**Time Series Analysis: Homework 4** (20232863 Keywoong Bae)

1. (related with HW#3, prob. 4) Estimated the parameters of your identified model in HW#3, prob. 4. Also perform Ljung and Box test for the residuals.

You can see the detailed process in Rcode.

- Estimated parameters

| | AR1 | intercept |
| --- | --- | --- |
| | 0.4342 | 35.2418 |
| s.e. | 0.0719 | 0.9652 |

- Box-Ljung test.

X-squared = 10.754, df = 10, p-value = 0.3717

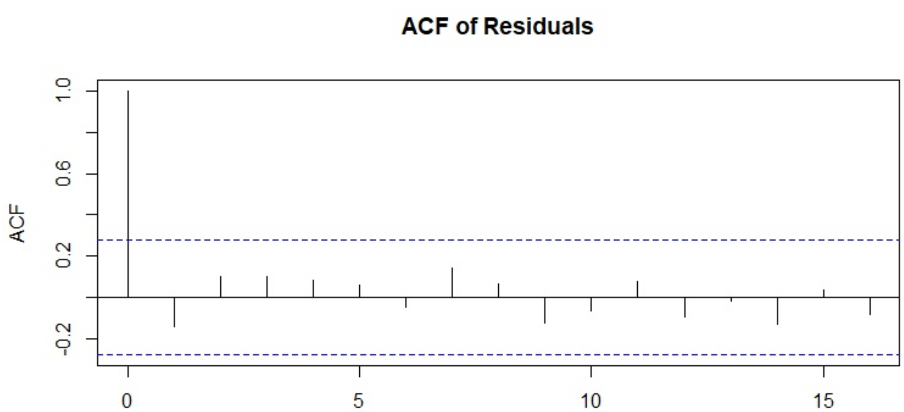
2. Identify the model for J08 series and estimate the related parameters. Perform diagnostic checking for the residuals.

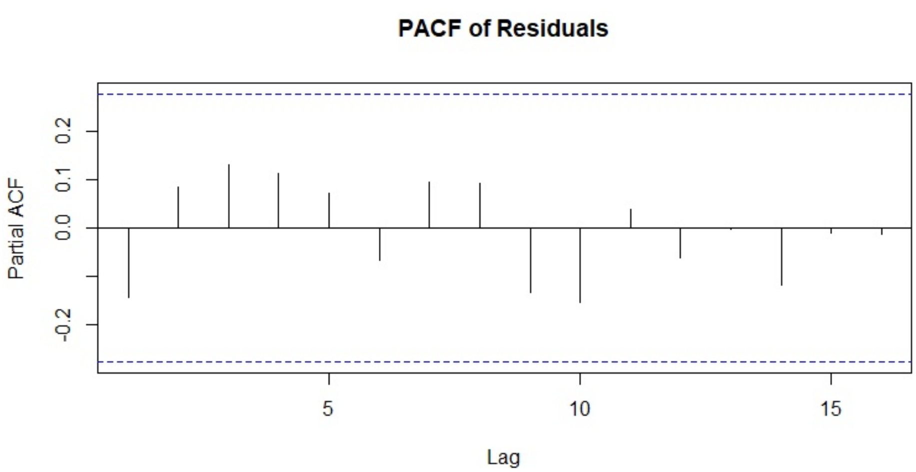
· AR(1) 모형.

| | ar1 | intercept |
|---|---|---|
| | 0.4433 | 18.7317 |
| s.e. | 0.1349 | 1.5904 |

· Diagnostic checking for residuals.

$\chi^2 = 5.7739$, $df = 10$, p-value $= 0.8339$.



