INDR 450/550

Spring 2022

Lecture 7: ARIMA processes (2)

March 7, 2022

Fikri Karaesmen

Announcements

- Class Exercise at the end of lecture today. If you are participating online, please upload your document under Course Contents/Class Exercises
- HW 1 now available.
- The first two labs were uploaded. Please follow them.

ARMA Framework

- We can combine AR-terms and MA-terms. The resulting models are called ARMA and include both AR and MA components.

$$\begin{aligned} Y_t &= c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + \\ &\quad + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + ... + \theta_q \epsilon_{t-q} + \epsilon_t \end{aligned}$$

This is useful in practice because we need flexible models to fit data. Real auto-correlations rarely correspond to pure AR or MA processes.

ARMA: Mixing AR and MA

- ARMA: mixing AR and MA terms
- Ex: $$Y_t = c + \phi_1 Y_{t-1} + \theta_1 \varepsilon_{t-4} + \varepsilon_t$$

- A flexible model: Geometrical decrease starting from from lag 1 but there is an additional single spike at lag 4.

Model Identification

- We have a fairly rich framework but we need to have a way of guiding the model fitting process.
- As we will see, a good part of model fitting may be automated but we obtain much better results if we can do some preliminary analysis to guide the fitting process.
- How do we identify the correct model (how many AR and MA terms at which lags?)
- This cannot be done by simply plotting the data.
- The ACF plot is useful but we need more help.

Model Identification: ACF and PACF

- Auto-Correlation Function (ACF) and Partial Auto-Correlation Function (PACF)
- The ACF gives the $Corr(Y_{t-k}, Y_t)$ as a function of the lag k.
- The PACF is the coefficient that corresponds to the coefficient lag-k when we run a linear regression with lagged observations on the right hand side.
- The ACF and PACF capture different aspects. For instance an AR(1) process has the highest AC at lag 1 but also geometrically decreasing AC's at lags 2, 3 etc. The PA coefficient just takes a non-zero value at lag 1 but there is no PA at other lags.

$$Y_t = c + \phi_1 Y_{t-1} + \epsilon_t$$

Summary of ACF and PACF patterns for simple AR and MA models

Process	ACF	PACF
AR(1)	Exponential decay: on positive side if $\phi_1 > 0$ and alternating in sign starting on negative side if $\phi_1 < 0$.	Spike at lag 1, then cuts off to zero: spike positive if $\phi_1 > 0$, negative if $\phi_1 < 0$.
AR(p)	Exponential decay or damped sine-wave. The exact pattern depends on the signs and sizes of $\phi_1, \ldots, \phi_p$.	Spikes at lags 1 to p, then cuts off to zero.
MA(1)	Spike at lag 1 then cuts off to zero: spike positive if $\theta_1 < 0$, negative if $\theta_1 > 0$.	Exponential decay: on negative side if $\theta_1 > 0$ and alternating in sign starting on positive side if $\theta_1 < 0$.
MA(q)	Spikes at lags 1 to q, then cuts off to zero.	Exponential decay or damped sine-wave. The exact pattern depends on the signs and sizes of $\theta_1, \ldots, \theta_q$.

Table 7-2: *Expected patterns in the ACF and PACF for simple AR and MA models.*

Makridakis, Wheelwright and Hyndman (1997)

ARIMA Framework

- Finally, we incorporate the basic transformations that are needed to convert the original series to a stationary series. One basic operation is differencing (multiple times if necessary).
- ARMA processes that require differencing are called ARIMA (Auto Regressive Integrated Moving Averages). Integration in this context is viewed as undoing the differencing (i.e. summation).
- We use the convention ARIMA(p, d, q) to denote that the original series was differenced d times, and then p AR and q MA terms were used on the differenced series.
- The ARIMA class is a broad and useful class.

