

Chapter 17 시계열분석 v1.1

Sangkon Han(sangkon@pusan.ac.kr)

2023-07-02

Contents

비정상성 시계열을 정상성 시계열로 변경	2
단계 1: AirPassengers 데이터 셋 가져오기	2
단계 2: 차분 적용 - 평균 정상화	2
단계 3: 로그 적용 - 분산 정상화	3
실습: 단일 시계열 자료 시각화	4
단계 1: WWWusage 데이터 셋 가져오기	4
단계 2: 시계열 자료 추세선 시각화	5
실습: 다중 시계열 자료 시각화	5
단계 1: 데이터 가져오기	5
단계 2: 데이터프레임으로 변환	6
단계 3: 단일 시계열 자료 추세선 시각화(1,000개 데이터 대상)	6
단계 4: 다중 시계열 자료 추세선 시각화(1,000개 데이터 대상)	6
실습: 시계열 요소분해 시각화	7
단계 1: 시계열 자료 준비	7
단계 2: 시계열 자료 생성 - 시계열 자료 형식으로 객체 생성	7
단계 3: 추세선 확인 - 각 요인(추세, 순환, 계절, 불규칙)을 시각적으로 확인	8
단계 4: 시계열 분해	8
단계 5: 시계열 분해와 변동요인 제거	9
단계 6: 추세요인과 불규칙요인 제거	11
실습: 시계열 요소 분해 시각화	13
단계 1: 시계열 자료 생성	13
단계 2: 자기 상관 함수 시각화	13
단계 3: 부분 자기 상관 함수 시각화	14
실습: 시계열 자료의 추세 패턴 찾기 시각화	15
단계 1: 시계열 자료 생성	15
단계 2: 추세선 시각화	15
단계 3: 자기 상관 함수 시각화	16
단계 4: 차분 시각화	17
실습: 이동평균법을 이용한 평활하기	18
단계 1: 시계열 자료 생성	18
단계 2: 평활 관련 패키지 설치	19
단계 3: 이동평균법으로 평활 및 시각화	19
실습: 계절성이 없는 정상성 시계열분석	19
단계 1: 시계열 자료 특성 분석	19

단계 1-1: 데이터 준비	19
단계 1-2: 시계열 객체 생성(12개월: 2015 2월 ~ 2016년 1월)	20
단계 1-3: 추세선 시각화	20
단계 2: 정상성 시계열 변환	20
단계 3: 모형 식별과 추정	21
단계 4: 모형 생성	22
단계 5: 모형 진단(모형의 타당성 검정)	22
단계 5-1: 자기 상관 함수에 의한 모형 진단	22
단계 5-2: Box-Ljung에 의한 잔차항 모형 진단	23
단계 6: 미래 예측(업무 적용)	23
실습: 계절성을 갖는 정상성 시계열분석	25
단계 1: 시계열 자료 특성 분석	25
단계 1-1: 데이터 준비	25
단계 1-2: 시계열 자료 생성	25
단계 1-3: 시계열 요소 분해 시각화	26
단계 2: 정상성 시계열 변환	26
단계 3: 모형 식별과 추정	27
단계 4: 모형 생성	27
단계 5: 모형 진단(모형 타당성 검정)	28
단계 5-1: 자기 상관 함수에 의한 모형 진단	28
단계 5-2: Box-Ljung에 의한 잔차항 모형 진단	28
단계 6: 미래 예측(업무 적용)	29

