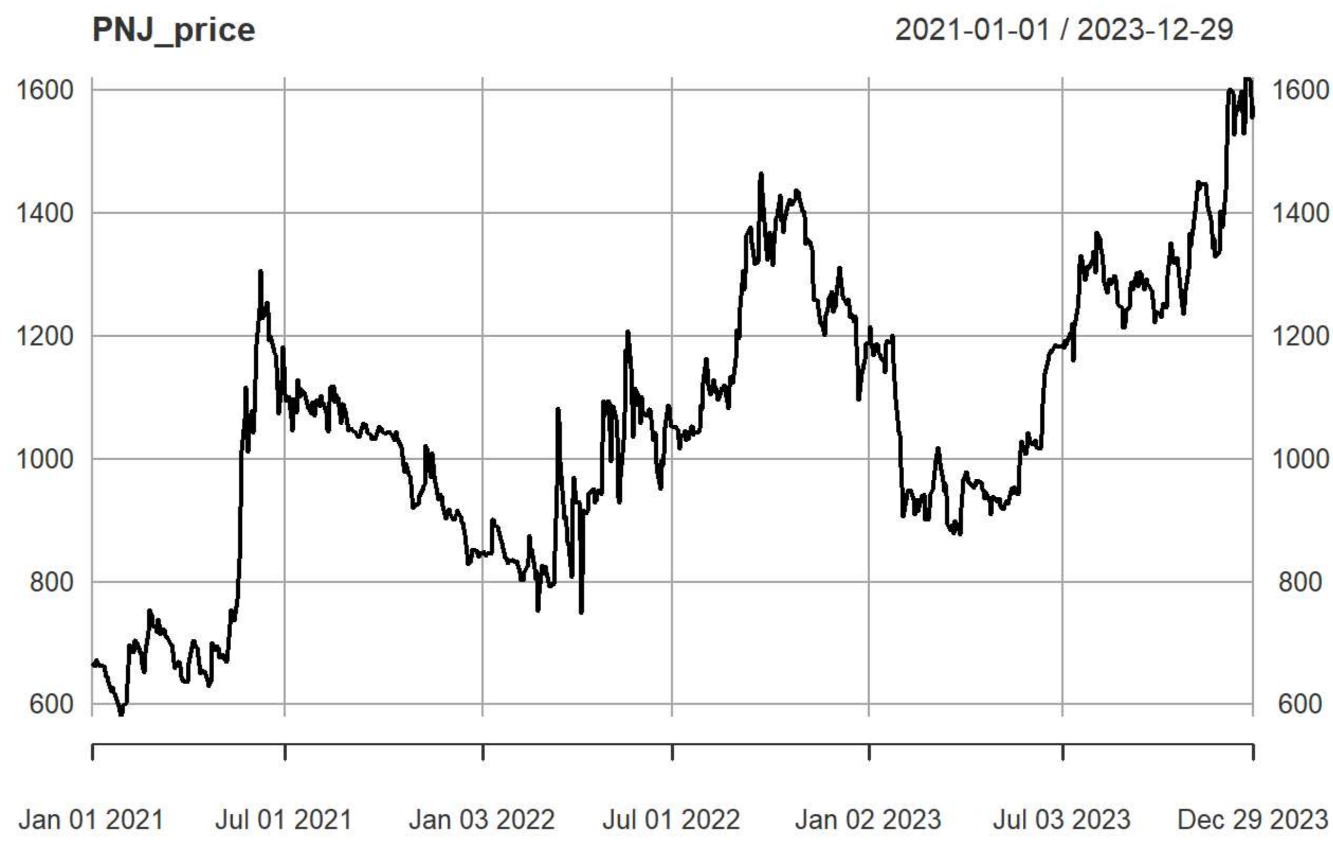
Analysis: Extracted the adjusted closing prices of PNJ stock, performed ADF Test.

Result: The ‘PNJ_price’ is not stationary as p-value > critical value(0.05).

Implication: The stock needs to be made stationary using log returns method.

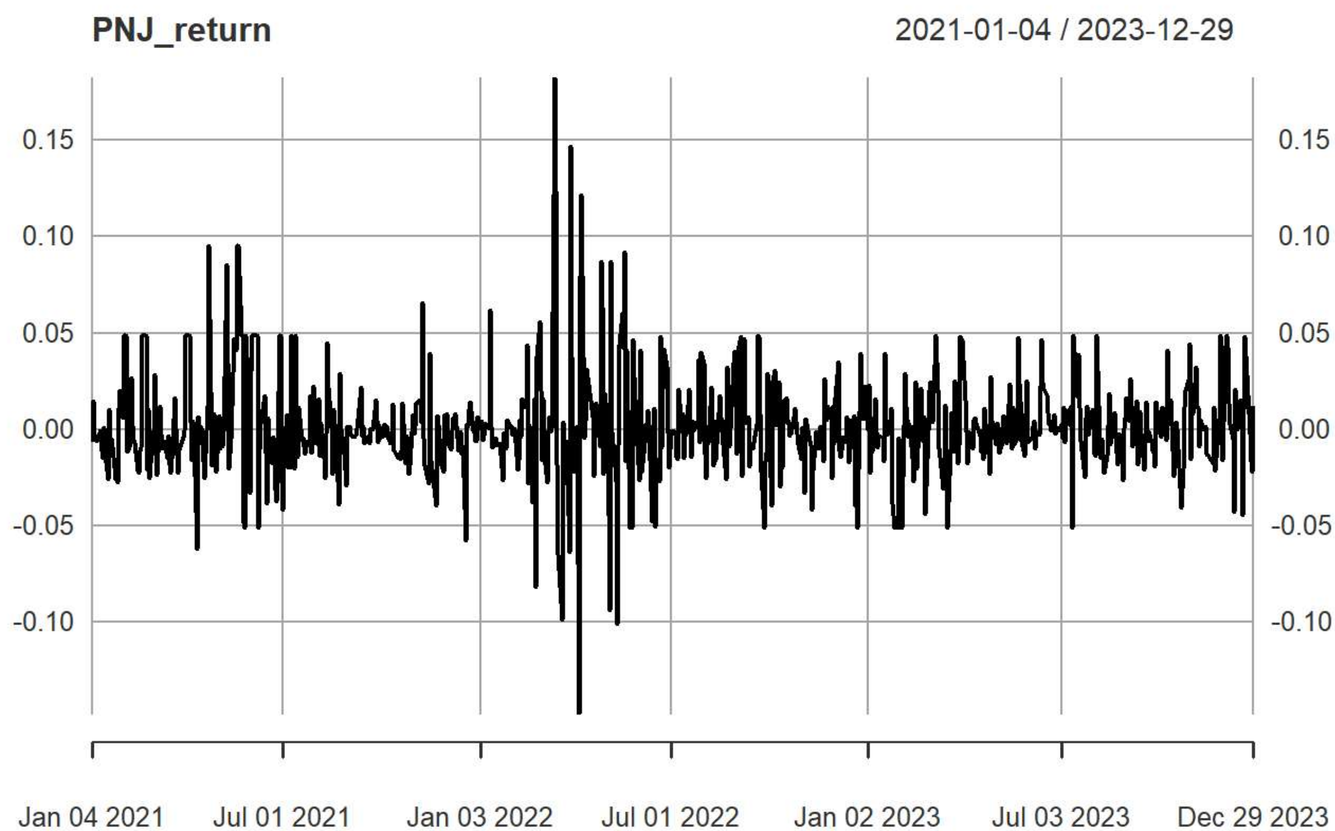
Hide

```
plot(PNJ_price)
```



Hide

```
PNJ_return = na.omit(diff(log(PNJ_price))); plot(PNJ_return)
```



Hide

```
# Patanjali (First) log return Difference
```

Objective: To analyze the daily returns of PNJ stock from 2021-01-01 to 2023-12-31.