ARIMA Framework: example

- ARIMA(1,1,0) refers to a process which was differenced once, and has an AR term on the difference:

$$\begin{aligned} i)\ W_t &= Y_t - Y_{t-1} \\ ii)\ W_t &= c + \phi_1 W_{t-1} + \epsilon_t \end{aligned}$$

We can revert the transformations to recover the original process:

$$Y_t = Y_{t-1} + W_t = Y_{t-1} + c + \phi_1 W_{t-1} + \epsilon_t$$

Finally replacing W_{t-1} by $Y_{t-1} - Y_{t-2}$, we have:

$$Y_t = Y_{t-1} + W_t = c + Y_{t-1} + \phi_1(Y_{t-1} - Y_{t-2}) + \epsilon_t$$

ARIMA Framework: example

- ARIMA(1,2,1) refers to a process which was differenced twice, and has an AR term and MA term on the second difference:

$$\begin{aligned} i)\ W_t &= Y_t - Y_{t-1} \\ ii)\ Z_t &= W_t - W_{t-1} \\ iii)\ Z_t &= c + \phi_1 Z_{t-1} + \theta_1 \epsilon_{t-1} + \epsilon_t \end{aligned} \tag{8}$$

SARIMA Framework: taking into account seasonality

- In addition to differencing (to remove trend), another common transformation is seasonal differencing.
- This leads to the bigger framework of Seasonal ARIMA (SARIMA).
- The convention is SARIMA$(p, d, q)(P, D, Q, m)$. The second parenthesis refers to the seasonal terms: P is the number of seasonal AR terms, D refers to the degree of seasonal differencing, Q to the number of seasonal MA terms and m the length of the season.

SARIMA Framework: Example

- SARIMA(1,0,1)(1,1,0,12) refers to a process which has an one regular AR term and was seasonally differenced once and has an AR term on the seasonal difference. The length of the season is 12.

This gets quite messy to write in terms of the original series and is very difficult to do without the backshift notation. We'll take a look at next time.

Forecasting in the ARIMA Framework

- Once we pick a model such as ARIMA(1,0,1), the data is estimated from the parameters by Maximum Likelihood Estimation (i.e. find the parameters ϕ_1 and ϵ_1 that would make the observed series most probable).
- This is the hard part of the task but is done by numerical optimization and software has become reliable. item Let us assume that the MLE estimators of the parameters are $\phi_1 = 0.2$ and $\theta_1 = -0.5$.
- The model is then:

$$Y_t = 0.2Y_{t-1} - 0.5\epsilon_{t-1} + \epsilon_t$$

- To 'forecast' from the above process we simply plug in the observed values in the above evolution equation. Assume that $y_{t-1} = 20$

$$\hat{y}_t = 0.2(20) - 0.5(20 - \hat{y}_{t-1})$$

Forecasting in the ARIMA Framework

- Note that $y_1, y_2, ...y_{t-1}$ are observable to us but $\epsilon_1, \epsilon_2, ...\epsilon_{t-1}$ are not observable, we therefore estimate

$$\hat{\epsilon}_{t-1} = y_{t-1} - \hat{y}_{t-1}$$

- If we are forecasting for time $t+1$ using data up to time $t-1$. We proceed with:

where we replaced the observation y_t with its estimator from the model $\hat{y}_t$. Since we have not observed y_t, our best estimator for $\hat{\epsilon}_t = 0$. The multi-step look ahead forecast simply reduces to:

$$\hat{y}_{t+h|t-1} = 0.2\hat{y}_{t+h-1} \text{ for } h = 1, 2, 3...$$

Forecasting in the ARIMA Framework

- Things get messier if we take more complicated models,but the principles are the same. Let us take ARIMA(1,2,1).
- Let us assume that this time the MLE estimators of the parameters are $c = 100$, $\phi_1 = 0.4$ and $\theta_1 = 0.6$.
- The model is then:

$$Z_t = 100 + 0.4Z_{t-1} + 0.6\epsilon_{t-1} + \epsilon_t$$

  where $Z_t = (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2})$.
- For this model, we need the last three observations $y_{t-1}, y_{t-2}, y_{t-3}$ to forecast for period t.