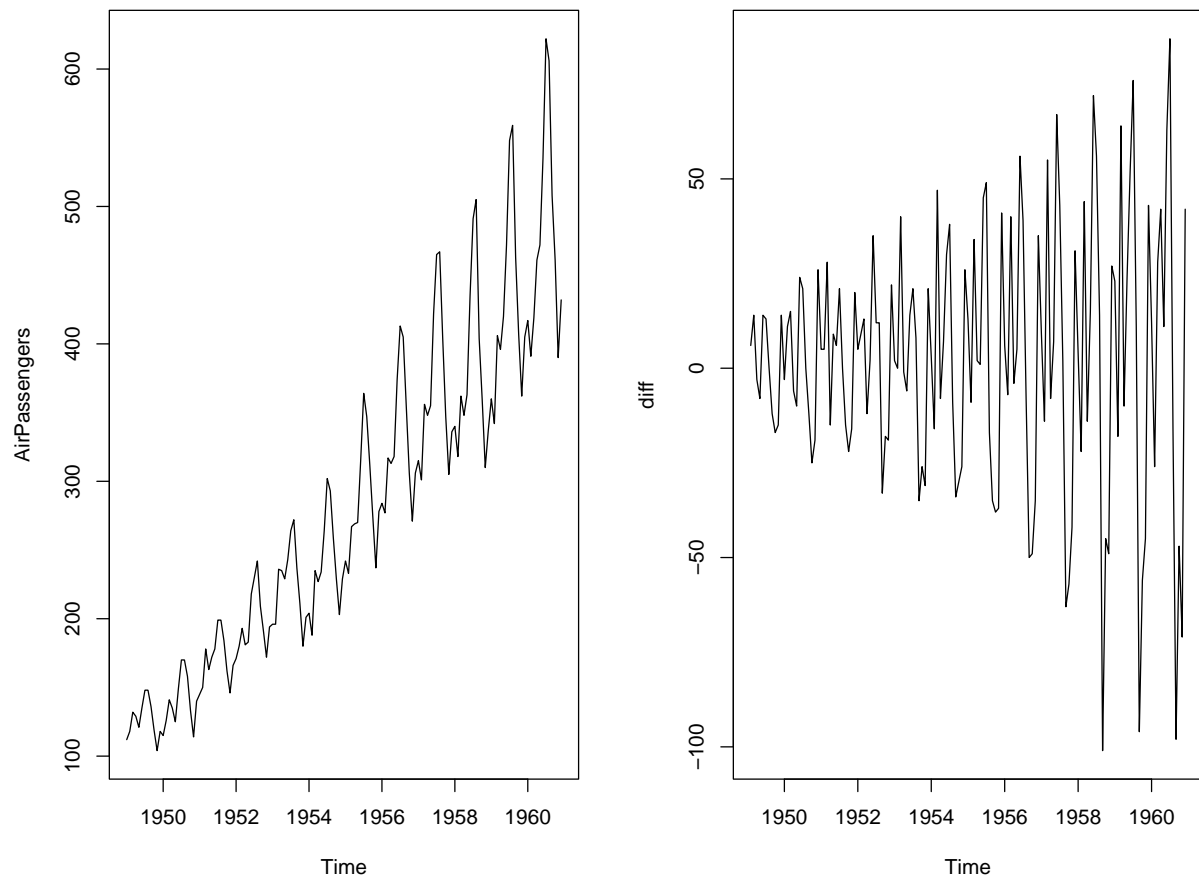
비정상성 시계열을 정상성 시계열로 변경

단계 1: AirPassengers 데이터 셋 가져오기

```
data(AirPassengers)
```

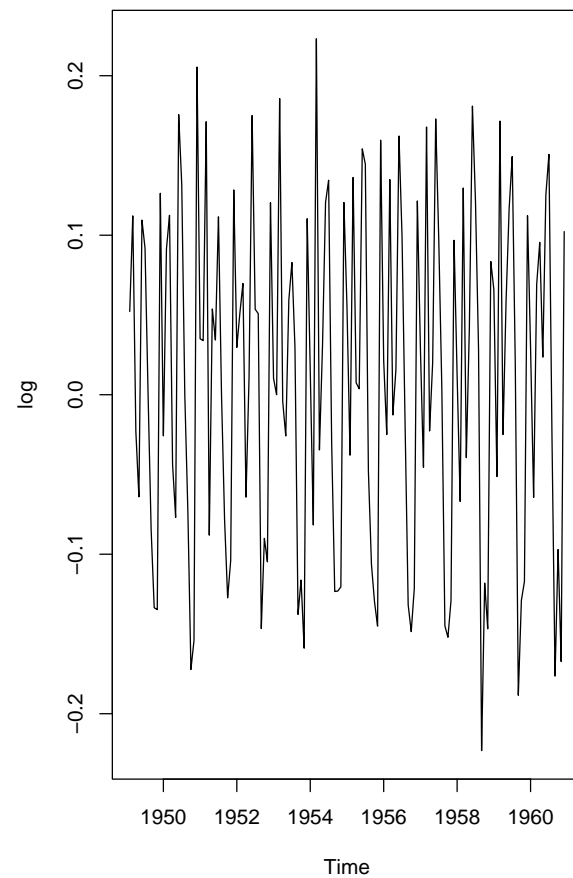
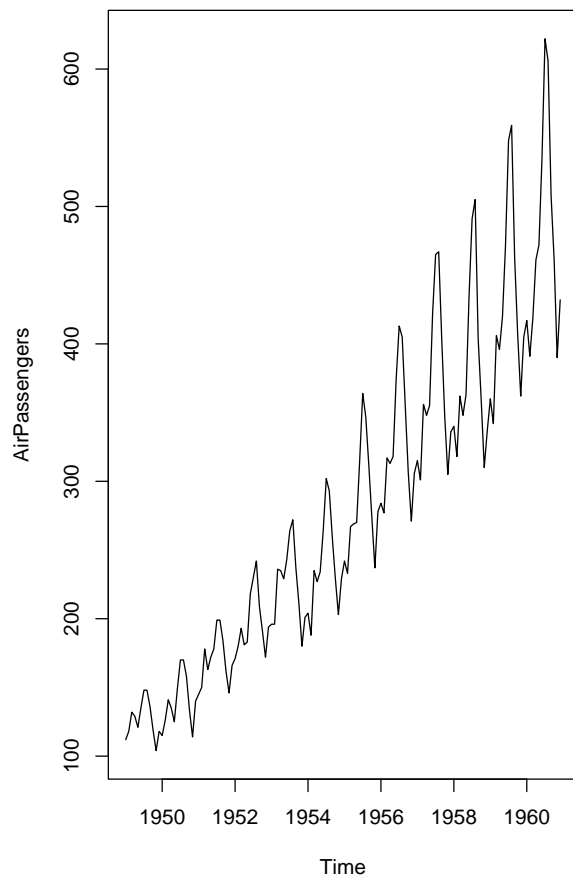
단계 2: 차분 적용 - 평균 정상화

```
par(mfrow = c(1, 2))
ts.plot(AirPassengers)
diff <- diff(AirPassengers)
plot(diff)
```



단계 3: 로그 적용 - 분산 정상화

```
par(mfrow = c(1, 2))
plot(AirPassengers)
log <- diff(log(AirPassengers))
plot(log)
```



실습: 단일 시계열 자료 시각화

단계 1: WWWusage 데이터 셋 가져오기

```
data("WWWusage")
str(WWWusage)
```

```
## Time-Series [1:100] from 1 to 100: 88 84 85 85 84 85 83 85 88 89 ...
```

```
WWWusage
```

```
## Time Series:
```

```
## Start = 1
```

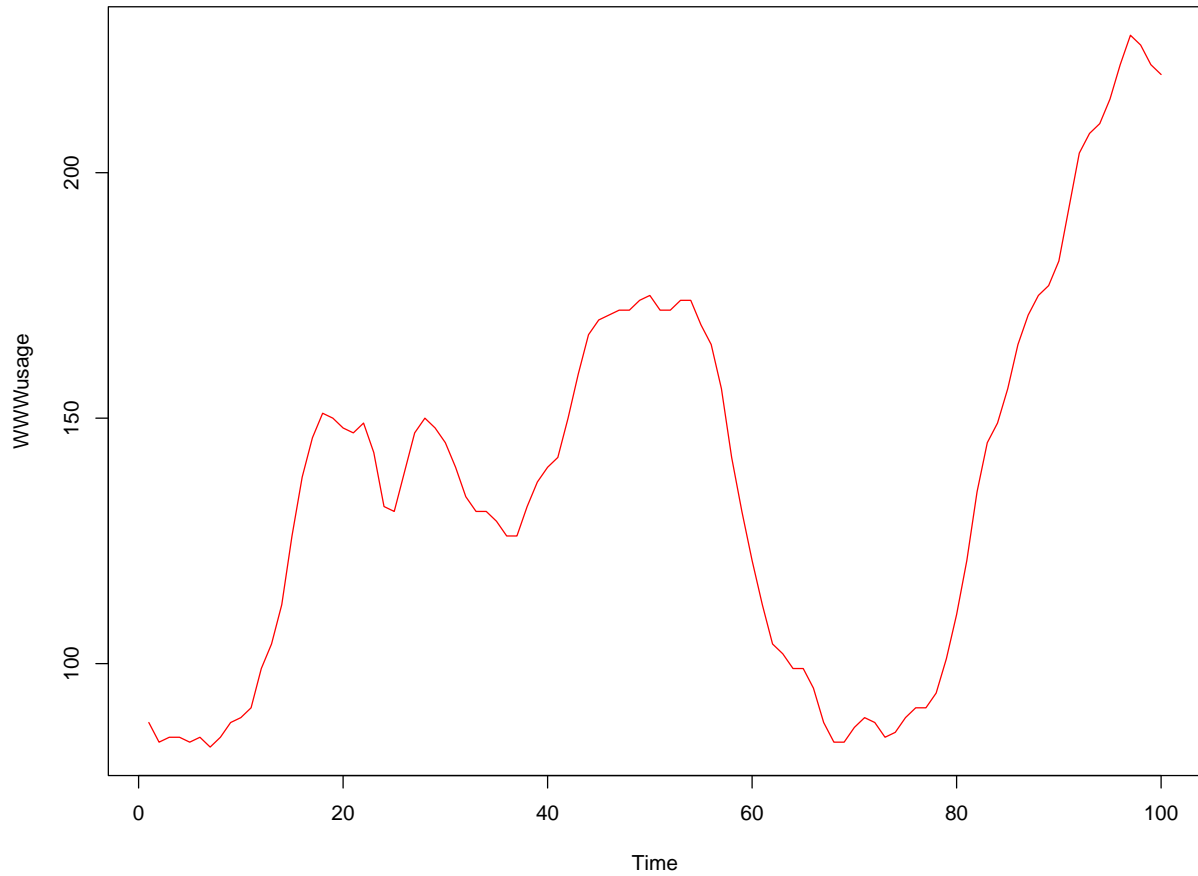
```
## End = 100
```

```
## Frequency = 1
```

```
## [1] 88 84 85 85 84 85 83 85 88 89 91 99 104 112 126 138 146 151
## [19] 150 148 147 149 143 132 131 139 147 150 148 145 140 134 131 131 129 126
## [37] 126 132 137 140 142 150 159 167 170 171 172 172 174 175 172 172 174 174
## [55] 169 165 156 142 131 121 112 104 102 99 99 95 88 84 84 87 89 88
## [73] 85 86 89 91 91 94 101 110 121 135 145 149 156 165 171 175 177 182
## [91] 193 204 208 210 215 222 228 226 222 220
```

