STATISTICAL TECHNIQUES AND TIME SERIES

Contents

[1. Downloading Data 3](#_Toc89810374)

[2. Transforming the series to log returns. 3](#_Toc89810375)

[3. ACF and PACF function 3](#_Toc89810376)

[4. Ljung-Box Test 4](#_Toc89810377)

[5. Stationarity Testing 4](#_Toc89810378)

[6. Normality testing: 4](#_Toc89810379)

[7. ARIMA Model Fitting and Lag order Determination 5](#_Toc89810380)

[8. Coefficients for the ARIMA(1,0,0) 6](#_Toc89810381)

[9. Residuals from an ARIMA fit 6](#_Toc89810382)

Table of figures

[Figure 1: ACF PLOT 4](file:///C:\Users\POORVAA\Documents\STAT_CW.docx#_Toc89811540)

[Figure 2: PACF plot 4](#_Toc89811541)

[Figure 3: Residuals plot 6](#_Toc89811542)

In this report, S&P500 index price data covering the period 2009/01/01 to 2020/12/32 (yyyy/mm/dd) is downloaded and includes the processing and analyzing based on this data.

# Downloading Data

Code: library(quantmod)

getSymbols("^GSPC", from ="2009-01-01", to="2020-12-31")

The ‘Quantmod’ is a package for R. It stands for ‘Quantitative Financial Modelling Framework’. It helps to download the data using ‘getSymbols()’, charting using chartSeries(), and allows us to work on data using technical indicators like moving average, Bollinger band, momentum, ROC, MACD, RSI, etc.

# Transforming the series to log returns.

Before any forecasting it is essential to make sure that the time series data is stationary. Stationary time series has a constant mean and variance over time. This stationary TS is time independent. In stationary time series its easier to predict future values because there is no underlining trends or seasonality.

Differencing the data is one way to make the data stationary. Differencing stabilizes the mean while logarithmic transformations help to stabilize the variance.

Diff(log()) reduces the trend and seasonality in timeseries.

Code: ret\_a = quantmod::periodReturn(GSPC$GSPC.Adjusted,period="daily", type="log")

OR

returns = diff(log(GSPC$GSPC.Adjusted))

Diff() computes the successive differences of adjusted price i.e. (x2-x1), (x3-x2), (x4-x3), and so on.

Diff(Log(ts)) simply calculates [ log(Xt)-log(Xt-1) ]

# ACF and PACF function

ACF stands for Auto-correlation function and PACF stands for Partial Correlation function. ACF describes us how the present values of series are related with its lagged values or past values. ACF also considers time series components like trend, seasonality and residuals to find correlations. While PACF helps to find the correlation of the residuals. Residuals are the remaining data after removing all the trends and seasonality. PACF skips the correlation already found by ACF using present values and lag values and instead find the correlation of the residuals to reveal the hidden information which can be modeled by next lag.

Code: acf(ret\_a)

pacf(ret\_a)

sample autocorrelations of order k=0, 1, 2,...n can be obtained by computing the following expression with the observed series yt, where t=1, 2,...n and is mean. ρ(Rho)denotes the autocorrelation function.

ρ(k)=

Partial auto correlation can be obtained by

Where, = original series yt - sample mean

Chart, histogram

Description automatically generated

Chart, box and whisker chart

Description automatically generated

Figure 1: ACF PLOT

Figure 2: PACF plot

# Ljung-Box Test

Code: Box.test(ret\_a, type = "Ljung-Box")

Output: > Box.test(ret\_a, type = "Ljung-Box")

Box-Ljung test

data: ret\_a

X-squared = 64.816, df = 1, **p-value = 7.772e-16**

Provided output includes a p-value. If the value is less than or equal to 0.05 then the data contains significant autocorrelation while p-value exceeding 0.05 then it provides no such evidence.

Since the output indicates p-value as 7.772e-16, which is far less than 0.05 it simply means the time series data provides evidence of autocorrelation.

# Stationarity Testing

Code: tseries::adf.test(ret\_a)

Stationarity time series is a time series which has constant or a stable mean and variance over time. Stationarity identification is carried out by ADF testing. ADF stands for Augmented Dickey-Fuller test. This test involves hypothesis testing which signifies null hypothesis as ‘not a stationary data’ and Alternate hypothesis as a ‘stationary data’. ADF test detrends the data and recenters the data around mean equals to 0. By using adf function of tseries package the output provided following: P-value is 0.01 which suggests that data is stationary and mean reverting while and the model rejects the hypothesis.