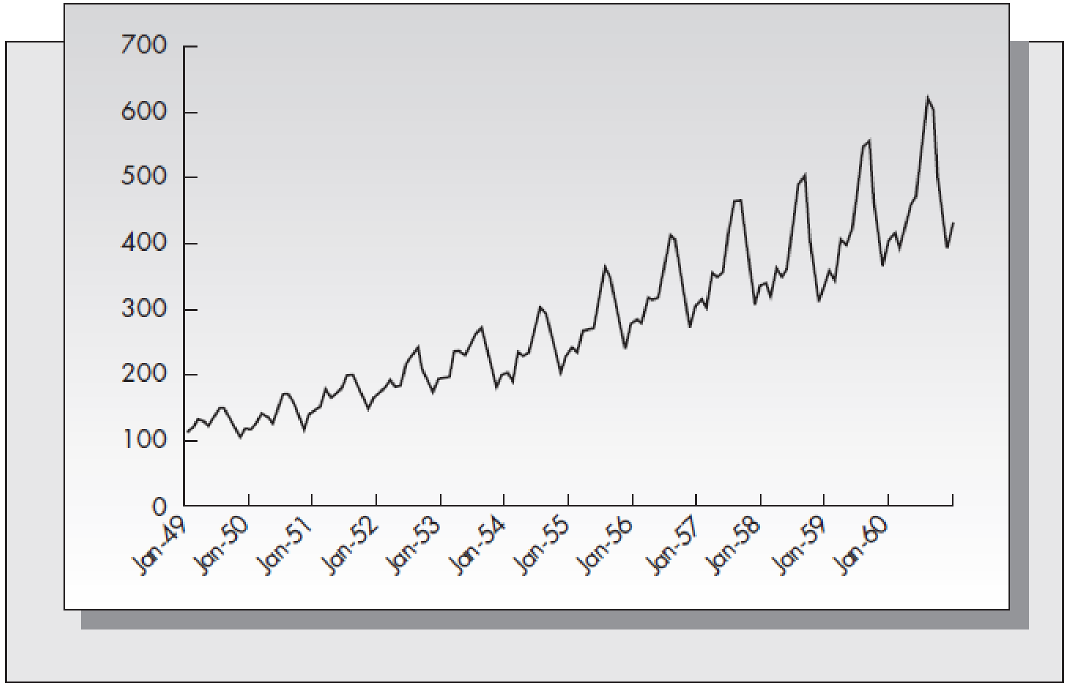
Forecasting in the ARIMA Framework

- With some guidance from our part, software enables us to fit models and perform comparisons across models.
- We can also perform exhaustive searches of a large class of models.
- It is then critical to validate and interpret the results.

Case: Forecasting Airline Passenger Demand

- Airline passenger demand from 1949 to 1960.

FIGURE 2–17
International airline passengers (thousands)

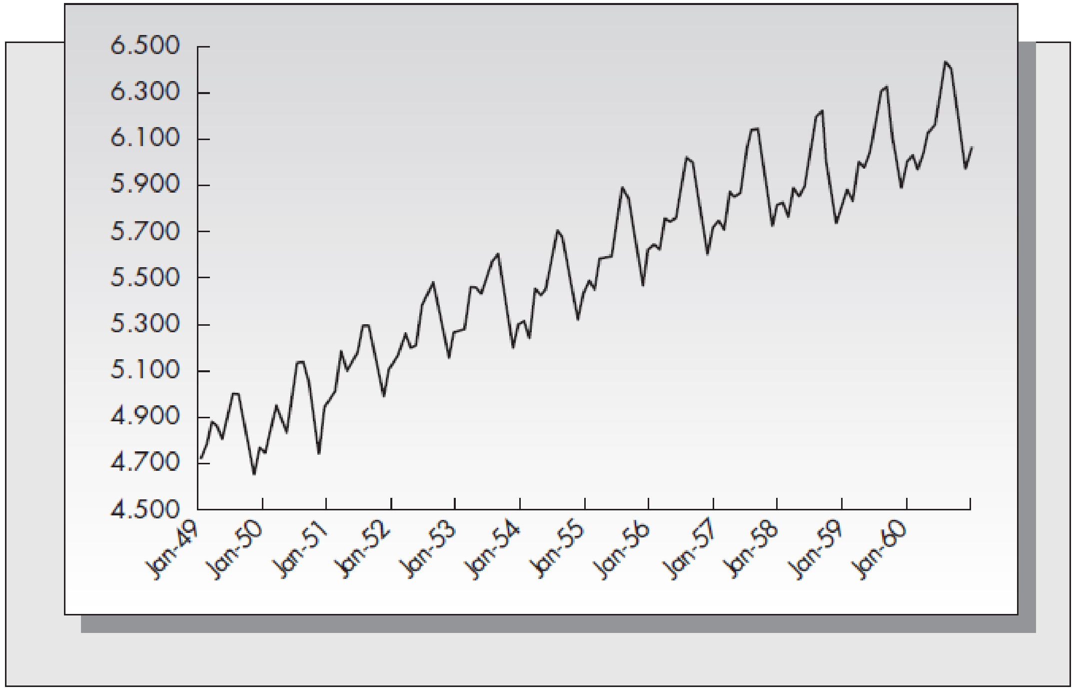


- There is trend and seasonality and increasing variance (fluctuations are increasing).