단계 2: 시계열 자료 추세선 시각화

```
ts.plot(WWWusage, type = "l", col = "red")
```



실습: 다중 시계열 자료 시각화

단계 1: 데이터 가져오기

```
data(EuStockMarkets)
head(EuStockMarkets)
```

```
##          DAX      SMI      CAC  FTSE
## [1,] 1628.75 1678.1 1772.8 2443.6
## [2,] 1613.63 1688.5 1750.5 2460.2
## [3,] 1606.51 1678.6 1718.0 2448.2
## [4,] 1621.04 1684.1 1708.1 2470.4
## [5,] 1618.16 1686.6 1723.1 2484.7
## [6,] 1610.61 1671.6 1714.3 2466.8
```

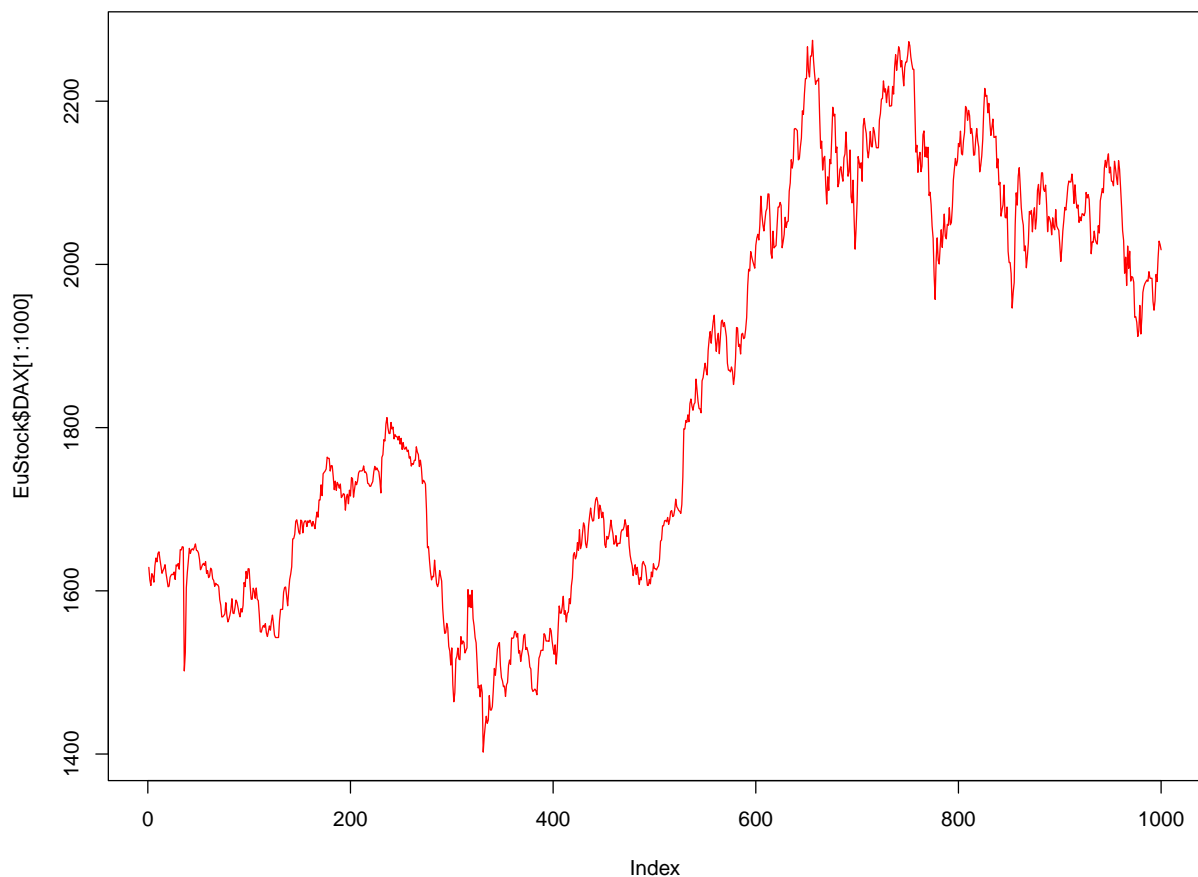
단계 2: 데이터프레임으로 변환

```
EuStock <- data.frame(EuStockMarkets)
head(EuStock)
```

```
##      DAX    SMI    CAC   FTSE
## 1 1628.75 1678.1 1772.8 2443.6
## 2 1613.63 1688.5 1750.5 2460.2
## 3 1606.51 1678.6 1718.0 2448.2
## 4 1621.04 1684.1 1708.1 2470.4
## 5 1618.16 1686.6 1723.1 2484.7
## 6 1610.61 1671.6 1714.3 2466.8
```

