



# 시계열 분석

## 시계열 분석

일정한 시간을 간격으로 표시된 자료의 특성을 파악하여 미래를 예측하는 방법

독립변수(x)와 종속변수(y)가 시간단위(xt)를 가진다.

회귀분석 vs 시계열 분석 - 시간을 고려하는가

## 시계열 데이터의 특성

시계열 정보 = 규칙적인 패턴 + 불규칙적인 패턴

### 규칙적인 패턴

자기상관성 - 이전의 결과와 이후의 결과 사이에서 발생

바로 이전의 결과의 영향을 받을 수도 있지만 Delay가 발생하기도 한다.

이동평균 - 이전에 생긴 불규칙한 사건이 이후의 결과에 편향성을 초래

### 불규칙적인 패턴(White Noise)

평균이 0이며 일정한 분산을 지닌 정규분포에서 추출된 임의의 수

대표적인 모델 : AR, MA, ARMA, ARIMA, ARIMAX

## 시계열 모형

### 1.자기상관(AR) - 자기상관성을 시계열 모형으로 구성한 것

바로 직전의 값 1개가 다음 값에 영향을 미치는 모형

$$X(t) = a * X(t-1) + c + u * e(t)$$

### 2이동평균(MA) - 시간이 지날수록 어떠한 변수의 평균값이 지속적으로 증/감이 생길 수 있음

직전의 값 1개에서 발생한 오차가 다음 데이터에 영향을 준다고 가정한 모형

$$X(t) = a * e(t-1) + c + u * e(t)$$

### 3.ARMA 모형 - AR+MA

$$ARMA(1,1)$$

$$X(t) = a * X(t-1) + b * e(t-1) + c + u * e(t)$$

### 4.ARIMA 모형 - 과거의 데이터가 지닌 추세(trend)까지 반영한 모형

추세는 자기 자신의 추세만 반영하며 백색소음(White Noise)을 고려하지 않는다.

- 추세관계

두 변수 X-Y간에 cointegration이 0보다 크면 X의 값이 이전 값보다 증가하면 Y값도 증가

두 변수 X-Y간에 cointegration이 0보다 작으면 X의 값이 이전 값보다 증가하면 Y값은 감소

$$ARIMA(1, 1, 1)$$

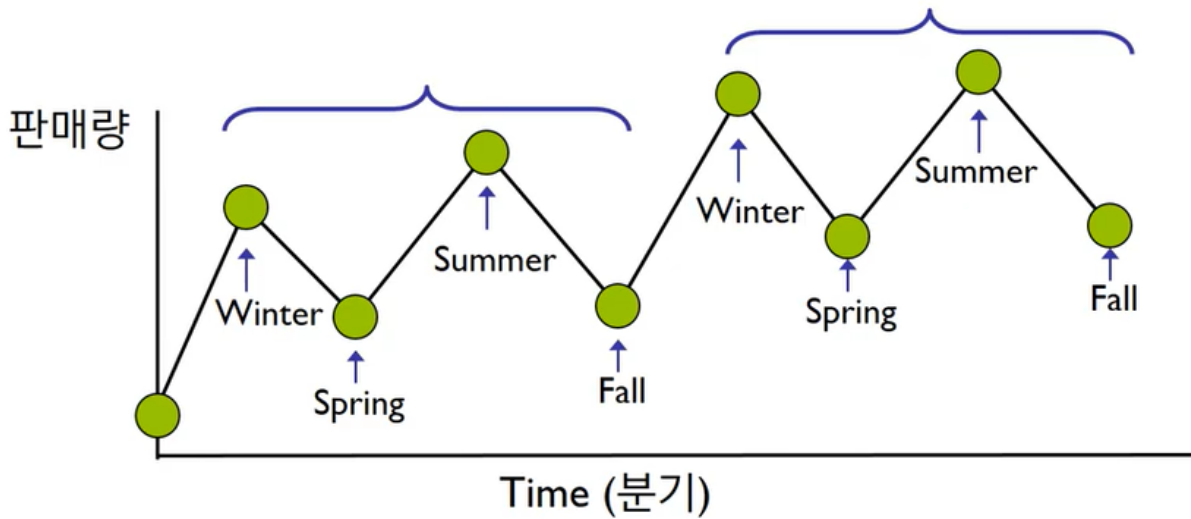
$$a * X(t) - X(t - 1) = b * X(t - 1) + c * e(t - 1) + d + u * e(t)$$