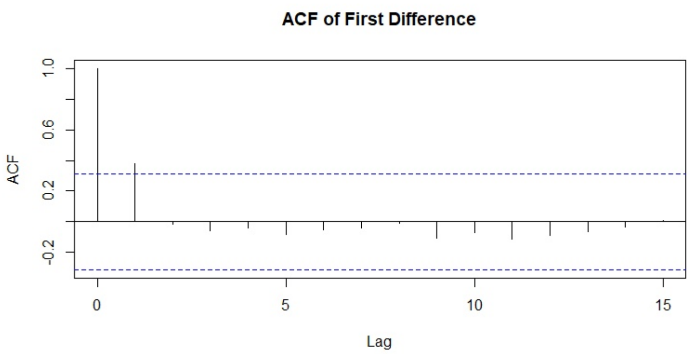
**Time Series Analysis: Homework 4** (20232863 Keywoong Bae)

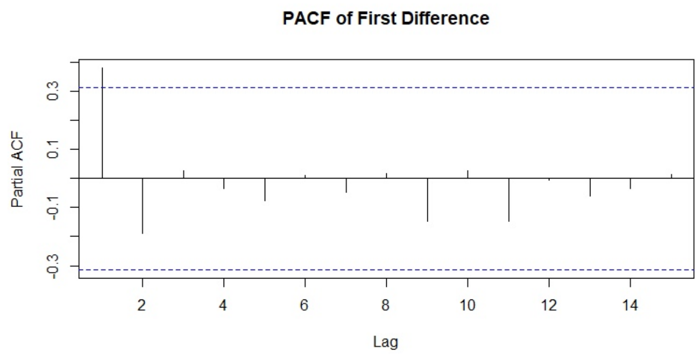
3. Consider J05 series.

(a) Obtain ACF and PACF of the first differenced series.

(b) Estimate the following three models and suggest the best model.

AR(1), MA(1), ARMA(1,1)




```
> print(summary(ar1_model))

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

Coefficients:
         ar1  intercept
      0.3762    28.2201
s.e.  0.1465    26.9828

sigma^2 estimated as 11386:  log likelihood = -237.55,  aic = 481.1

Training set error measures:
                     ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set -0.9571547 106.7072 71.45644 700.6397 867.1336 0.8344099 0.07266774
```

```
> print(summary(ma1_model))

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

Coefficients:
         ma1  intercept
      0.4554    25.6357
s.e.  0.1474    24.2741

sigma^2 estimated as 10985:  log likelihood = -236.89,  aic = 479.78

Training set error measures:
                    ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 0.4376077 104.8076 68.65417 635.1649 827.1112 0.8016872 0.003665733
```

```
> print(summary(arma11_model))

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

Coefficients:
         ar1     ma1  intercept
      0.0083  0.4487    25.6524
s.e.  0.3356  0.3109    24.3797

sigma^2 estimated as 10984:  log likelihood = -236.89,  aic = 481.77

Training set error measures:
                    ME    RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 0.4250506 104.807 68.65703 635.6779 827.3483 0.8017206 0.002079242
```

**Time Series Analysis: Homework 4** (20232863 Keywoong Bae)

4. Consider an ARMA(1,3) model.

(a) Calculate one-step-ahead forecast at time n.

(b) Compute its variance of forecast error in (a).

(c) Derive an updating formula for the one-step-ahead forecast using minimum information.

(a)

```
> print(summary(arma13_model))

Call:
arima(x = timeseries, order = c(1, 0, 3))

Coefficients:
         ar1     ma1     ma2      ma3  intercept
      0.9301  0.5443  0.0422  -0.0374   656.6192
s.e.  0.0646  0.1777  0.2090   0.1805   284.5703

sigma^2 estimated as 10911:  log likelihood = -244.28,  aic = 500.57

Training set error measures:
                   ME     RMSE      MAE       MPE     MAPE      MASE       ACF1
Training set 11.22328 104.4566 68.48153 -1.222137 12.44121 0.8300507 -0.0363478
```

(b)

```
> print(forecast_n$pred)
Time Series:
Start = 41
End = 41
Frequency = 1
[1] 1206.545
```

(c)

```
> forecast_error_variance <- forecast_n$se^2
> print(forecast_error_variance)
Time Series:
Start = 41
End = 41
Frequency = 1
[1] 10911.17
```