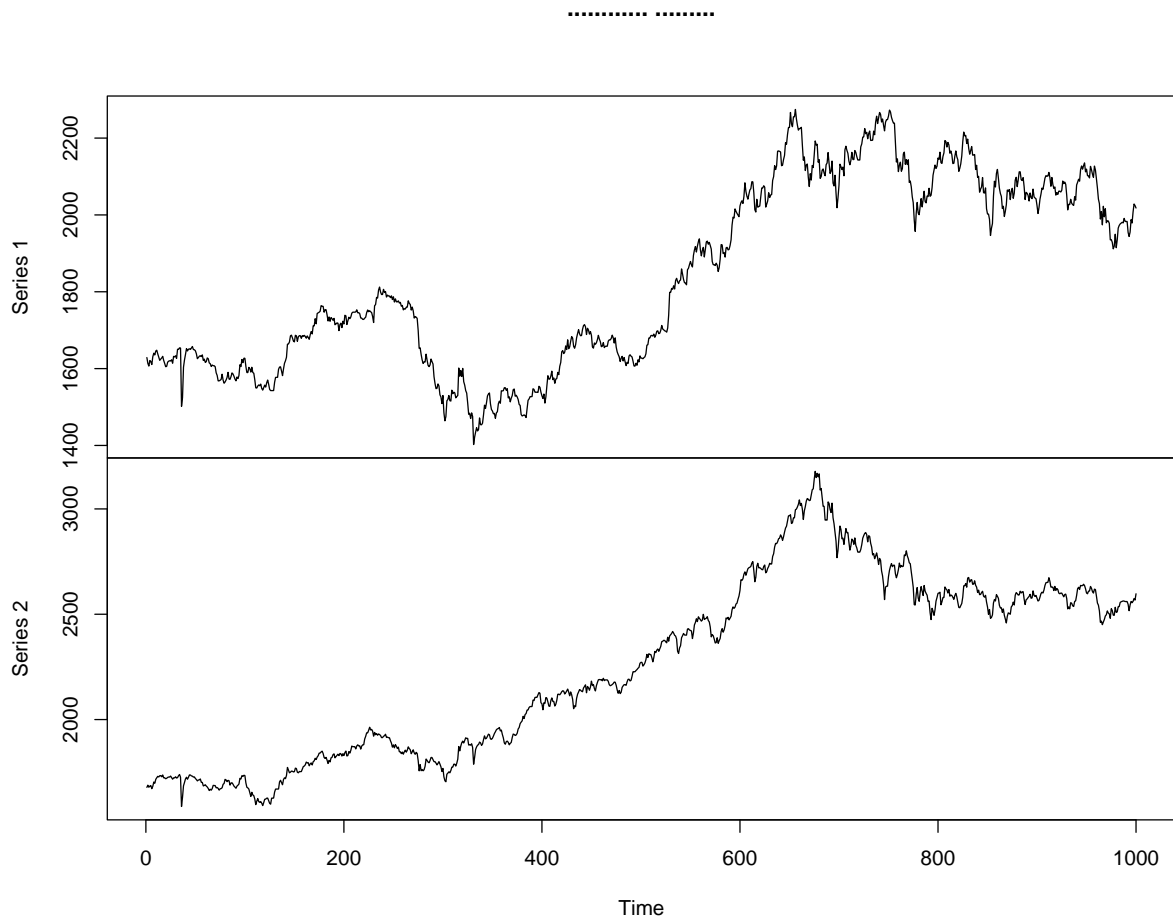
단계 3: 단일 시계열 자료 추세선 시각화(1,000개 데이터 대상)

```
plot(EuStock$DAX[1:1000], type = "l", col = "red")
```



단계 4: 다중 시계열 자료 추세선 시각화(1,000개 데이터 대상)

```
plot.ts(cbind(EuStock$DAX[1:1000], EuStock$SMI[1:1000]), main = "주가지수 추세선")
```



실습: 시계열 요소분해 시각화

단계 1: 시계열 자료 준비

```
data <- c(45, 56, 45, 43, 69, 75, 58, 59, 66, 64, 62, 65,
          55, 49, 67, 55, 71, 78, 71, 65, 69, 43, 70, 75,
          56, 56, 65, 55, 82, 85, 75, 77, 77, 69, 79, 89)
length(data)
```

```
## [1] 36
```

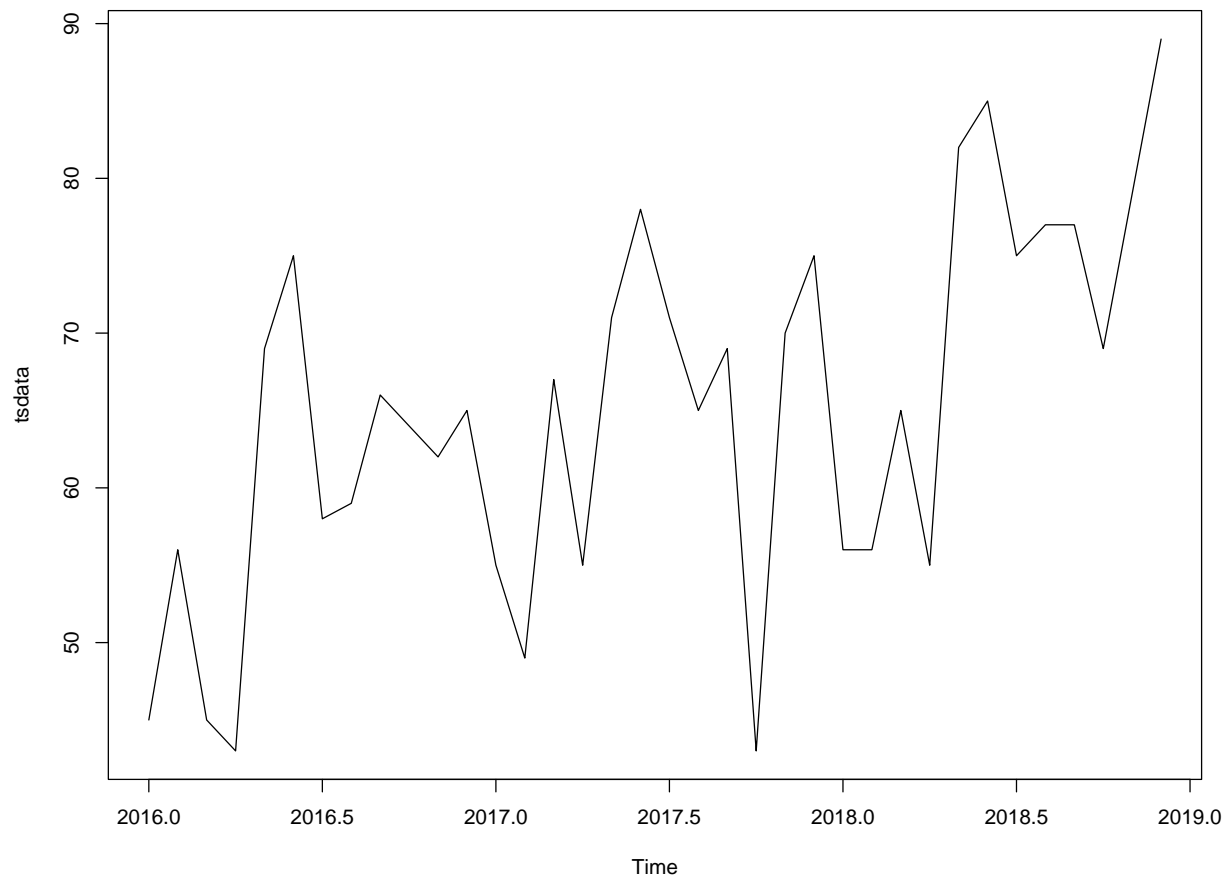
단계 2: 시계열 자료 생성 - 시계열 자료 형식으로 객체 생성

```
tsdata <- ts(data, start = c(2016, 1), frequency = 12)
tsdata
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 2016  45  56  45  43  69  75  58  59  66  64  62  65
## 2017  55  49  67  55  71  78  71  65  69  43  70  75
## 2018  56  56  65  55  82  85  75  77  77  69  79  89
```