Output: Augmented Dickey-Fuller Test

data: ret\_a Dickey-Fuller = -15.41, Lag order = 14, p-value = 0.01

alternative hypothesis: stationary

# Normality testing:

Normality testing helps us to understand that if the timeseries data is normally distributed or not. Package ‘nortest’ contains testing functions as follows:

Lilliefors test, Anderson-Darling test, Shapiro-Francia test, Cramer-von Mises test and Pearson chi-squared test.

If p 0.05, then the data population is not normally distributed. Otherwise it provides no such evidence.

The null hypothesis of these tests initially suggests that the data is normally distributed until proven.

|  |  |  |
| --- | --- | --- |
| > sf.test(ret\_a) | > lillie.test((ret\_a)) | >ad.test(as.numeric((ret\_a))) |
| Shapiro-Francia normality test | Lilliefors (Kolmogorov-Smirnov) normality test | Anderson-Darling normality test |
| data: ret\_a | data: (ret\_a) | data: as.numeric((ret\_a)) |
| W = 4.8722e-05 | D = 0.99587 | A = 77.113, |
| p-value < 2.2e-16 | p-value < 2.2e-16 | p-value < 2.2e-16 |

As per the output of the tests p-value is 2.2e-16 (0**.**00..02), so the data population is most probably not normally distributed.

# ARIMA Model Fitting and Lag order Determination

To make a time series model stationary we first difference(**d**) it. After differencing we add AR(Auto regressive model) and MA (Moving Average model). AR term is indicated using **P**. P refers to the number of lags of Y to use as predictors. Whereas MA term is indicated using **q**. It refers to the number of lagged forecast errors that should go into the ARIMA model.

Equation for ARIMA model is as follows:

Yt = α

Where α – constant

– Linear combination of Lags of Y upto P

– Linear combination of lagged forecast errors

Better fitting ARIMA model has comparatively less AIC and Sigma values.

According to the output we received through auto.arima(), lag order coefficients are (2,0,0)

As our model is already stationary auto.arima() funtion did not require to difference it. So the value of d and q are 0. Differencing is needed only when the timeseries data is non-stationary.

**Output:** Series: ret\_a

|  |  |
| --- | --- |
| Code:  **GSPC\_arm1 <- auto.arima(ret\_a)** | Code:  **GSPC\_ARM100 <- arima(ret\_a, order = c(1,0,0))** |
| ARIMA(2,0,0) with non-zero mean | arima(x = ret\_a, order = c(1, 0, 0)) |
| Coefficients:  ar1 ar2 mean  -0.1356 0.0733 5e-04  s.e. 0.0181 0.0181 2e-04 | Coefficients:  ar1 intercept  -0.1464 5e-04  s.e. 0.0180 2e-04 |
| sigma^2 estimated as 0.0001329:  log likelihood=9194.55 | sigma^2 estimated as 0.0001335: log likelihood = 9186.41, |
| AIC=-18381.11 AICc=-18381.09 BIC=-18357.05 | aic = -18366.82  (observed Value of AIC is lesser) |

As the auto.arima() suggests lag order 2 i.e. (2,0,0) and after trying multiple combinations, Lag order 1 i.e. (1,0,0) to be the best one. As per the tests on auto correlations and stationary, this time series model is stationary and shows evidence of auto correlations. So ARIMA(1,0,0) will fit efficiently. ARIMA(1,0,0) is also called as first-order auto regressive model.

# Coefficients for the ARIMA(1,0,0)

Code: GSPC\_ARM100$coef

The forecasting equations in this case is:

Yt = α = > Yt = 5e-04

Where α is intercept and indicates lag order.

# Residuals from an ARIMA fit

Code: checkresiduals(GSPC\_ARM100)

Output:

Ljung-Box test

data: Residuals from ARIMA(1,0,0) with non-zero mean

Q\* = 83.699, df = 8, p-value = 8.771e-15

Model df: 2. Total lags used: 10

Chart, box and whisker chart

Description automatically generated with medium confidence

Figure 3: Residuals plot

Autocovariance measures linear dependence between present values and lagged values of the same series. The mean of the residuals is close to zero but not zero. It means some information left in residuals which can be used for forecasting.The ACF plot shows that all autocorrelations are partially within the limits and there is no significant correlation in the residual series. Residuals seems symmetric based on its shape like bell.

# Bibliography

Anon., n.d. *Machine learning plus.* [Online]   
Available at: https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/  
[Accessed 06 December 2021].

Nau, R., 2021. *Statistical Forecasting: notes on regression and time series analysis.* [Online]   
Available at: https://people.duke.edu/~rnau/411arim.htm  
[Accessed 06 December 2021].

Otexts, n.d. *Forecasting principles and practice.* [Online]   
Available at: https://otexts.com/fpp2/stationarity.html  
[Accessed 06 december 2021].