Step 1: Logarithmic transformation

- To stabilize variance take $Z_t = \ln(Y_t)$

FIGURE 2–18
Natural log of international airline passengers



- The fluctuations now appear stable.
- Trend and seasonality still remain.

Step 2: Detrend and deseasonalize by differencing

- First order differencing to remove linear trend.
- 12 month differencing to remove annual seasonality.
- Check autocorrelations after differencing:

TABLE 2–4
Autocorrelations for the Transformed Airline Data Pictured in Figure 2–19 (after taking logarithms and two levels of differencing)

Lag	Autocorrelation	Lag	Autocorrelation	Lag	Autocorrelation
1	−0.34	13	0.15	25	−0.10
2	0.11	14	−0.06	26	0.05
3	−0.20	15	0.15	27	−0.03
4	0.02	16	−0.14	28	0.05
5	0.06	17	0.07	29	−0.02
6	0.03	18	0.02	30	−0.05
7	−0.06	19	−0.01	31	−0.05
8	0.00	20	−0.12	32	0.20
9	0.18	21	0.04	33	−0.12
10	−0.08	22	−0.09	34	0.08
11	0.06	23	0.22	35	−0.15
12	−0.39	24	−0.02	36	−0.01

- Still significant auto-correlation at 1 lag and 12 lag (additional MA terms are needed) but no other significant AC left.

Step 3: Final model and parameter estimation

- Final model (parameters optimized in Statistical software):

$$z_t = z_{t-1} + z_{t-12} - z_{t-13} + \epsilon_t - 0.333\epsilon_{t-1} - 0.544\epsilon_{t-12} + 0.181\epsilon_{t-13}.$$

- And don't forget:

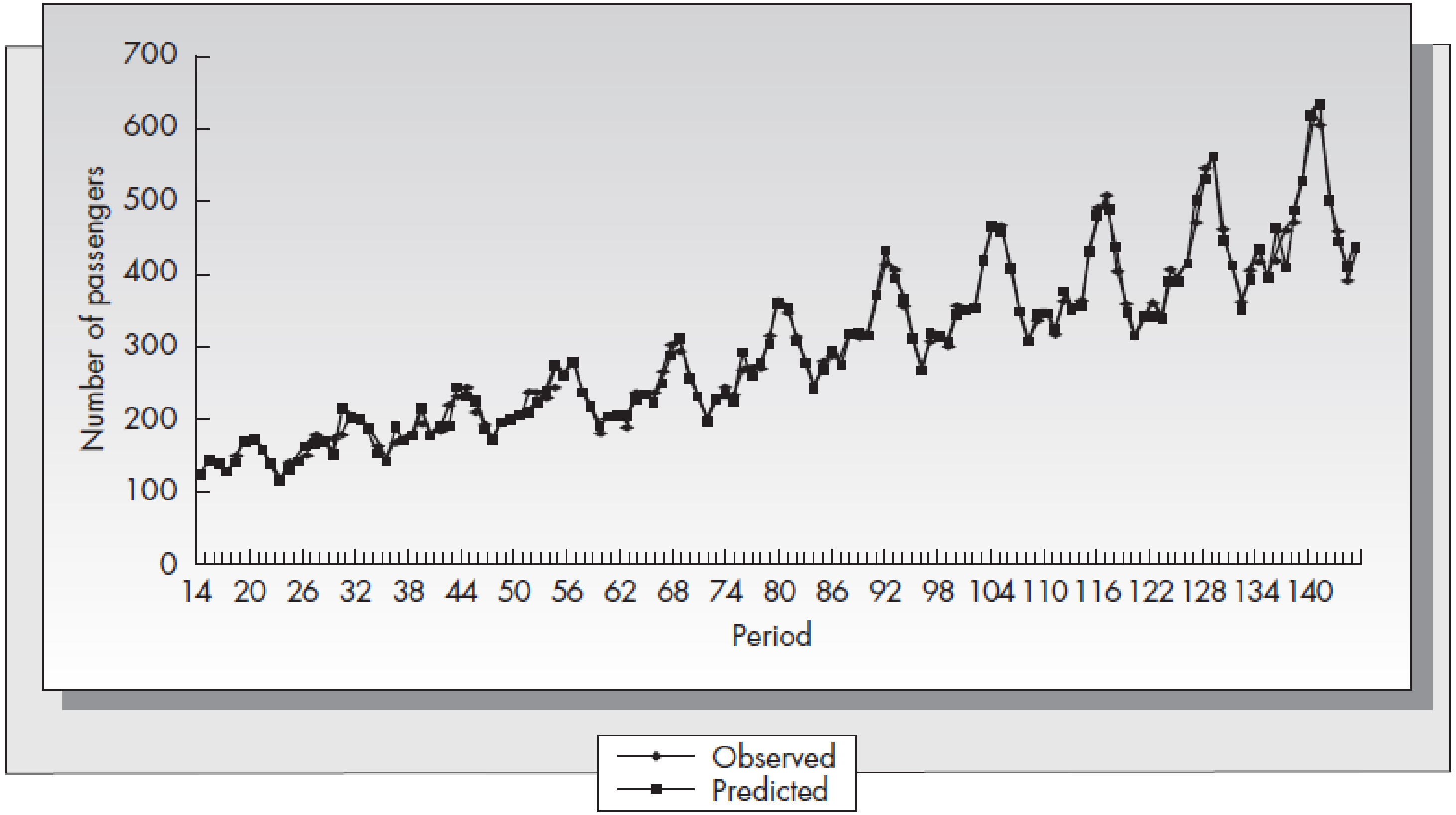
$$Z_t = \ln(Y_t), \quad Y_t = e^{Z_t}$$

- Because of the first order differencing and the yearly differencing, the first forecast can be made for period 14.

Results

- From period 14 on:

FIGURE 2–20
Observed versus predicted number of airline sales



- The resulting forecasts are excellent!

Model Fitting Examples: Simulated AR(1)

- Example: Let us try the model fitting implementation on a synthetic case. Let's generate data from an AR-1 process.

$$Y_t = c + \phi_1 Y_{t-1} + \epsilon_t$$

- We would need to plot the ACF and PACF to have some guidance, but since we have synthesized data, we already know that the best fit is ARIMA(1,0,0).
- We can then let the software estimate the intercept and the first lag AR coefficients: $\hat{c}$ and $\hat{\phi}_1$ and assess the results.

The model that is simulated is $Y_t = 250 + 0.7\, Y_{t-1} + \varepsilon_t$ and $\sigma^2=100$.

Model Fitting Examples

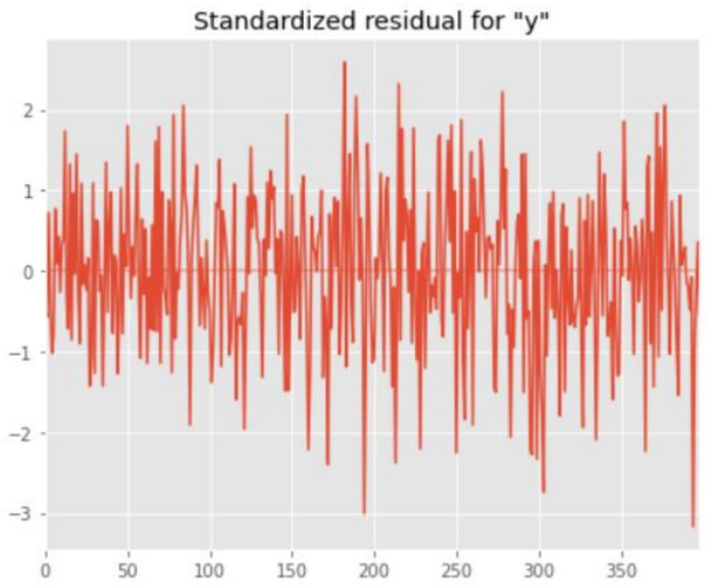
```
7]: # Fit the model
modar = sm.tsa.statespace.SARIMAX(y_ar[100:499], trend='c', order=(1,0,0))
res = modar.fit(disp=False)
print(res.summary())
```