$$X(t) = [X(t - 1) + b * X(t - 1) + c * e(t - 1) + d + u * e(t)] / a$$

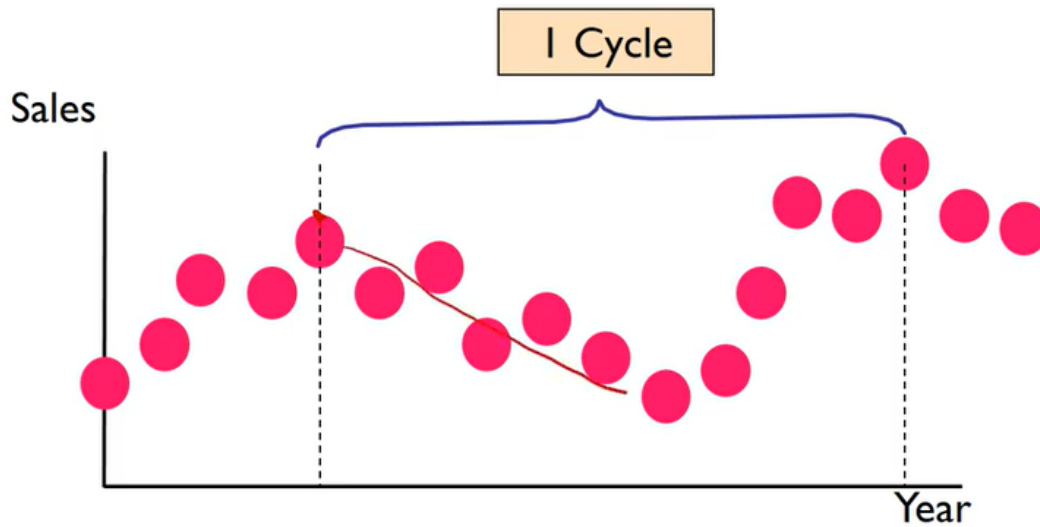
1.추세(Trend) - 세부적인 데이터가 아닌 전체적인 동향을 이용하는 것



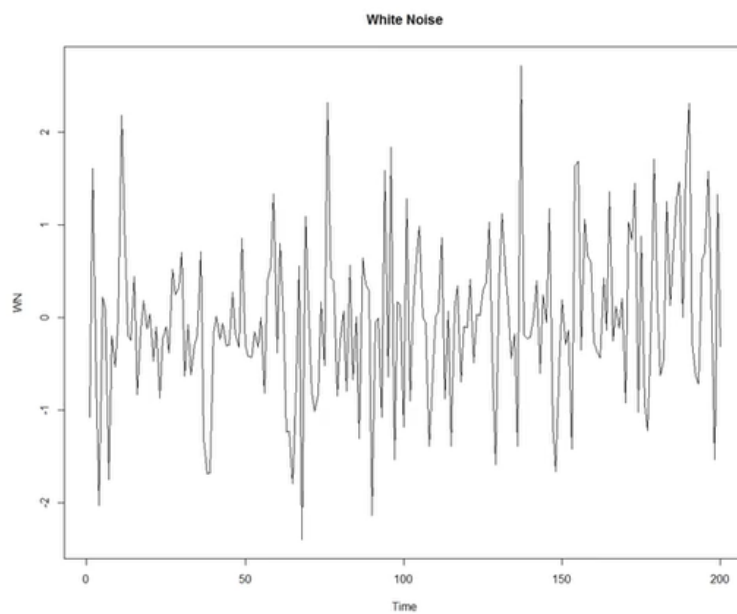
2.계절성(Seasonality) - 특정한 기간을 토대로 기간마다 어떤 패턴을 가지는 지 확인하는 것