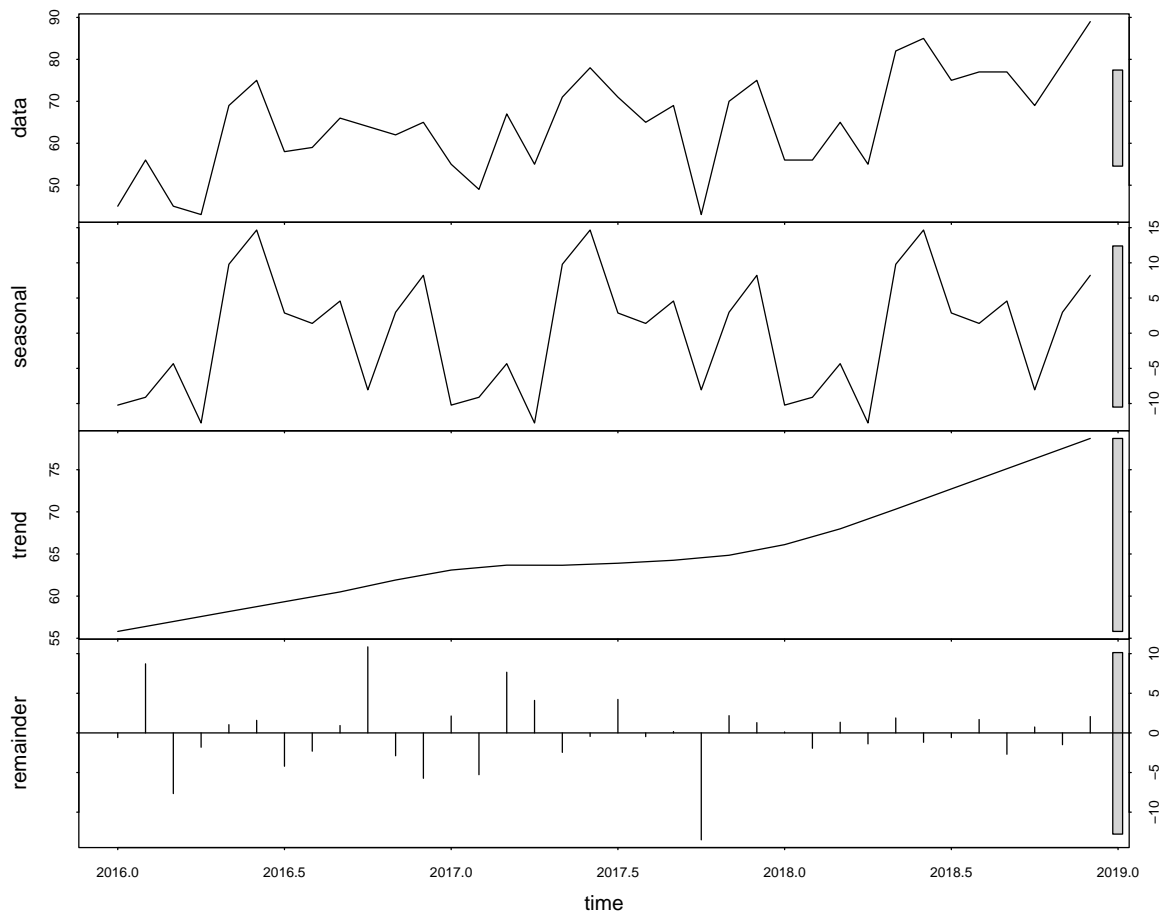
단계 3: 추세선 확인 - 각 요인(추세, 순환, 계절, 불규칙)을 시각적으로 확인

```
ts.plot(tsdata)
```