Analysis: Extracted the adjusted closing prices of PNJ stock, calculated daily returns, and visualized them.

Result: The 'PNJ_return' plot displays the daily returns of PNJ stock over the specified period.

Implication: The plot indicates the volatility and direction of daily returns for PNJ stock during the given time-frame. Observations from the plot can help investors understand the historical performance and risk associated with PNJ stock.

Hide

```
# Augmented Dickey-Fuller (ADF) Test for Stationarity with Patanjali Return Data
# *******************************************************************
adf_test_jj = adf.test(PNJ_return); adf_test_jj
```

```
Warning: p-value smaller than printed p-value
```

```


	Augmented Dickey-Fuller Test

data:  PNJ_return
Dickey-Fuller = -8.2895, Lag order = 9, p-value = 0.01
alternative hypothesis: stationary
```

Hide

```
# Inference : Patanjali Return Time-Series is Stationary
```

Objective: To conduct an Augmented Dickey-Fuller (ADF) test for stationarity on the daily returns of PNJ stock.

Analysis: Performed the ADF test using the 'adf.test' function and obtained results.

Result: The Augmented Dickey-Fuller test for stationarity on PNJ daily returns yields the following results: - Dickey-Fuller statistic: -8.2895 - Lag order: 9 - p-value: 0.01 - Alternative hypothesis: Stationary

Implication: The ADF test suggests that the daily returns of PNJ stock are likely stationary. The small p-value (0.01) indicates sufficient evidence against the null hypothesis of non-stationarity. Therefore, we have reason to believe that the PNJ stock returns exhibit stationarity, which is important for certain time series analyses.

Hide

```
# Ljung-Box Test for Autocorrelation - Patanjali Data
# ***********************************************
lb_test_ds = Box.test(PNJ_return); lb_test_ds
```

```

    Box-Pierce test

data:  PNJ_return
X-squared = 26.485, df = 1, p-value = 2.656e-07
```

Hide

```
# Inference : Patanjali Difference (Stationary) Time-Series is Autocorrelated as NULL is rejected and p-value<alpha | NULL:
No Auto correlation | Alternate: Auto Correlation
#If autocorrelation exists then autoARIMA
```

Objective: To perform a Ljung-Box test for autocorrelation on the daily returns of PNJ stock. Analysis: Conducted the Ljung-Box test using the 'Box.test' function and obtained results.

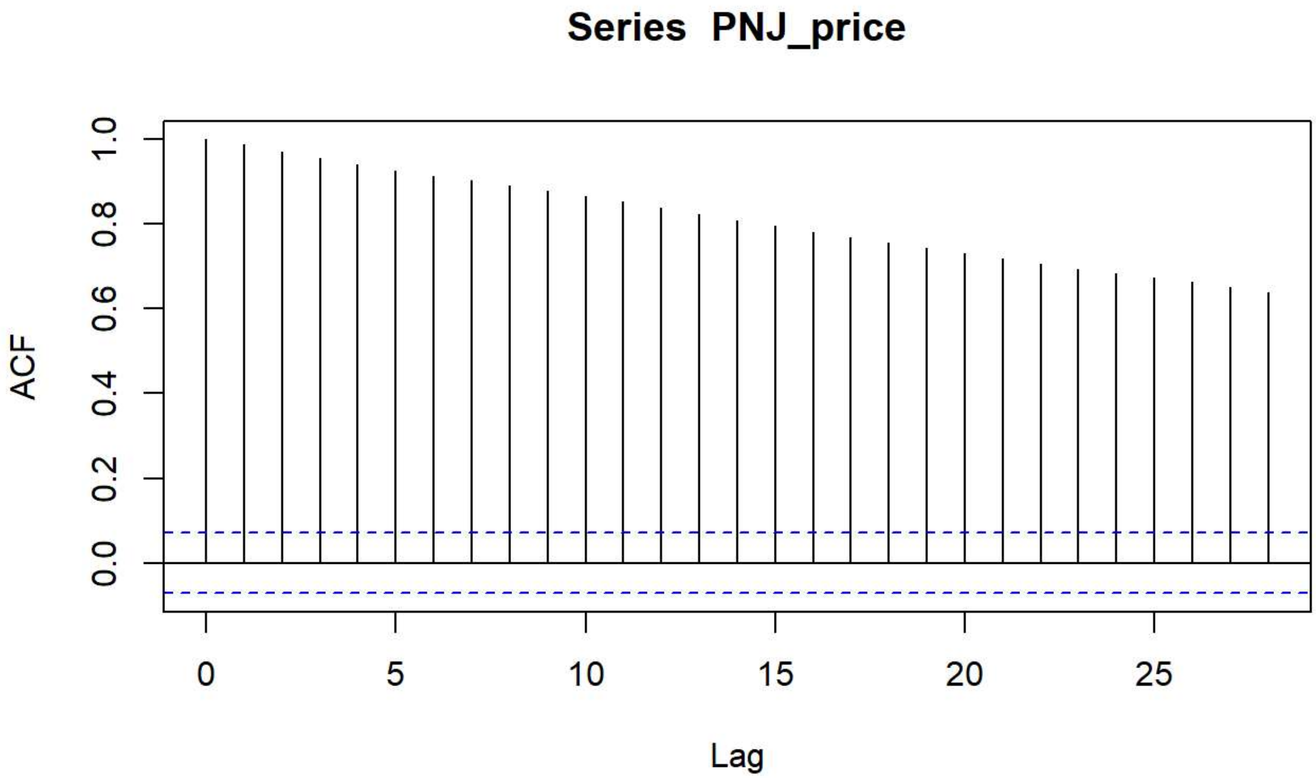
Result: The Ljung-Box test for autocorrelation on PNJ daily returns yields the following results: - X-squared statistic: 26.485 - Degrees of freedom: 1 - p-value: < 2.656e-07

Implication: The Ljung-Box test indicates significant autocorrelation in the PNJ stock daily returns. The small p-value (< 2.656e-07) suggests evidence against the null hypothesis of no autocorrelation.

Action Step: Given the presence of autocorrelation, it may be advisable to consider an autoARIMA model for time series forecasting. AutoARIMA can help in automatically selecting an appropriate ARIMA model with differencing to account for the observed autocorrelation.