3.순환(Cyclical) - 2~3년 정도의 기간을 주기로 순환 발생

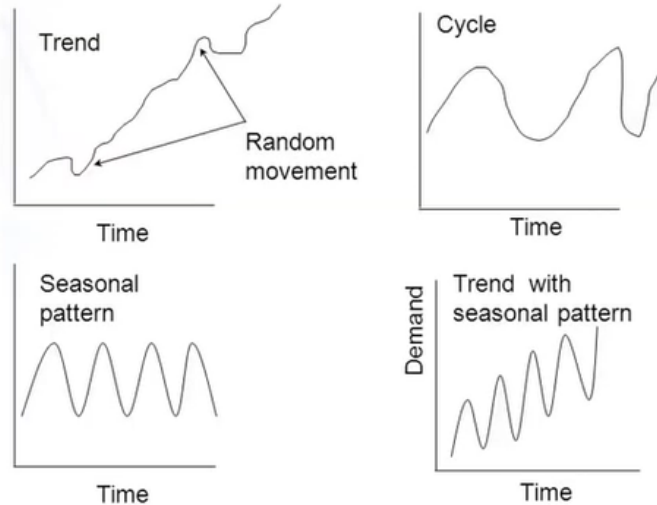


#### 4.우연변동(Random) - 추세, 계절성으로 설명할 수 없는 시계열 데이터

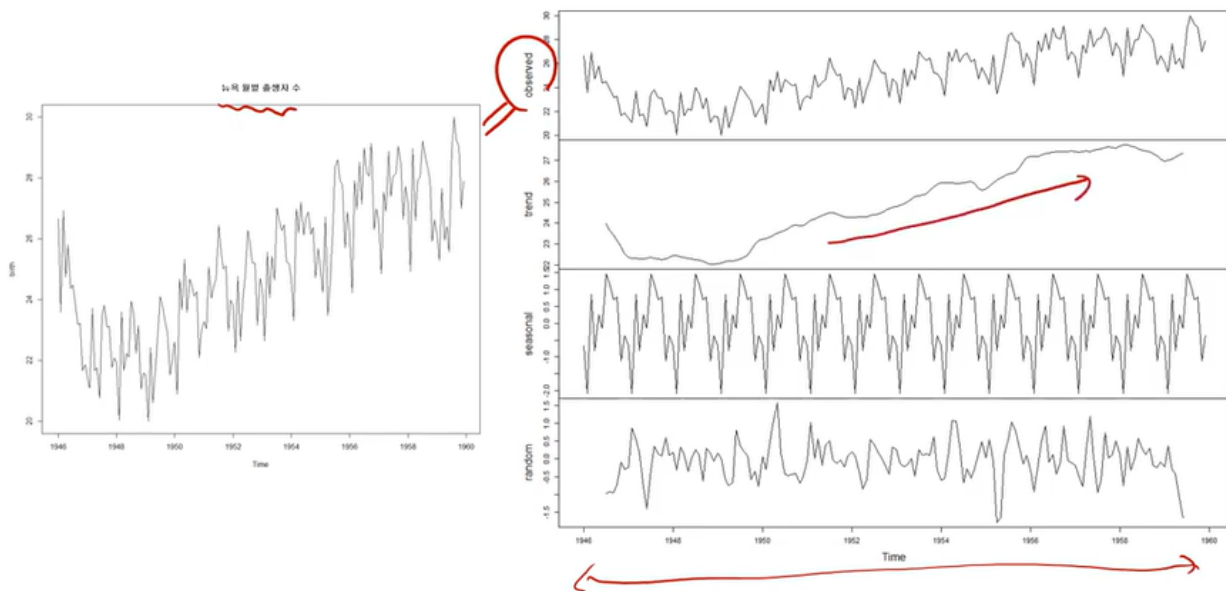


- 시간에 따른 규칙적인 움직임과는 무관하게 랜덤한 원인에 의해 나타나는 변동
- 백색잡음 (White Noise): 평균이 0이고 분산이 일정한 시계열 데이터

- 추세변동 (Trend)
- 순환변동 (Cycle)
- 계절변동 (Seasonal variations)
- 우연변동 (Random fluctuation)



\*observed = trend + seasonal + random

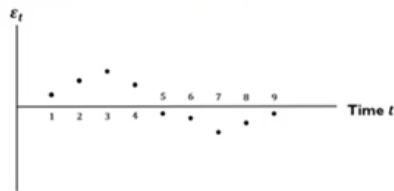


### 자기상관성 - Autocorrelation

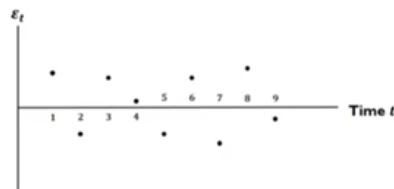
자기 자신(데이터)가 한 단계 아래로 밀린 데이터 간의 상관성 ( $x_1$ 과  $x_1'$ 의 상관계수)