단계 4: 시계열 분해

```
plot(stl(tsdata, "periodic"))
```

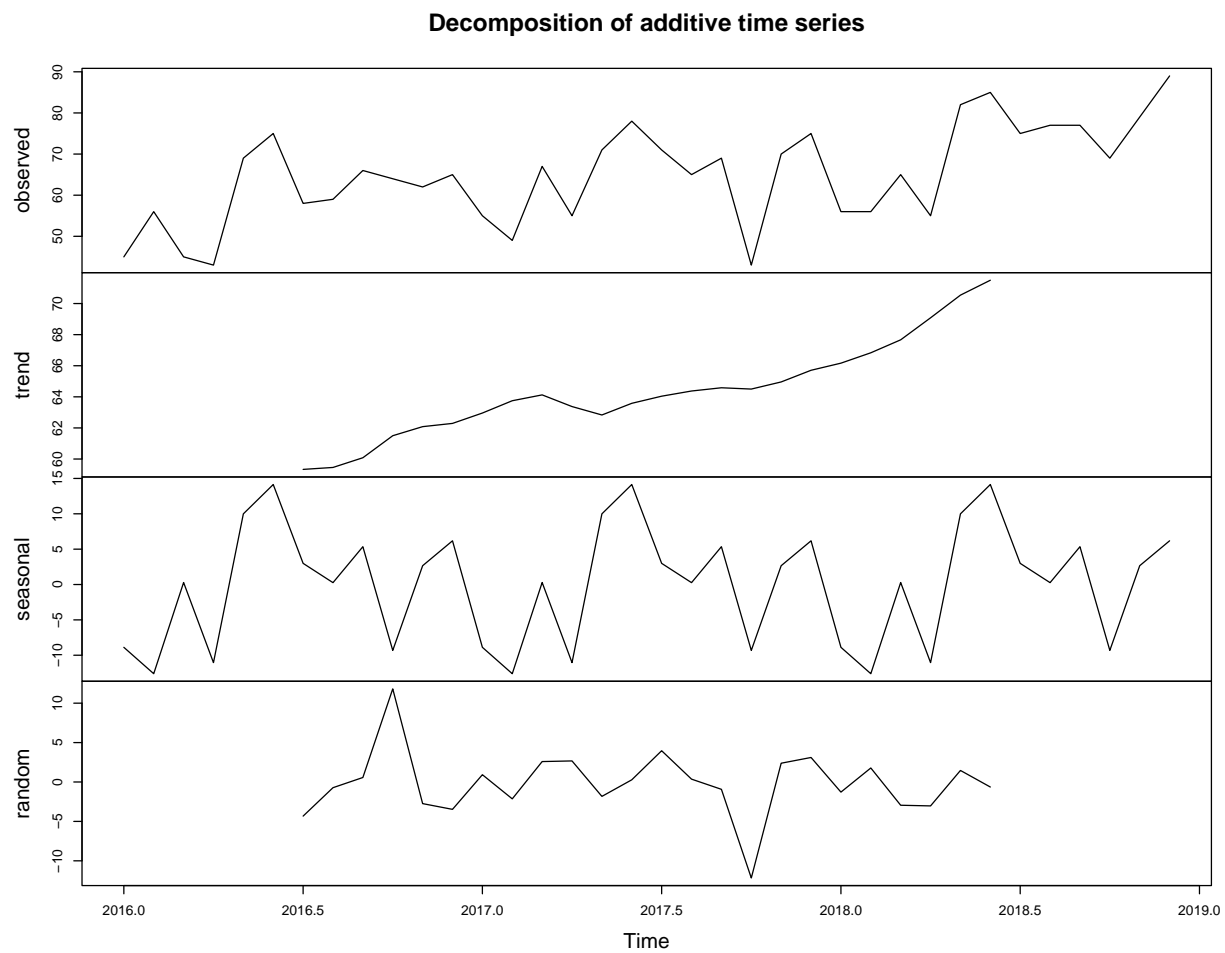



단계 5: 시계열 분해와 변동요인 제거

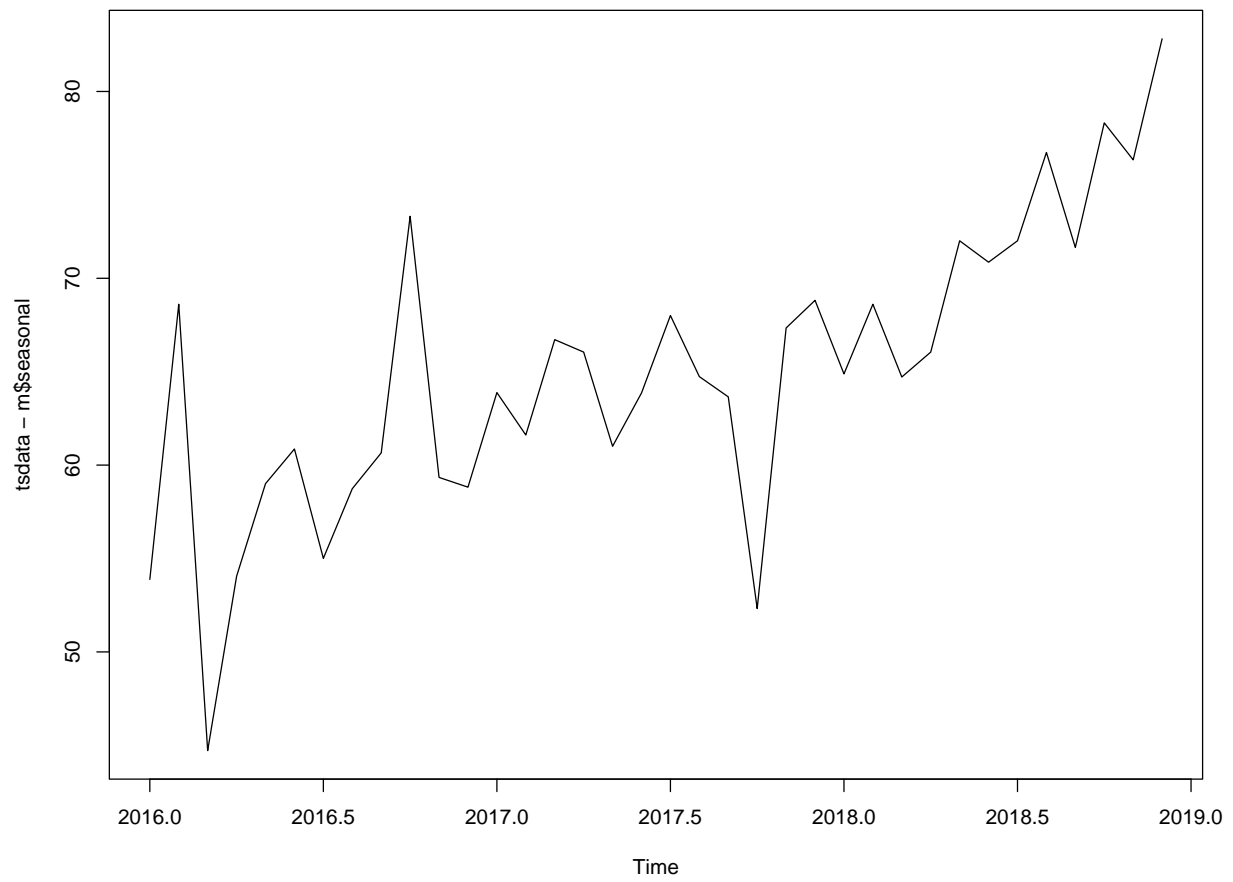
```
m <- decompose(tsddata)
attributes(m)
```

```
## $names
## [1] "x"          "seasonal" "trend"     "random"    "figure"    "type"
##
## $class
## [1] "decomposed.ts"
```

```
plot(m)
```