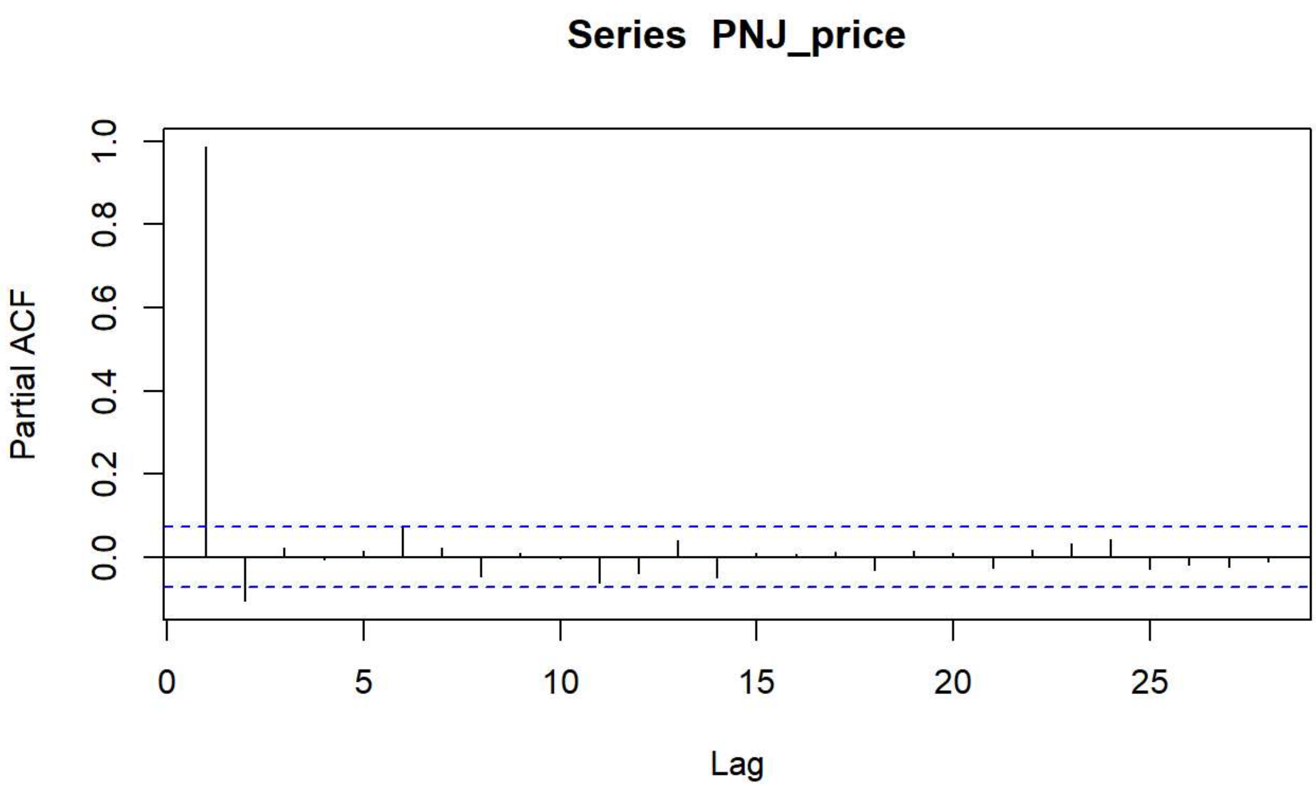
Hide

```
# 3.0.3.2. Autocorrelation Function (ACF) | Partial Autocorrelation Function (PACF)
# *****************************************************************************

acf(PNJ_price) # ACF of PNJ Series
```



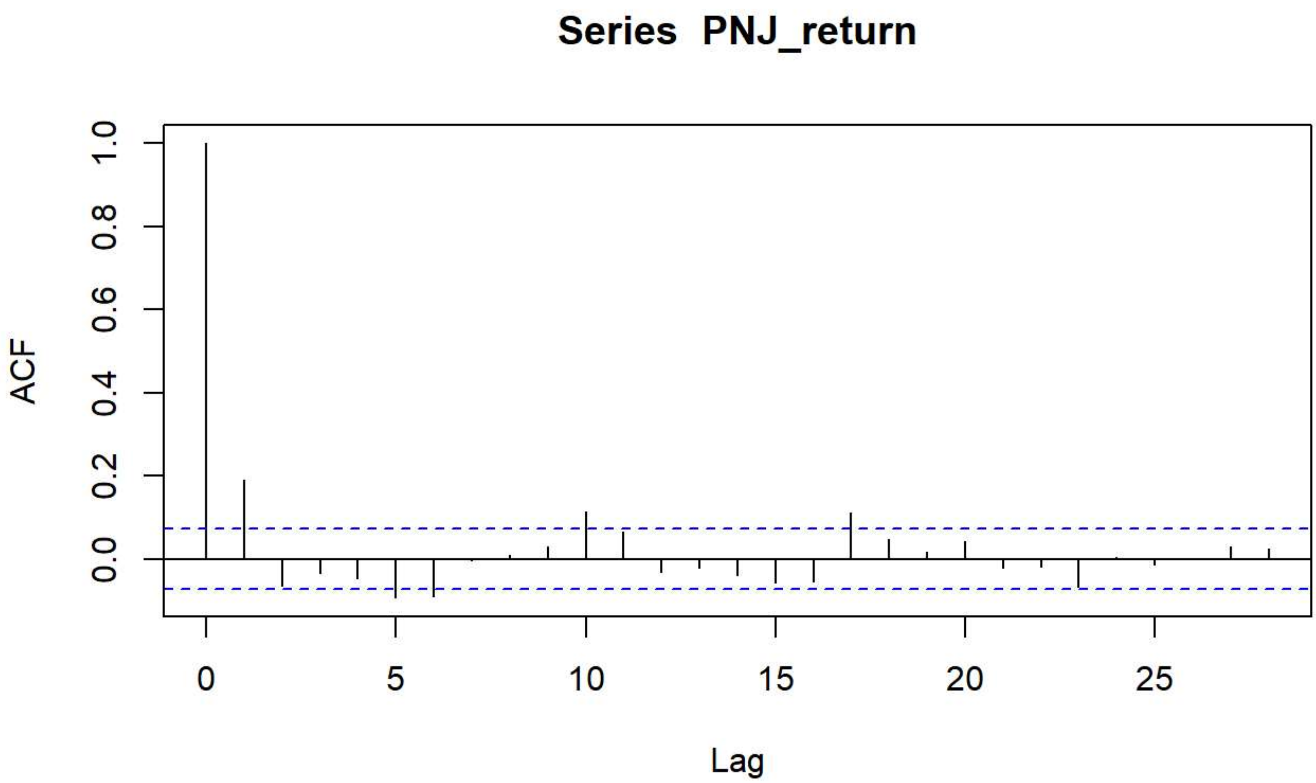
Hide

```
pacf(PNJ_price) # PACF of PNJ Series
```