$x_t$	$x'_t$
2	0
3	2
5	3
6	5
7	6
9	7
0	9

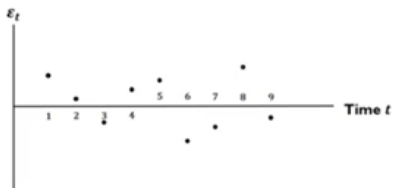
Positive Autocorrelation - 양수 다음엔 양수, 음수 다음엔 음수가 나오는 자기 상관성



Negative Autocorrelation - 양수 다음엔 음수, 음수 다음엔 양수가 나오는 자기 상관성



Random Autocorrelation - 규칙적인 패턴이 없는 자기 상관성



\*자기상관성이 없다면 기본 회귀분석 기법과 차이가 없기 때문에 꼭 확인이 필요함

회귀분석은  $cov(x_1, x_2) = 0$ 으로 가정하지만 시계열 데이터는 이를 위반할 가능성이 높음

### Darbin-Watson Test - 자기상관성 여부 확인 방법

귀무가설과 대립가설을 통해 확인하는 방법

1. Positive Autocorrelation 확인 방법

검정통계량(d) < dL - 대립가설 채택

검정통계량(d) > dU - 귀무가설 채택

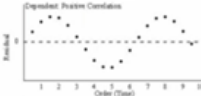
dL < d < du - 부적절한 검증 방법

• For the first-order positive autocorrelation.

$\chi, \chi'$   $\left\{ \begin{array}{l} H_0 : \rho = 0 \quad \text{The error terms are not autocorrelated.} \\ H_1 : \rho > 0 \quad \text{The error terms are positively autocorrelated.} \end{array} \right.$

$$d = \frac{\sum_{i=2}^n (e_i - e_{i-1})^2}{\sum_{i=1}^n e_i^2} \quad \text{where } e_i = y_i - \hat{y}_i$$

If  $d < d_L$  reject  $H_0 : \rho = 0$   
 If  $d > d_U$  do not reject  $H_0 : \rho = 0$   
 If  $d_L < d < d_U$  test is inconclusive.



예제

$$H_0 : \rho = 0$$

$$H_1 : \rho > 0$$

$$d = \frac{\sum_{i=2}^{20} (e_i - e_{i-1})^2}{\sum_{i=1}^{20} e_i^2} = \frac{8195.2065}{7587.9154} = 1.08$$

$$\alpha = 0.05 \quad d_L = 1.20 \quad \text{and} \quad d_U = 1.41$$

$$d = 1.08 < d_L = 1.20$$

Reject  $H_0$   
Errors are positively autocorrelated.

## 2. Negative Autocorrelation 확인 방법

if  $(4-d) < d_L$  - 대립가설 채택

if  $(4-d) > d_U$  - 귀무가설 채택

$d_L < (4-d) < d_U$  - 부적절한 검증 방법

Consider testing the null hypothesis

$H_0$ : The error terms are not autocorrelated

Versus the alternative hypothesis

$H_1$ : The error terms are negatively autocorrelated

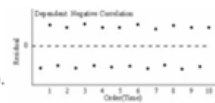
Durbin and Watson have shown that based on setting the probability of a Type I error equal to  $\alpha$ , the points  $d_{L,\alpha}$  and  $d_{U,\alpha}$  are such that

$$d = \frac{\sum_{i=2}^n (e_i - e_{i-1})^2}{\sum_{i=1}^n e_i^2} \quad \text{where } e_i = y_i - \hat{y}_i$$

If  $(4-d) < d_{L,\alpha}$  we reject  $H_0$ .

If  $(4-d) > d_{U,\alpha}$  we do not reject  $H_0$ .

If  $d_{L,\alpha} \leq (4-d) \leq d_{U,\alpha}$  the test is inconclusive.