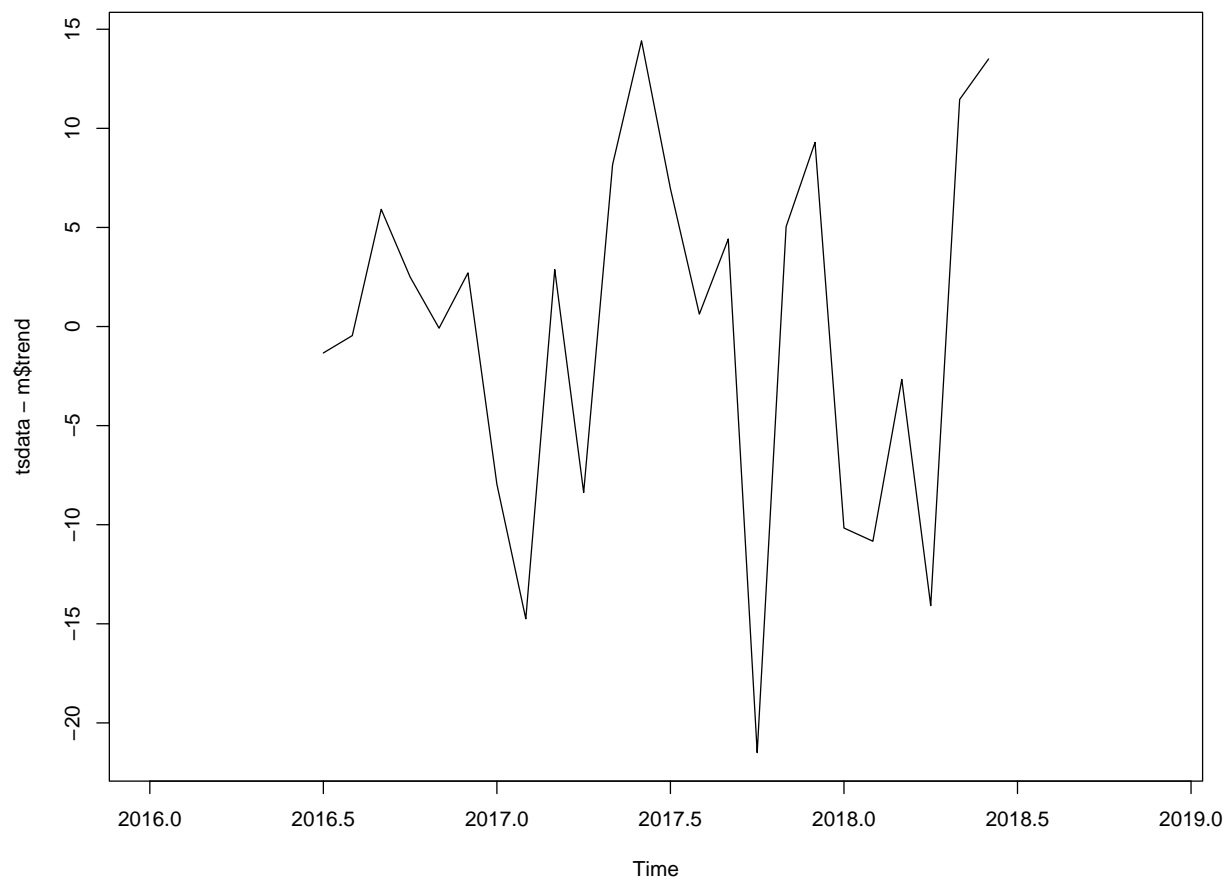


```
par(mfrow = c(1, 1))  
plot(tsddata - m$seasonal)
```

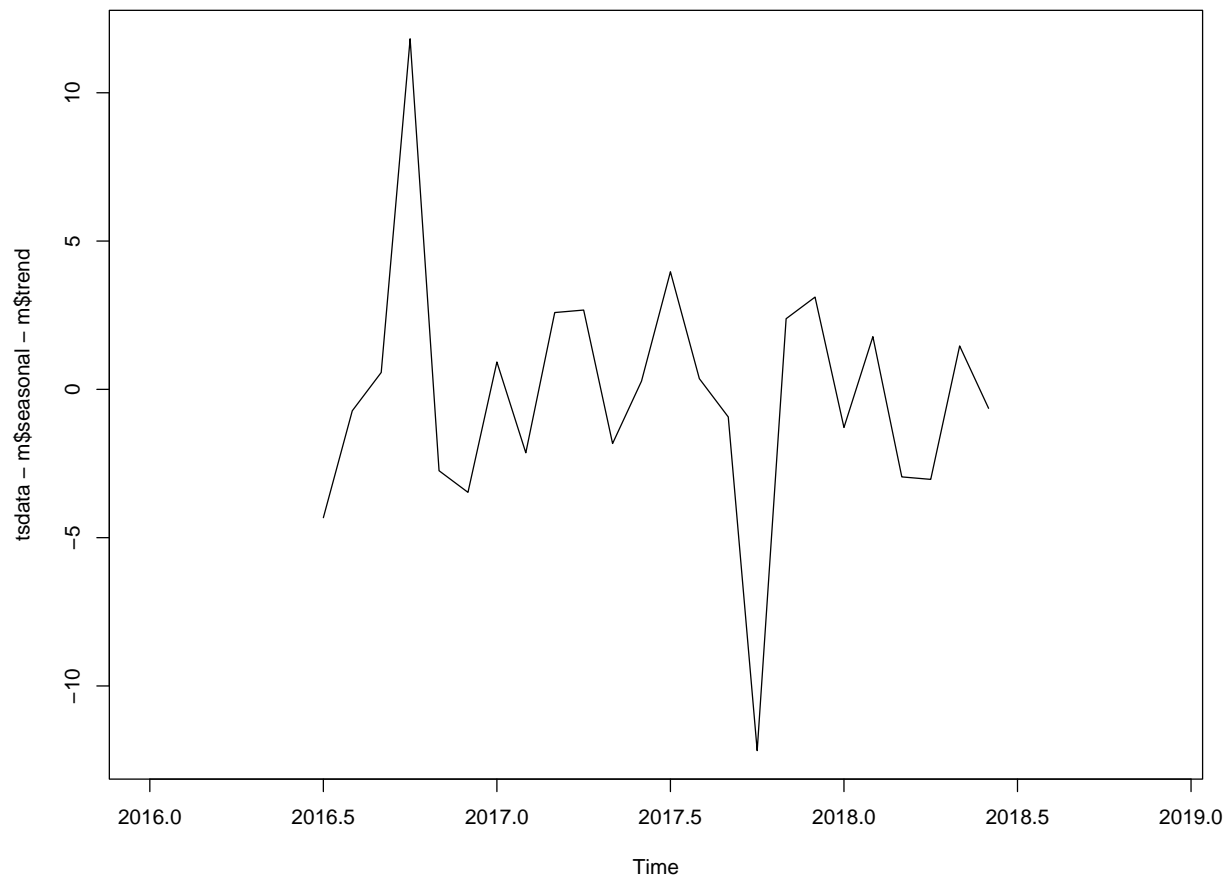


단계 6: 추세요인과 불규칙요인 제거

```
plot(tsdata - m$trend)
```



```
plot(tsdata - m$seasonal - m$trend)
```



실습: 시계열 요소 분해 시각화

단계 1: 시계열 자료 생성

```
input <- c(3180, 3000, 3200, 3100, 3300, 3200,
           3400, 3550, 3200, 3400, 3300, 3700)
length(input)
```