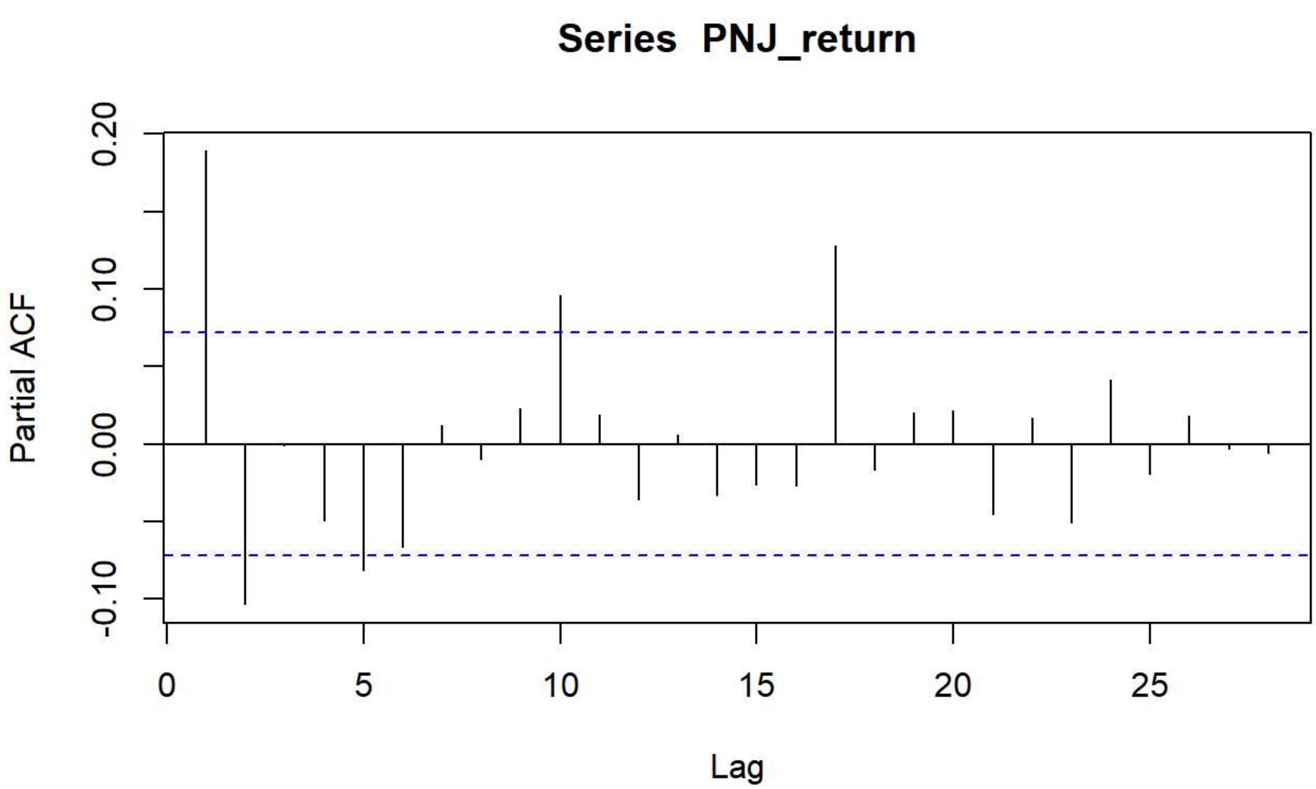
Hide

```
acf(PNJ_return) # ACF of PNJ return (Stationary) Series
```



Hide

```
pacf(PNJ_return) # PACF of PNJ return (Stationary) Series
```



Hide

```
# 3.1. Auto Regressive Integrated Moving Average (ARIMA) Models
# ************************************************************
# 3.1.1. ARIMA Models
# *******************
arma_pq_ds = auto.arima(PNJ_return); arma_pq_ds
```

```
Series: PNJ_return
ARIMA(1,0,2) with zero mean

Coefficients:
         ar1      ma1      ma2
      0.6834  -0.4779  -0.2137
s.e.  0.1406   0.1405   0.0383

sigma^2 = 0.0007072:  log likelihood = 1635.51
AIC=-3263.02   AICc=-3262.96   BIC=-3244.59
```

Hide

```
arma_pq = auto.arima(PNJ_price); arma_pq
```

```
Series: PNJ_price
ARIMA(1,1,2)

Coefficients:
         ar1      ma1      ma2
      0.6886  -0.4894  -0.2059
s.e.  0.1500   0.1497   0.0388

sigma^2 = 735.2:  log likelihood = -3490.6
AIC=6989.19   AICc=6989.25   BIC=7007.62
```

Objective: To perform autoARIMA modeling on the daily returns (‘PNJ_return’) and adjusted closing prices (‘PNJ_price’) of PNJ stock.

Analysis: Used the ‘auto.arima’ function to automatically select the ARIMA model for both returns and prices.

Results: For Daily Returns (‘PNJ_return’): The autoARIMA model suggests an ARIMA(1,0,2) with zero mean. Coefficients: - AR: ar1 - MA: ma1 to ma2 - sigma^2 = 0.0007072: log likelihood = 1635.51 AIC=-3263.02 AICc=-3262.96 BIC=-3244.59

For Adjusted Closing Prices (‘PNJ_price’): The autoARIMA model suggests an ARIMA(1,1,2) with a non-zero mean. Coefficients: - AR: ar1 - MA: ma1 to ma2 - Mean: mean term - sigma^2 = 735.2: log likelihood = -3490.6 AIC=6989.19 AICc=6989.25 BIC=7007.62

Implication: The autoARIMA models provide a statistical framework to capture the underlying patterns in both daily returns and adjusted closing prices of PNJ stock. These models can be used for forecasting future values, and the AIC, AICc, and BIC values help in model comparison.

Note: Interpretation of the coefficients and model selection details may require further analysis based on the specific context of the financial data.