## 시계열 모형 선택방법 (Box-Jenkins ARIMA Procedure)

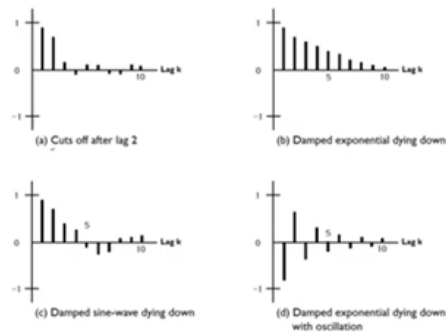
주관적인 방법이기 때문에 다시 한 번 검증이 필요함

## Identification ARIMA Model

- ❖ Graphical method: making inferences from the patterns of the sample autocorrelation and partial autocorrelation functions of the series

Model	ACF	Partial ACF
MA(q)	Cut off after lag q (q시차 이후 0으로 절단)	Die out (지수적으로 감소, 소멸하는 sine함수 형태)
AR(p)	Die out (지수적으로 감소, 소멸하는 sine함수 형태)	Cut off after lag p (p시차 이후 0으로 절단)
ARMA(p, q)	Die out (시차 (q-p)이후 부터 소멸)	Die out (시차 (q-p)이후 부터 소멸)

1번의 경우 - Cut Off, 2,3,4번의 경우는 모두 Die Out을 의미함.



### 시계열 모형

1. 자기상관 - AR - 이전의 값이 이후의 값에 영향을 미치고 있는 경향을 반영

= 자기 자신(x)에 lag된 값들과의 관계를 모델링하는 방법

$$y_t = \phi_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t$$

$\underbrace{\phi_1}_{\chi_1} \quad \underbrace{\phi_2}_{\chi_2} \quad \dots \quad \underbrace{\phi_p}_{\chi_p}$

2. 이동평균 - MA - 시간이 지날수록 어떤 평균값이 지속적으로 증가하거나 감소하는 경향 반영

= 연속적인 Error term으로 y와의 관계를 모델링하는 방법

$$y_t = \theta_0 + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q}$$

### 3. ARMA - AR+MA

$$y_t = \phi_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q}$$

\*1,2,3번 방법은 시계열 데이터가 정상성을 가질 때 적용이 가능한 모델이다.

정상성을 만들어주기 위해 differencing을 사용한다

4. ARIMA - ARMA 모형에 과거의 데이터가 지니고 있던 추세까지 반영

I(differencing)을 추가한 모델링 기법 (AR(p), I(d), MA(q))

p - Independent Variable의 개수

d - Differencing의 횟수

q - 파라미터의 개수

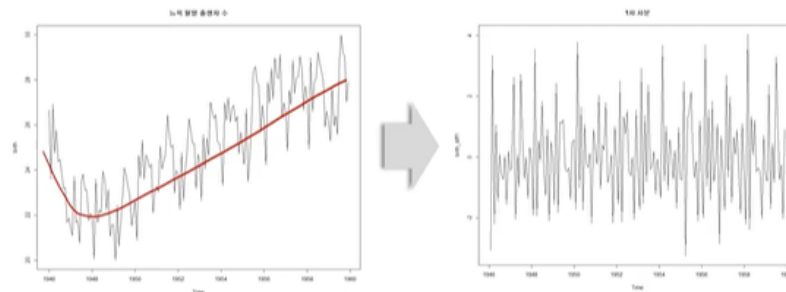
**Differencing - 차분(현 시점 데이터에서 d시점 이전 데이터를 뺀 것)**

효과 : 비정상성 시계열 데이터도 차분을 하면 정상성을 가질 확률이 높음

$$1차 차분: Y_t = X_t - X_{t-1} = \nabla X_t$$

$$2차 차분: Y_t^{(2)} = X_t - X_{t-2} = \nabla^{(2)} X_t$$

$$d차 차분: Y_t^{(d)} = X_t - X_{t-d} = \nabla^{(d)} X_t$$



X		
2	X	Y
7	2	5
10	7	3
5	10	-5
8	5	3
	8	-

\*오리지널 데이터가 정상성을 가지면 차분이 필요없음

\*오리지널 데이터가 Constant하다면 1차 차분으로 충분하다 (복잡할 경우 최대 2차 차분)

\*대부분의 문제는 2차 차분으로 충분하다.