5. Consider the series J14. Assume that it is identified as MA(2).
(a) Estimate the model using the first 200 observations.
(b) Obtain the one-step-ahead forecasts for the period thereafter and compare the actual values to compute forecast errors.
(c) Find the forecast performance measures, RMSE, MAD and MAPE for the forecasts in (b).

```
> ma2_model <- arima(timeseries[1:200], order = c(0, 0, 2))
> print(summary(ma2_model))

Call:
arima(x = timeseries[1:200], order = c(0, 0, 2))

Coefficients:
          ma1     ma2  intercept
      -0.6262  0.3637    -0.0490
s.e.   0.0631  0.0717     0.0341

sigma^2 estimated as 0.4279:  log likelihood = -199.15,  aic = 406.3

Training set error measures:
                     ME      RMSE       MAE      MPE     MAPE      MASE       ACF1
Training set 0.00205375 0.6541054 0.5275271 14.74782 417.3699 0.4778032 0.01010519
```

```
> # Obtain one-step-ahead forecasts for the period thereafter
> forecast_values <- predict(ma2_model, n.ahead = length(timeseries) - 200)$pred
> # Compare actual values to forecast values
> actual_values <- timeseries[201:length(timeseries)]
> forecast_errors <- actual_values - forecast_values
> actual_values
 [1]  0.06519868 -0.29935161  0.17358436  0.37036815  1.00027248 -2.24527957  0.19289826
 [8]  0.25045363 -0.31401811  0.67796408 -0.02159384 -0.56804637 -0.51660822  1.12140768
[15] -0.67298549  0.68307552  0.05529077  0.18348220  0.63844074  0.53522552 -0.15087304
[22]  0.06817626  0.92361101 -0.58433536  0.16357980 -0.54791505  0.48658575 -1.21916575
[29]  0.23942887 -0.39678418 -0.15273543  1.07147491 -2.75890970  1.57477204 -1.85797150
[36] -0.48410827 -0.56417613  1.42925889 -0.68869806  0.71376205 -1.19159542  1.89932158
[43] -0.44632788 -0.57541256  1.35933652  0.08133858 -0.20105248 -0.68060908  1.14239704
[50] -0.12708151
> forecast_errors
Time Series:
Start = 201
End = 250
Frequency = 1
 [1]  0.27468394 -0.28726983  0.22263324  0.41941703  1.04932136 -2.19623069  0.24194713
 [8]  0.29950251 -0.26496923  0.72701295  0.02745504 -0.51899749 -0.46755934  1.17045656
[15] -0.62393661  0.73212440  0.10433965  0.23253108  0.68748961  0.58427440 -0.10182416
[22]  0.11722514  0.97265989 -0.53528648  0.21262868 -0.49886618  0.53563463 -1.17011687
[29]  0.28847775 -0.34773530 -0.10368656  1.12052378 -2.70986083  1.62382092 -1.80892262
[36] -0.43505939 -0.51512725  1.47830777 -0.63964919  0.76281093 -1.14254654  1.94837046
[43] -0.39727900 -0.52636369  1.40838540  0.13038746 -0.15200360 -0.63156020  1.19144592
[50] -0.07803263
```

```
> # Compute Root Mean Squared Error (RMSE)
> RMSE <- sqrt(mean(forecast_errors^2))
> # Compute Mean Absolute Deviation (MAD)
> MAD <- mean(abs(forecast_errors))
> # Compute Mean Absolute Percentage Error (MAPE)
> MAPE <- mean(abs(forecast_errors / actual_values)) * 100
> # Print forecast performance measures
> cat("RMSE:", RMSE, "\n")
RMSE: 0.9128336
> cat("MAD:", MAD, "\n")
MAD: 0.694335
> cat("MAPE:", MAPE, "%\n")
MAPE: 110.9544 %
```

**Time Series Analysis: Homework 4** (20232863 Keywoong Bae)