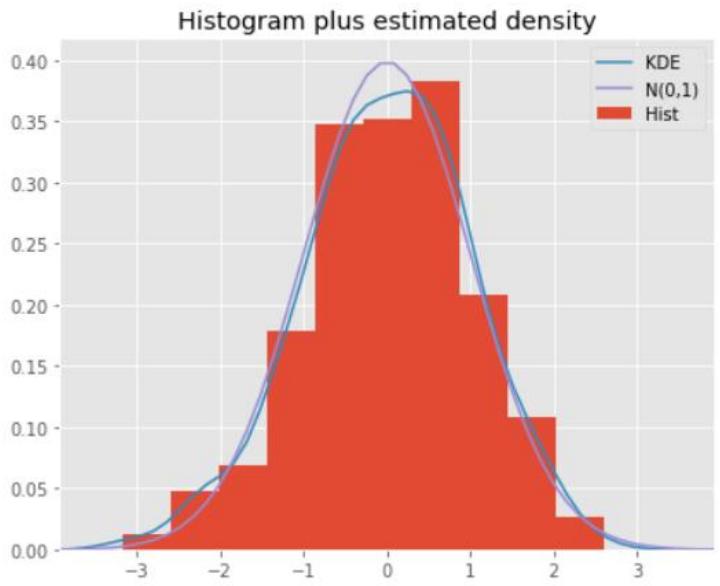
```
                               SARIMAX Results
==============================================================================
Dep. Variable:                      y   No. Observations:                  399
Model:               SARIMAX(1, 0, 0)   Log Likelihood               -1458.647
Date:                Tue, 01 Mar 2022   AIC                           2923.295
Time:                        10:09:29   BIC                           2935.261
Sample:                             0   HQIC                          2928.034
                                - 399
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept    293.9478     33.949      8.658      0.000     227.408     360.487
ar.L1          0.6470      0.041     15.866      0.000       0.567       0.727
sigma2        87.7254      6.350     13.815      0.000      75.279     100.172
===================================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):                 3.89
Prob(Q):                              0.95   Prob(JB):                         0.14
Heteroskedasticity (H):               1.44   Skew:                            -0.24
Prob(H) (two-sided):                  0.04   Kurtosis:                         3.01
===================================================================================
```

The fitted model is $Y_t = 293 + 0.64\ Y_{t-1} + \varepsilon_t$

The true model was $Y_t = 250 + 0.7\ Y_{t-1} + \varepsilon_t$ and $\sigma^2=100$.