## 알아야 할 용어

### AIC - Akaike's Information Criterion

$$AIC = 2 (\log\text{-likelihood}) + 2k$$

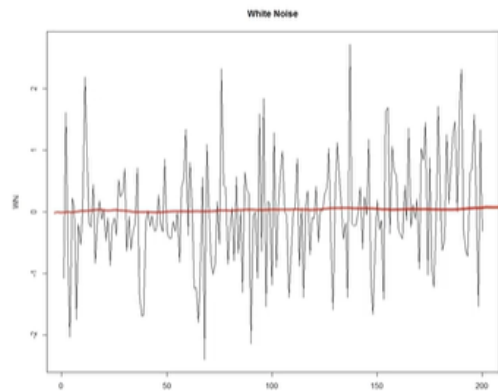
log-likelihood : 모형 적합도를 나타낸 척도

k = 모형 파라미터의 개수(독리변수의 개수)

\*정규분포를 따르는 잔차를 가진 경우 -  $AIC = n \log(\sigma^2) + 2k$

### 정상 프로세스 (Stationary Process)

시간에 관계없이 평균과 분산이 일정한 시계열 데이터



### 정상성 확인

acf - AutoCorrelation Function

pacf - Partial AutoCorrelation Function

lag - ex)lag = 1 → 현재 데이터와 1시점을 미룬 데이터 (차분 파트 참조)

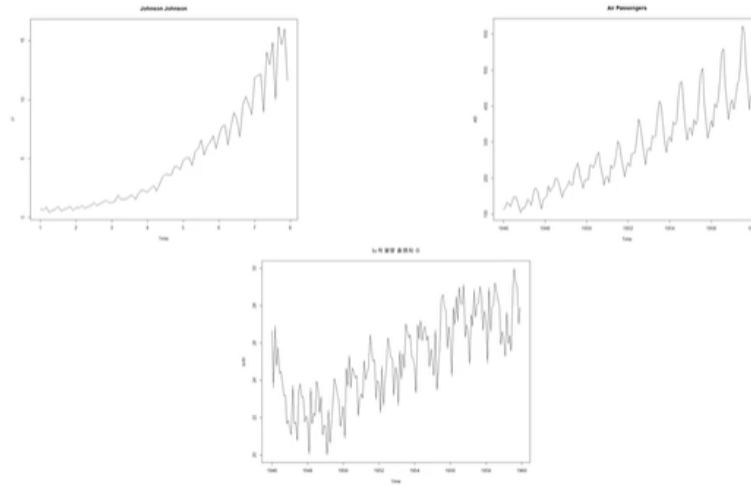
정상성 확인

```
import Statsmodels.api as sm

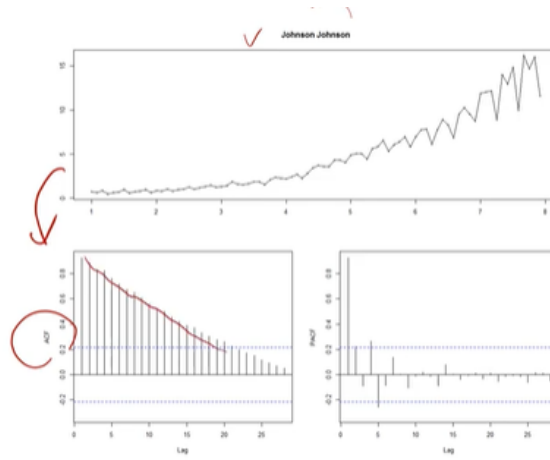
sm.graphics.tsa.plot_acf(train.values.squeeze(), lags=30, ax=ax[0]) #acf =
sm.graphics.tsa.plot_pacf(train.values.squeeze(), lags=30, ax=ax[1])
```

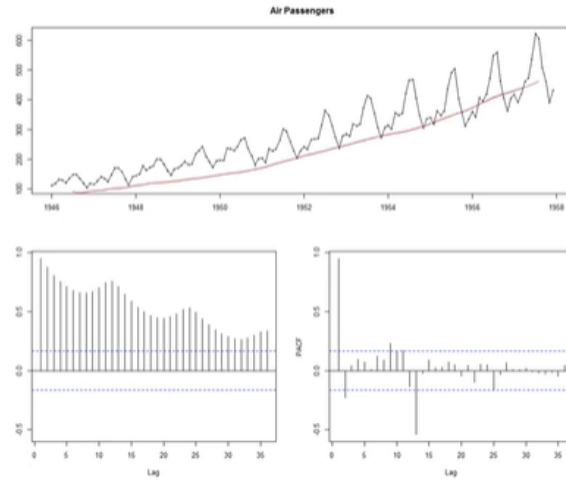
### 비정상 프로세스 (Nonstationary Process)