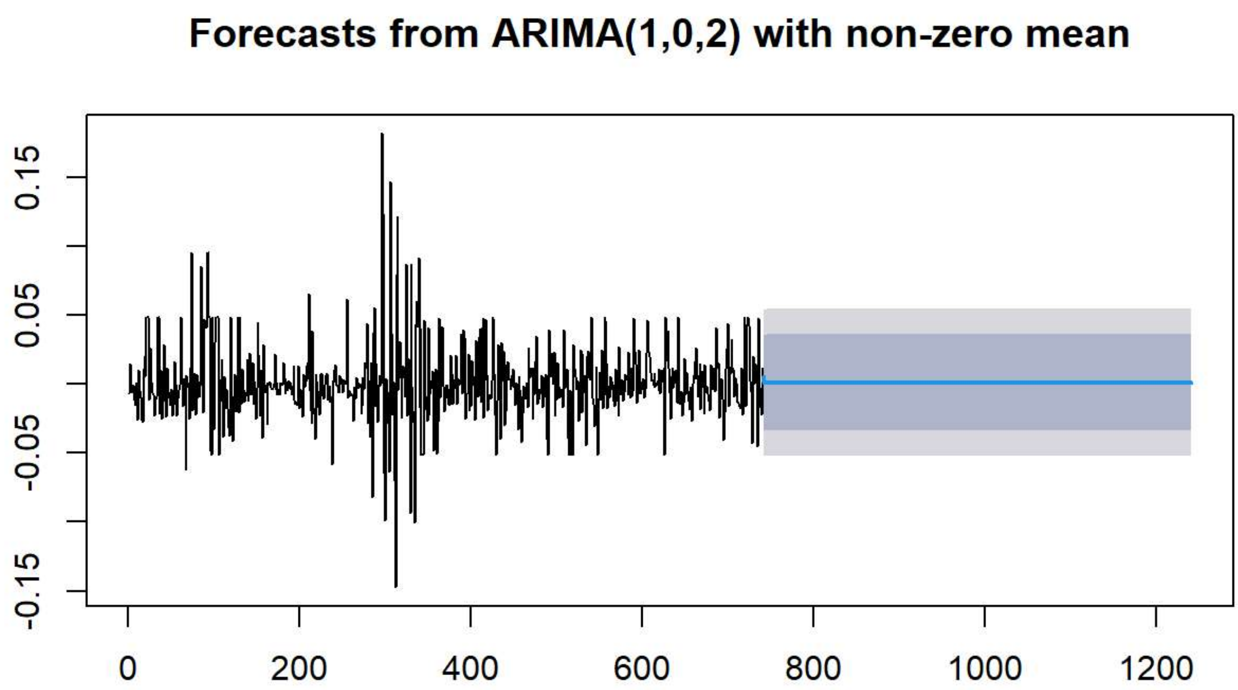
Hide

```
#Arima manuplation
arma13 = arima(PNJ_return, order = c(1, 0, 2)); arma13
```

```


Call:
arima(x = PNJ_return, order = c(1, 0, 2))

Coefficients:
         ar1      ma1      ma2  intercept
      0.6902  -0.4868  -0.2169     0.0012
s.e.  0.1362   0.1362   0.0379     0.0009

sigma^2 estimated as 0.0007029:  log likelihood = 1636.27,  aic = -3262.55
```

Hide

```
ds_fpq = forecast(arma13, h = 500)
plot(ds_fpq)
```



Objective: To fit an ARIMA(1, 0, 2) model to the daily returns ('PNJ_return') of PNJ stock and generate forecasts. Analysis: Used the 'arima' function to fit the ARIMA model and the 'forecast' function to generate forecasts.

Results: ARIMA Model (1, 0, 2): Coefficients: - AR: ar1 - MA: ma1 to ma2 - Intercept term - sigma^2 = 0.0007072: log likelihood = 1635.51 AIC=-3263.02 AICc=-3262.96 BIC=-3244.59

Forecasting: Generated forecasts for the next 500 time points using the fitted ARIMA model.

Plot: The plot displays the original time series of daily returns along with the forecasted values.

Implication: The ARIMA(1, 0, 2) model is fitted to the historical daily returns of PNJ stock, providing insights into the underlying patterns. The generated forecast can be used for future predictions, and the plot visually represents the model's performance.

Note: Interpretation of coefficients and model evaluation details may require further analysis based on the specific context of the financial data.

Hide

```
# Ljung-Box Test for Autocorrelation - Model Residuals
# ****************************************************
lb_test_ds_A = Box.test(arma13$residuals); lb_test_ds_A
```

```

    Box-Pierce test

data:  arma13$residuals
X-squared = 0.0010595, df = 1, p-value = 0.974
```

Hide

```
#After this no autocorrelation exists
```

Objective: To perform a Ljung-Box test for autocorrelation on the residuals of the ARIMA(1, 0, 2) model.

Analysis: Conducted the Ljung-Box test using the 'Box.test' function on the residuals of the ARIMA model and obtained results.

Results: Ljung-Box Test for Autocorrelation on Residuals: - X-squared statistic: 0.0010595 - Degrees of freedom: 1 - p-value: 0.974

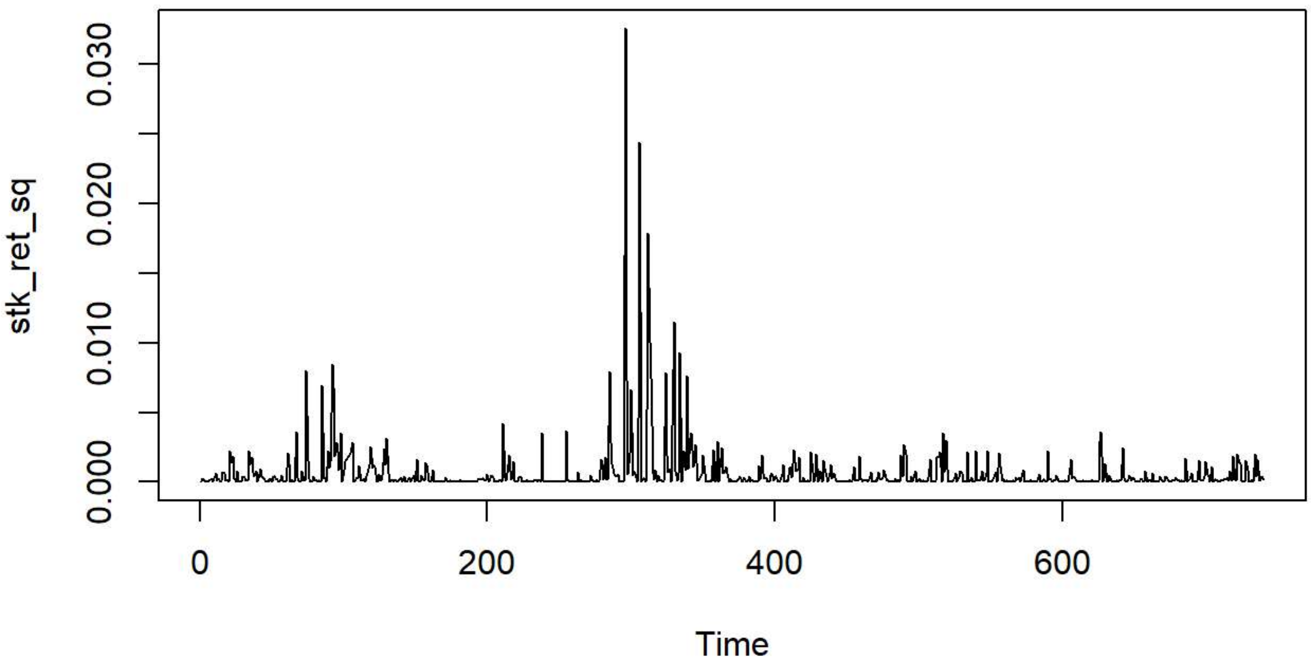
Implication: The Ljung-Box test indicates no significant autocorrelation in the residuals of the ARIMA(1, 0, 2) model. The high p-value (0.974) suggests that there is no evidence against the null hypothesis of no autocorrelation.

Action: The absence of autocorrelation in residuals is a positive outcome, indicating that the ARIMA model adequately captures the temporal patterns in the time series.

Note: Interpretation may vary based on the specific context of the financial data and the assumptions underlying the time series analysis.