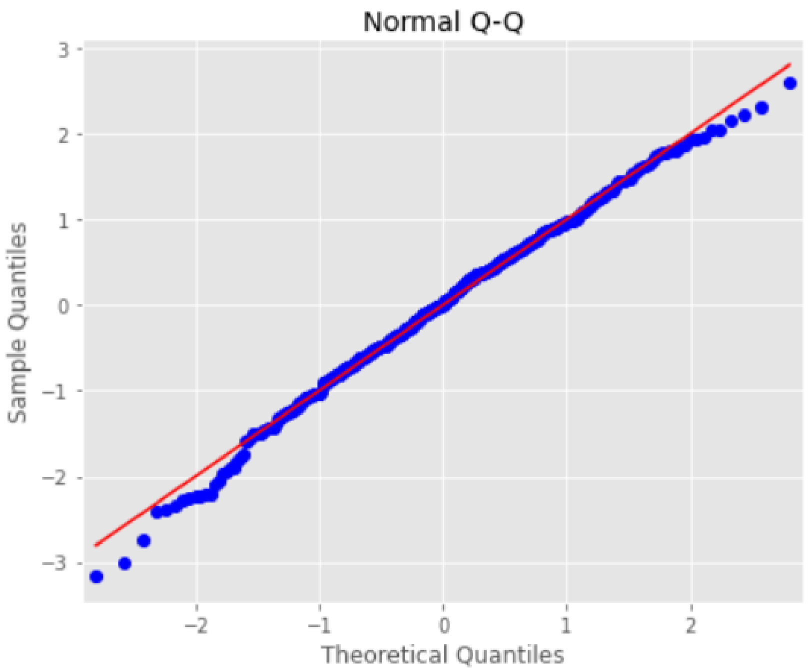
Model Fitting Examples: Residual Checks

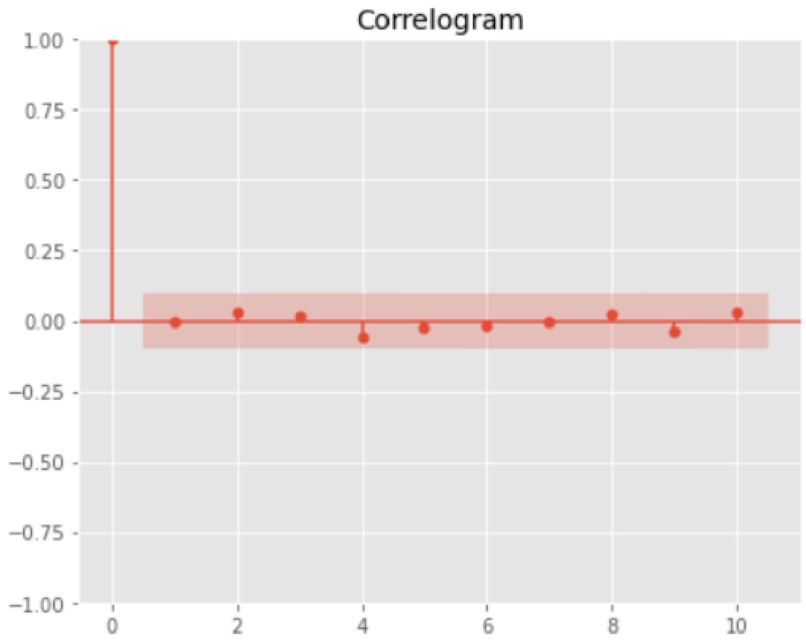




The residuals seem random and the histogram looks like a normal distribution.