시간에 따라서 평균 혹은 분산이 일정하지 않은 시계열 데이터



### 비정상성 확인 (Autocorrelation Function의 패턴을 이용)





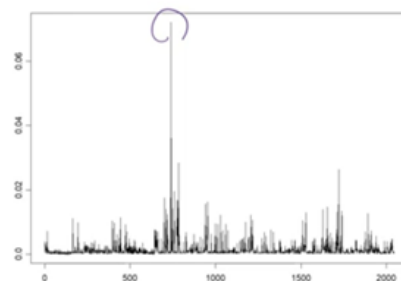
## Box-Jenkins ARIMA Procedure 절차

- 1.데이터 전처리
- 2.임시 사용 모델 선택
- 3.파라미터 추정
- 4.모델 적합성 검증
- 5.적합할 시 모델 사용

### 1.Original 데이터 확인

```
fig = df.plot()
fig

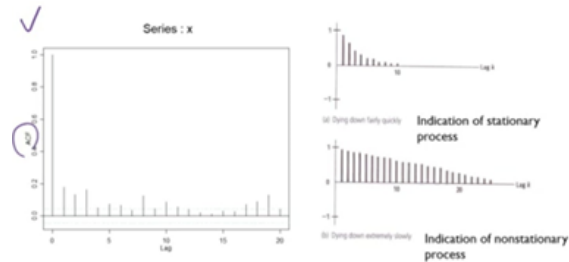
#시계열 분해
decomposition = sm.tsa.seasonal_decompose(df['passengers'], model='additive', period=1) #trend, seasonal, Resid를 확인하기 위한 작업
fig = decomposition.plot()
fig.set_size_inches(10,10)
```



### 2.정상성 or 비정상성 확인 (acf 확인)

```
fig, ax = plt.subplots(1, 2, figsize=(10,5))
fig.suptitle('Raw Data')

sm.graphics.tsa.plot_acf(train.values.squeeze(), lags=30, ax=ax[0])
sm.graphics.tsa.plot_pacf(train.values.squeeze(), lags=30, ax=ax[1])
```

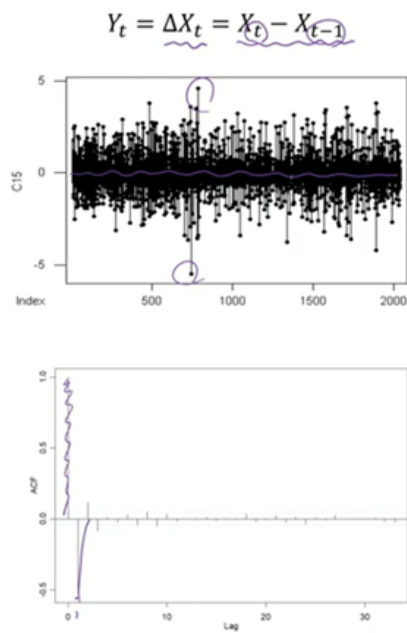


\*lag 0은 무조건 1 (데이터 자기 자신과 비교하는 것이기 때문)

\*lag 1부터 보았을 때 값이 천천히 감소하는 것을 확인 (비정상성의 대표적 현상)