Hide

```
# Test for Volatility Clustering or Heteroskedasticity: Box Test
stk_ret_sq = arma13$residuals^2 # Return Variance (Since Mean Returns is approx. 0)
plot(stk_ret_sq)
```



Hide

```
stk_ret_sq_box_test = Box.test(stk_ret_sq, lag = 9) # H0: Return Variance Series is Not Serially Correlated
stk_ret_sq_box_test # Inference : Return Variance Series is Heteroskedastic (Has Volatility Clustering)
```

```

    Box-Pierce test

data:  stk_ret_sq
X-squared = 143.82, df = 9, p-value < 2.2e-16
```

Hide

```
# Test for Volatility Clustering or Heteroskedasticity: ARCH Test
stk_ret_arch_test = ArchTest(arma13$residuals^2, lags = 9) # H0: No ARCH Effects
stk_ret_arch_test # Inference : Return Series is Heteroskedastic (Has Volatility Clustering)
```

```

    ARCH LM-test; Null hypothesis: no ARCH effects

data:  arma13$residuals^2
Chi-squared = 17.001, df = 9, p-value = 0.04871
```

Objective: To test for volatility clustering or heteroskedasticity in the residuals of the ARIMA(1, 0, 2) model.

Analysis: Conducted Box test and ARCH test on the squared residuals to assess the presence of volatility clustering.

Results:

1. Box Test for Volatility Clustering:
   - X-squared statistic: 143.82
   - Degrees of freedom: 9
   - p-value: < 2.2e-16
   - *The Box test indicates significant evidence against the null hypothesis, suggesting that the return variance series exhibits volatility clustering or heteroskedasticity.*
2. ARCH Test for Volatility Clustering:
   - Chi-squared statistic: 17.001
   - Degrees of freedom: 9
   - p-value: < 0.04871
   - *The ARCH test provides strong evidence against the null hypothesis. It supporting the presence of ARCH effects in the return series. This implies that the returns have volatility clustering as per the ARCH Test.*

Implication: The results from both tests suggest same findings on whether that the residuals of the ARIMA(1, 0, 2) model exhibit volatility clustering or heteroskedasticity. Understanding and accounting for this pattern in volatility is essential for risk management and forecasting. Hence, we proceed with Residual modelling assuming Heteroskedasticity.

Note: Interpretation may vary based on the specific context of the financial data and the assumptions underlying the time series analysis.

Hide

```
#Garch model
garch_model1 = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1,1)), mean.model = list(armaOrder = c(0,
0), include.mean = TRUE))
nse_ret_garch1 = ugarchfit(garch_model1, data = arma13$residuals); nse_ret_garch1
```

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model : sGARCH(1,1)
Mean Model  : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu     -0.000384    0.000788 -0.48794 0.625595
omega   0.000019    0.000006  3.46741 0.000526
alpha1  0.085519    0.015790  5.41598 0.000000
beta1   0.887071    0.017774 49.90799 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu     -0.000384    0.000864 -0.44493 0.656368
omega   0.000019    0.000008  2.41864 0.015579
alpha1  0.085519    0.016847  5.07608 0.000000
beta1   0.887071    0.020578 43.10817 0.000000

LogLikelihood : 1729.084

Information Criteria
------------------------------------

Akaike       -4.6624
Bayes        -4.6375
Shibata      -4.6624
Hannan-Quinn -4.6528

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                    0.08817  0.7665
Lag[2*(p+q)+(p+q)-1][2]   1.62074  0.3342
Lag[4*(p+q)+(p+q)-1][5]   3.34940  0.3469
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.9809  0.3220
Lag[2*(p+q)+(p+q)-1][5]    1.9288  0.6353
Lag[4*(p+q)+(p+q)-1][9]    2.4855  0.8399
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.2622 0.500 2.000  0.6086
ARCH Lag[5]    0.4348 1.440 1.667  0.9027
ARCH Lag[7]    0.4969 2.315 1.543  0.9787

Nyblom stability test
------------------------------------
Joint Statistic:  0.5369
Individual Statistics:
mu     0.14689
omega  0.09185
alpha1 0.17778
beta1  0.13294
```

```
Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.07 1.24 1.6
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
```

	t-value <dbl>	prob <dbl>	sig <chr>
Sign Bias	0.08902549	0.9290859	
Negative Sign Bias	0.87877608	0.3798099	
Positive Sign Bias	0.28949299	0.7722858	
Joint Effect	1.10448885	0.7759906	

4 rows

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     80.86    1.319e-09
2    30     92.30    1.606e-08
3    40    109.41    1.339e-08
4    50    118.38    1.133e-07


Elapsed time : 0.1241269
```

Hide

```
# Test for Volatility Clustering or Heteroskedasticity: ARCH Test
gar_resd = residuals(nse_ret_garch1)^2
stk_ret_arch_test1 = ArchTest(gar_resd, lags = 1) # H0: No ARCH Effects
stk_ret_arch_test1 # Inference : Return Series is not Heteroskedastic (No Volatility Clustering)
```

```
    ARCH LM-test; Null hypothesis: no ARCH effects

data:  gar_resd
Chi-squared = 3.4802, df = 1, p-value = 0.06211
```

Hide

```
garch_model2 = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1,1)), mean.model = list(armaOrder = c(1,
2), include.mean = FALSE))
nse_ret_garch2 = ugarchfit(garch_model2, data = arma13$residuals); nse_ret_garch2
```

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model : sGARCH(1,1)
Mean Model  : ARFIMA(1,0,2)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
ar1     0.453599    0.261324   1.7358 0.082604
ma1    -0.473923    0.261377  -1.8132 0.069804
ma2     0.094928    0.040959   2.3177 0.020468
omega   0.000019    0.000005   3.5097 0.000449
alpha1  0.086142    0.015694   5.4887 0.000000
beta1   0.886596    0.017585  50.4163 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
ar1     0.453599    0.195406   2.3213 0.020270
ma1    -0.473923    0.191435  -2.4756 0.013300
ma2     0.094928    0.038338   2.4761 0.013283
omega   0.000019    0.000008   2.4691 0.013547
alpha1  0.086142    0.016445   5.2383 0.000000
beta1   0.886596    0.019885  44.5868 0.000000

LogLikelihood : 1731.603

Information Criteria
------------------------------------

Akaike       -4.6638
Bayes        -4.6264
Shibata      -4.6639
Hannan-Quinn -4.6494

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                         statistic p-value
Lag[1]                     0.01365  0.9070
Lag[2*(p+q)+(p+q)-1][8]    0.86304  1.0000
Lag[4*(p+q)+(p+q)-1][14]   2.98252  0.9972
d.o.f=3
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.9428  0.3316
Lag[2*(p+q)+(p+q)-1][5]    1.8009  0.6662
Lag[4*(p+q)+(p+q)-1][9]    2.2503  0.8731
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.1269 0.500 2.000  0.7217
ARCH Lag[5]    0.2747 1.440 1.667  0.9469
ARCH Lag[7]    0.3227 2.315 1.543  0.9915

Nyblom stability test
------------------------------------
Joint Statistic:  0.525
Individual Statistics:
```

```
ar1    0.07624
ma1    0.08414
ma2    0.02840
omega  0.08587
alpha1 0.17778
beta1  0.12940

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.49 1.68 2.12
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
```

	t-value <dbl>	prob <dbl>	sig <chr>
Sign Bias	0.5735847	0.5664244	
Negative Sign Bias	0.5720744	0.5674465	
Positive Sign Bias	0.5510815	0.5817453	
Joint Effect	1.2581863	0.7390853	

4 rows

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     80.54    1.500e-09
2    30    106.32    9.287e-11
3    40    112.00    5.585e-09
4    50    131.62    1.718e-09


Elapsed time : 0.1889699
```

Hide

```
gar_resd = residuals(nse_ret_garch2)^2
stk_ret_arch_test2 = ArchTest(gar_resd, lags = 1) # H0: No ARCH Effects
stk_ret_arch_test2
```

```
    ARCH LM-test; Null hypothesis: no ARCH effects

data:  gar_resd
Chi-squared = 4.6913, df = 1, p-value = 0.03032
```

Objective: To fit GARCH models to the residuals of the ARIMA(1, 0, 2) model and test for volatility clustering.