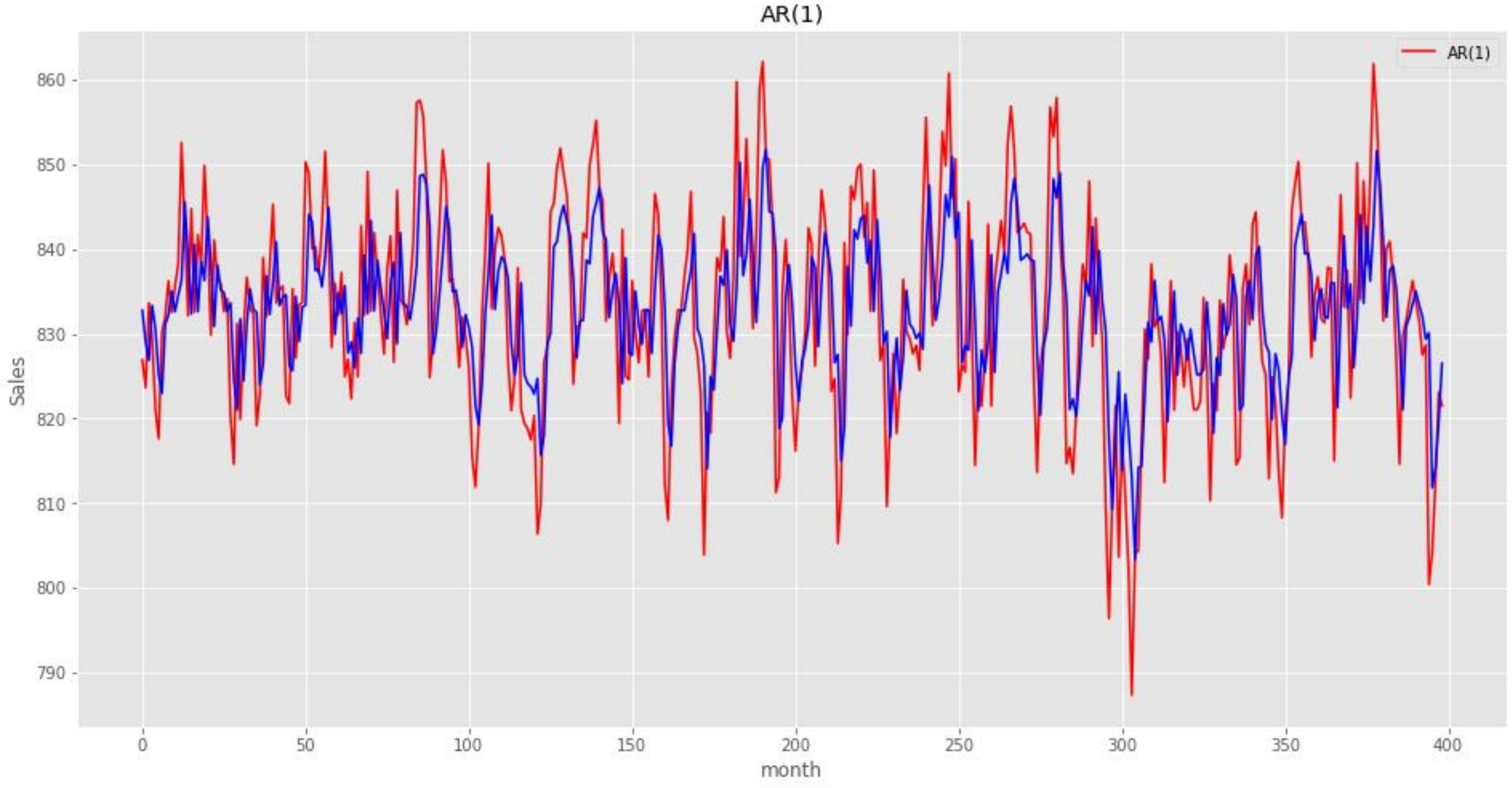
Model Fitting Examples: Residual Checks





The Q-Q plot confirms normality and there is no significant auto-correlation in the residuals. We can conclude that there errors are independent and normally distributed with mean zero and variance 87.72 (from the results table).

Model Fitting Examples: In-sample predictions



Model Fitting Examples: (wrong) ARMA

Let's check the effect of fitting a wrong model. For instance, we might wrongfully think that MA terms are needed at lags 1 and 3.

- We can also attempt to fit a wrong (or superficial) model.For instance, we can attempt to fit:

$$Y_t = c + \phi_1 Y_{t-1} + \theta_1 \epsilon_{t-1} + \theta_3 \epsilon_{t-3} + \epsilon_t$$

- Note that the above is not exactly ARIMA(1,0,3) since it does not contain the MA-term at the second lag.
- We would need a more complete specification and use ARIMA(1,0,[1,0,1]).

Model Fitting Examples: (wrong) ARMA

In [4]:
```
# Fit the model
restest = modtest.fit(disp=False)
print(restest.summary());
```

```
                               SARIMAX Results
==================================================================================
Dep. Variable:                          y   No. Observations:                  399
Model:             SARIMAX(1, 0, [1, 3])   Log Likelihood               -1483.013
Date:                    Sun, 06 Mar 2022   AIC                           2976.026
Time:                            18:47:31   BIC                           2995.971
Sample:                                 0   HQIC                          2983.926
                                    - 399
Covariance Type:                      opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept    275.9916     49.422      5.584      0.000     179.127     372.857
ar.L1          0.6693      0.059     11.292      0.000       0.553       0.786
ma.L1          0.0112      0.077      0.146      0.884      -0.140       0.162
ma.L3         -0.0270      0.061     -0.442      0.658      -0.147       0.093
sigma2        98.9059      7.420     13.329      0.000      84.362     113.450
===================================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):                 0.36
Prob(Q):                              0.96   Prob(JB):                         0.84
Heteroskedasticity (H):               0.92   Skew:                             0.05
Prob(H) (two-sided):                  0.64   Kurtosis:                         2.89
===================================================================================
```

Model Fitting Examples: (wrong) ARMA

- The resulting model is very different than the theoretical model we simulated.
- MSE 1 = 99.62, MSE 2 = 99.55
- Second model has lower MSE but it looks very suspicious because the p-values of the the two MA terms are not statistically significant.
- These are all signs of overfitting due to the additional parameters.
- We’ll do our best to avoid overfitting.