\*정상성을 띄는 경우 급격하게 감소하는 경향을 보임

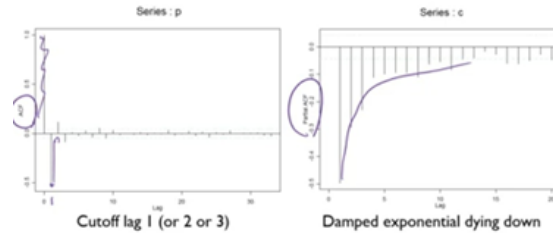
### 3.비정상성임을 확인하고 차분 진행



\*차분을 진행한 결과 정상성을 띄는 시계열 데이터로 변환 것을 확인

### 4.Box-Jenkins ARIMA Procedure을 통해 사용할 분석 모형 선택

Model	ACF	Partial ACF
✓ MA(q)	Cut off after lag q (q시차 이후 0으로 절단)	Die out (지수적으로 감소, 소멸하는 sine함수 형태)
✓ AR(p)	Die out (지수적으로 감소, 소멸하는 sine함수 형태)	Cut off after lag p (p시차 이후 0으로 절단)
✓ ARMA(p, q)	Die out (시차 (qp)이후 부터 소멸)	Die out (시차 (qp)이후 부터 소멸)



**\*Acf - Cut Off, Pacf - Die out 이라고 판단하여 MA 모델 적용**

## 5.MA모델 적용

```
model = ARIMA(train.values, order=(0,1,1))
model_fit = model.fit()
model_fit.summary()

#한 가지만 적용하는 것이 아닌 주변의 값들을 대입해 모델을 더 돌려본다.
#최적 파라미터 구하기
print('Examples of parameter combinations for Seasonal ARIMA')
p = range(0,3)
d = range(1,2)
q = range(0,3)
pdq = list(itertools.product(p, d, q))

aic = []
for i in pdq:
    model = ARIMA(train.values, order=i)
    model_fit = model.fit()
    print(f'ARIMA: {i} >>AIC : {round(model_fit.aic,2)}')
    aic.append(round(model_fit.aic,2))

#최적 파라미터 추출
optimal = [(pdq[i], j) for i, j in enumerate(aic) if j == min(aic)]
optimal

#모델링 (최적 파라미터 이용)
model_opt = ARIMA(train.values, order=(2,1,1))
model_opt_fit = model_opt.fit()
model_opt_fit.summary()
```

```
Method: Maximum Likelihood
Model: 0 1 1
Period: 1

Coefficients:
    MA : 0.85534

Variance-Covariance Matrix:
    ma(1) 0.0001311119
AIC: 5174.86349
```

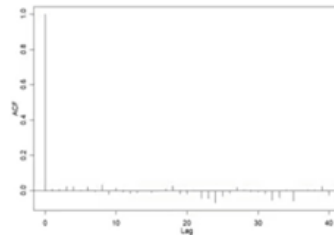
```
Method: Maximum Likelihood
Model: 0 1 2
Period: 1

Coefficients:
    MA : 0.82662 0.04391

Variance-Covariance Matrix:
    ma(1) 0.0004875777 -0.0004215513
    ma(2) -0.0004215513 0.0004875777
AIC: 5172.48219
```

## 6. 최적 파라미터 모델 성능 평가

ACF Plot of the residual values from the ARIMA (0,1,3) model



\*최적의 MA모형 파라미터가 (0,1,3)임을 확인

## SARIMA 모델

- 기존 ARIMA 모델에 계절성을 반영한 모델
- 각 계절에 따른 독립적인 ARIMA 모델이 합쳐진 형태
- ARIMA(p,d,q)(P,D,Q)s의 형태로 구성
  - s=12(월별인 경우), s=4(분기인 경우)

$$\begin{aligned}
 & \text{ARIMA } (1,1,1)(1,1,1)_4 \\
 & (1 - \phi_1 B)(1 - \phi_1 B^4)(1 - B)^1(1 - B^4)^1 y_t = (1 + \theta_1 B)(1 + \theta_1 B^4) a_t \\
 & \text{비계절 AR(1)} \quad \text{계절 AR(1)} \quad \text{비계절 차분 1} \quad \text{계절차분 4} \quad \text{비계절 MA(1)} \quad \text{계절 MA(1)} \\
 \\
 & \text{ARIMA } (p,d,q)(P,D,Q)_s \\
 & \phi_p(B)\Phi_P(B^s)(1 - B)^d(1 - B^s)^D y_t = \theta_q(B)\Theta_Q(B^s)a_t \\
 & \text{비계절 AR(p)} \quad \text{계절 AR(p)} \quad \text{비계절 차분 d} \quad \text{계절차분 D} \quad \text{비계절 MA(q)} \quad \text{계절 MA(q)}
 \end{aligned}$$