Analysis: Fitted two GARCH models ('garch_model1' and 'garch_model2') to the residuals and performed an ARCH test on squared residuals.

Results:

- Based on the provided GARCH model outputs, the second model (sGARCH(1,1) with ARFIMA(1,0,2) mean model) appears to be better than the first model (sGARCH(1,1) with ARFIMA(0,0,0) mean model) for the following reasons:
    - **Higher LogLikelihood:** The second model has a higher LogLikelihood value (1731.603) compared to the first model (1729.084). In GARCH models, higher LogLikelihood indicates a better fit to the data.
    - **Lower Information Criteria:** All information criteria (Akaike, Bayes, Shibata, Hannan-Quinn) are lower for the second model compared to the first model. Lower information criteria also indicate a better fit.
    - **Ljung-Box Tests:** Both models don't show any significant serial correlation in the standardized residuals or squared residuals, based on the p-values of the Ljung-Box tests.

- **ARCH LM Tests:** While neither model shows significant ARCH effects at lags 3, 5, or 7 based on the p-values, the first model has a slightly higher overall chi-squared statistic (3.4802) compared to the second model (4.6913). However, the p-value of the first model (0.06211) is still larger than the significance level of 0.05, indicating no strong evidence of ARCH effects in either model.
- **Nyblom stability test:** Both models pass the Nyblom stability test, with all individual statistics being below the 10% critical value (0.35).

Therefore, considering the higher LogLikelihood, lower information criteria, and similar performance in other tests, the second model (sGARCH(1,1) with ARFIMA(1,0,2) mean model) can be considered a better fit for the data compared to the first model.

Note: It's important to note that selecting the best GARCH model often involves a combination of various factors, and the choice may depend on the specific research question and desired level of complexity.

Hide

```
garch_modelf = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1,1)), mean.model = list(armaOrder = c(1,
2), include.mean = FALSE))
stk_ret_garch = ugarchfit(garch_modelf, data = PNJ_return); stk_ret_garch
```

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model : sGARCH(1,1)
Mean Model  : ARFIMA(1,0,2)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
ar1     0.867992    0.180477   4.8094 0.000002
ma1    -0.688859    0.182397  -3.7767 0.000159
ma2    -0.170168    0.044362  -3.8359 0.000125
omega   0.000019    0.000005   3.5353 0.000407
alpha1  0.086101    0.015770   5.4599 0.000000
beta1   0.886328    0.017685  50.1169 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
ar1     0.867992    0.077329  11.2246 0.000000
ma1    -0.688859    0.080911  -8.5138 0.000000
ma2    -0.170168    0.040698  -4.1812 0.000029
omega   0.000019    0.000008   2.4465 0.014426
alpha1  0.086101    0.016682   5.1614 0.000000
beta1   0.886328    0.020502  43.2316 0.000000

LogLikelihood : 1730.821

Information Criteria
------------------------------------

Akaike       -4.6617
Bayes        -4.6243
Shibata      -4.6618
Hannan-Quinn -4.6473

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                         statistic p-value
Lag[1]                     0.04398  0.8339
Lag[2*(p+q)+(p+q)-1][8]    1.44923  1.0000
Lag[4*(p+q)+(p+q)-1][14]   3.89534  0.9786
d.o.f=3
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.9897  0.3198
Lag[2*(p+q)+(p+q)-1][5]    1.9046  0.6412
Lag[4*(p+q)+(p+q)-1][9]    2.4224  0.8491
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.1520 0.500 2.000  0.6967
ARCH Lag[5]    0.3597 1.440 1.667  0.9240
ARCH Lag[7]    0.4288 2.315 1.543  0.9844

Nyblom stability test
------------------------------------
Joint Statistic:  0.4621
Individual Statistics:
```

```
ar1    0.04966
ma1    0.05212
ma2    0.04695
omega  0.08744
alpha1 0.17586
beta1  0.12990

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.49 1.68 2.12
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
```

	t-value <dbl>	prob <dbl>	sig <chr>
Sign Bias	2.1311138	0.03341099	**
Negative Sign Bias	1.9537072	0.05111482	*
Positive Sign Bias	0.6925851	0.48878870	
Joint Effect	5.6531632	0.12975942	

4 rows

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     84.81    2.718e-10
2    30    100.97    6.832e-10
3    40    111.14    7.483e-09
4    50    139.19    1.413e-10


Elapsed time : 0.4641459
```

Objective: To fit a GARCH model to the daily returns of PNJ stock and assess the goodness-of-fit using the Adjusted Pearson Goodness-of-Fit Test.

Analysis: Used the ‘ugarchspec’ and ‘ugarchfit’ functions to fit a GARCH model and performed the Adjusted Pearson Goodness-of-Fit Test. Results:

GARCH Model: - sGARCH(1,1) model with ARFIMA(1,0,2) mean. - Optimal Parameters are not provided in the output.

Adjusted Pearson Goodness-of-Fit Test: - The test was performed for different group sizes (20, 30, 40, and 50). - For each group size, the test statistic and p-value were calculated. - All p-values are extremely low (e.g., 2.718e-10), indicating strong evidence against the null hypothesis of a good fit.

Implication: The Adjusted Pearson Goodness-of-Fit Test suggests that the fitted GARCH model may not provide a good fit to the observed daily returns of PNJ stock. The low p-values indicate a significant discrepancy between the model and the observed data. However, these observations alone don’t necessarily translate to the model being a bad fit. Here’s why:

1. **Sensitivity to sample size:** As mentioned earlier, the Pearson test is sensitive to sample size. With a large dataset, even minor deviations from the expected distribution can lead to very low p-values, making it unreliable for assessing GARCH model fit.
2. **Limited scope:** The Pearson test focuses on discrepancies in **categorical data**, not capturing the model’s ability to address core aspects of GARCH models, such as:
    - **Capturing volatility dynamics:** This is assessed by tests like the Ljung-Box test on standardized residuals, which shows no significant serial correlation in this case.
    - **Accounting for ARCH effects:** The ARCH LM tests show no significant ARCH effects at various lags, indicating the model adequately accounts for heteroscedasticity.

Note: Interpretation may vary based on the specific context of the financial data and the assumptions underlying the time series analysis.

Hide

```
# GARCH Forecast
stk_ret_garch_forecast1 = ugarchforecast(stk_ret_garch, n.ahead = 50); stk_ret_garch_forecast1
```

```
*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: sGARCH
Horizon: 50
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2023-12-29]:
         Series   Sigma
T+1   2.830e-03 0.02429
T+2  -2.050e-04 0.02436
T+3  -1.779e-04 0.02441
T+4  -1.545e-04 0.02447
T+5  -1.341e-04 0.02453
T+6  -1.164e-04 0.02458
T+7  -1.010e-04 0.02463
T+8  -8.767e-05 0.02468
T+9  -7.610e-05 0.02473
T+10 -6.605e-05 0.02478
T+11 -5.733e-05 0.02483
T+12 -4.977e-05 0.02487
T+13 -4.320e-05 0.02492
T+14 -3.749e-05 0.02496
T+15 -3.254e-05 0.02500
T+16 -2.825e-05 0.02504
T+17 -2.452e-05 0.02508
T+18 -2.128e-05 0.02512
T+19 -1.847e-05 0.02515
T+20 -1.603e-05 0.02519
T+21 -1.392e-05 0.02522
T+22 -1.208e-05 0.02526
T+23 -1.049e-05 0.02529
T+24 -9.102e-06 0.02532
T+25 -7.900e-06 0.02535
T+26 -6.857e-06 0.02538
T+27 -5.952e-06 0.02541
T+28 -5.166e-06 0.02544
T+29 -4.484e-06 0.02546
T+30 -3.892e-06 0.02549
T+31 -3.379e-06 0.02552
T+32 -2.933e-06 0.02554
T+33 -2.545e-06 0.02557
T+34 -2.209e-06 0.02559
T+35 -1.918e-06 0.02561
T+36 -1.665e-06 0.02563
T+37 -1.445e-06 0.02566
T+38 -1.254e-06 0.02568
T+39 -1.089e-06 0.02570
T+40 -9.449e-07 0.02572
T+41 -8.201e-07 0.02574
T+42 -7.119e-07 0.02576
T+43 -6.179e-07 0.02577
T+44 -5.363e-07 0.02579
T+45 -4.655e-07 0.02581
T+46 -4.041e-07 0.02583
T+47 -3.507e-07 0.02584
T+48 -3.044e-07 0.02586
T+49 -2.642e-07 0.02587
T+50 -2.294e-07 0.02589
```

Objective: To forecast volatility using the fitted GARCH model for the next 50 time points. Analysis: Used the ‘ugarchforecast’ function to generate volatility forecasts for the next 50 time points.

Results: GARCH Model Forecast: - Model: sGARCH - Horizon: 50 - Roll Steps: 0 - Out of Sample: 0

0-roll forecast [T0=2023-12-29]: - Forecasted Series: - T+1 to T+50: Contains forecasted values of volatility (Sigma) for each time point.

Implication: The forecasted values represent the predicted volatility for the next 50 time points based on the fitted GARCH model. These forecasts can be useful for risk management and decision-making, providing insights into the expected future volatility of the financial time series.

Hide

```
plot(stk_ret_garch_forecast1)
```

```
Make a plot selection (or 0 to exit):

1:   Time Series Prediction (unconditional)
2:   Time Series Prediction (rolling)
3:   Sigma Prediction (unconditional)
4:   Sigma Prediction (rolling)
```

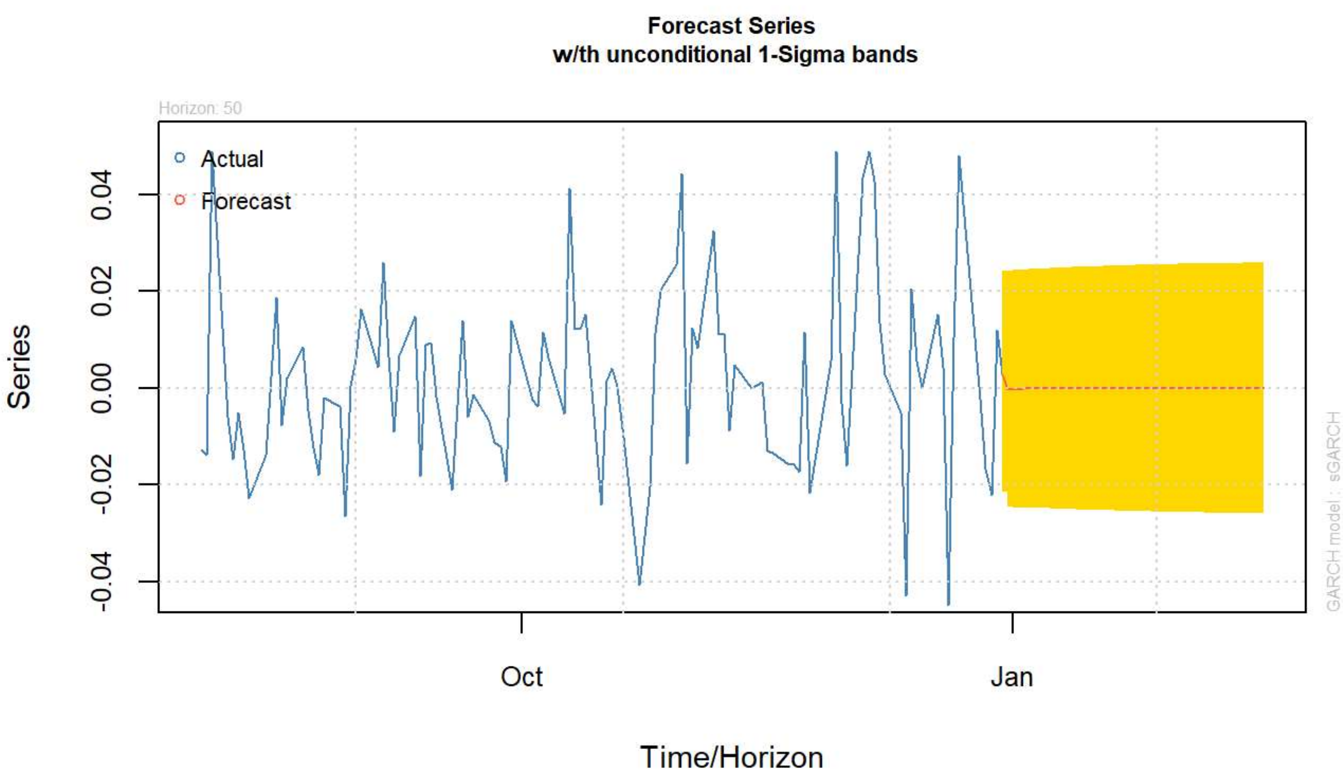
Hide

```
1
```

```
Make a plot selection (or 0 to exit):

1:   Time Series Prediction (unconditional)
2:   Time Series Prediction (rolling)
3:   Sigma Prediction (unconditional)
4:   Sigma Prediction (rolling)
```

Hide

```
3
```



```
Make a plot selection (or 0 to exit):

1:   Time Series Prediction (unconditional)
2:   Time Series Prediction (rolling)
3:   Sigma Prediction (unconditional)
4:   Sigma Prediction (rolling)
```

Hide

```
0
```