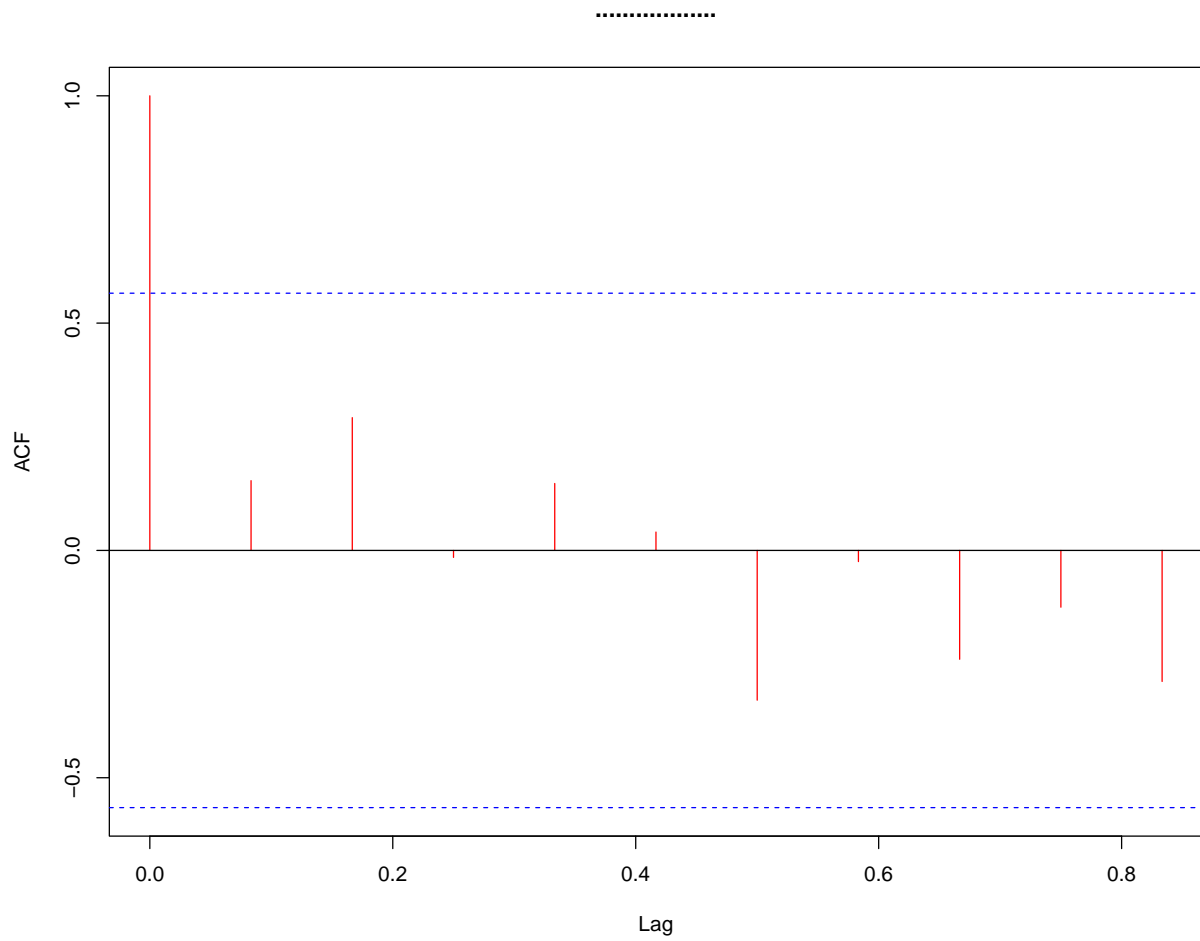
```
## [1] 12
```

```
tsdata <- ts(input, start = c(2015, 2), frequency = 12)
tsdata
```

```
##      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2015    3180 3000 3200 3100 3300 3200 3400 3550 3200 3400 3300
## 2016 3700
```

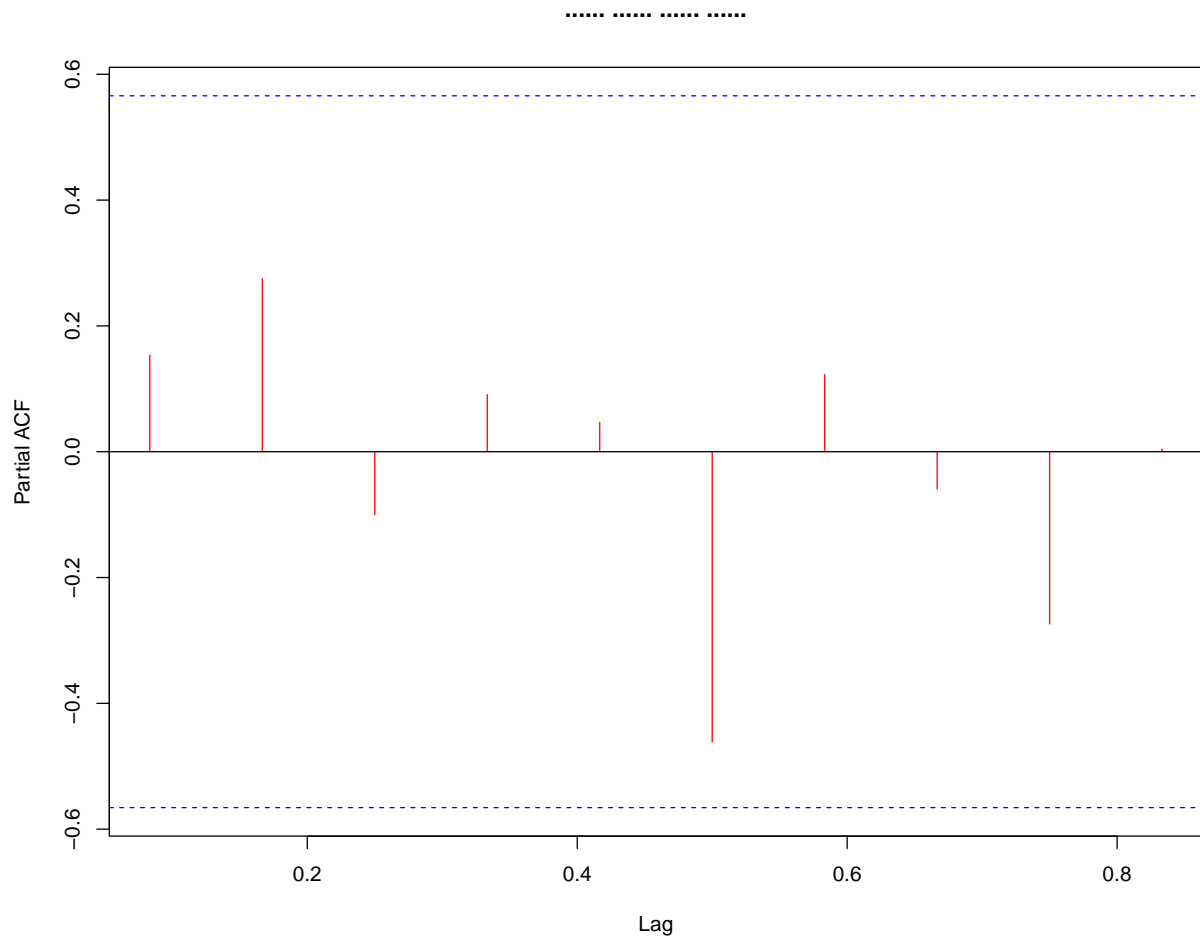
단계 2: 자기 상관 함수 시각화

```
acf(na.omit(tsdata), main = "자기상관함수", col = "red")
```



단계 3: 부분 자기 상관 함수 시각화

```
pacf(na.omit(tsddata), main = "부분 자기 상관 함수", col = "red")
```



실습: 시계열 자료의 추세 패턴 찾기 시각화

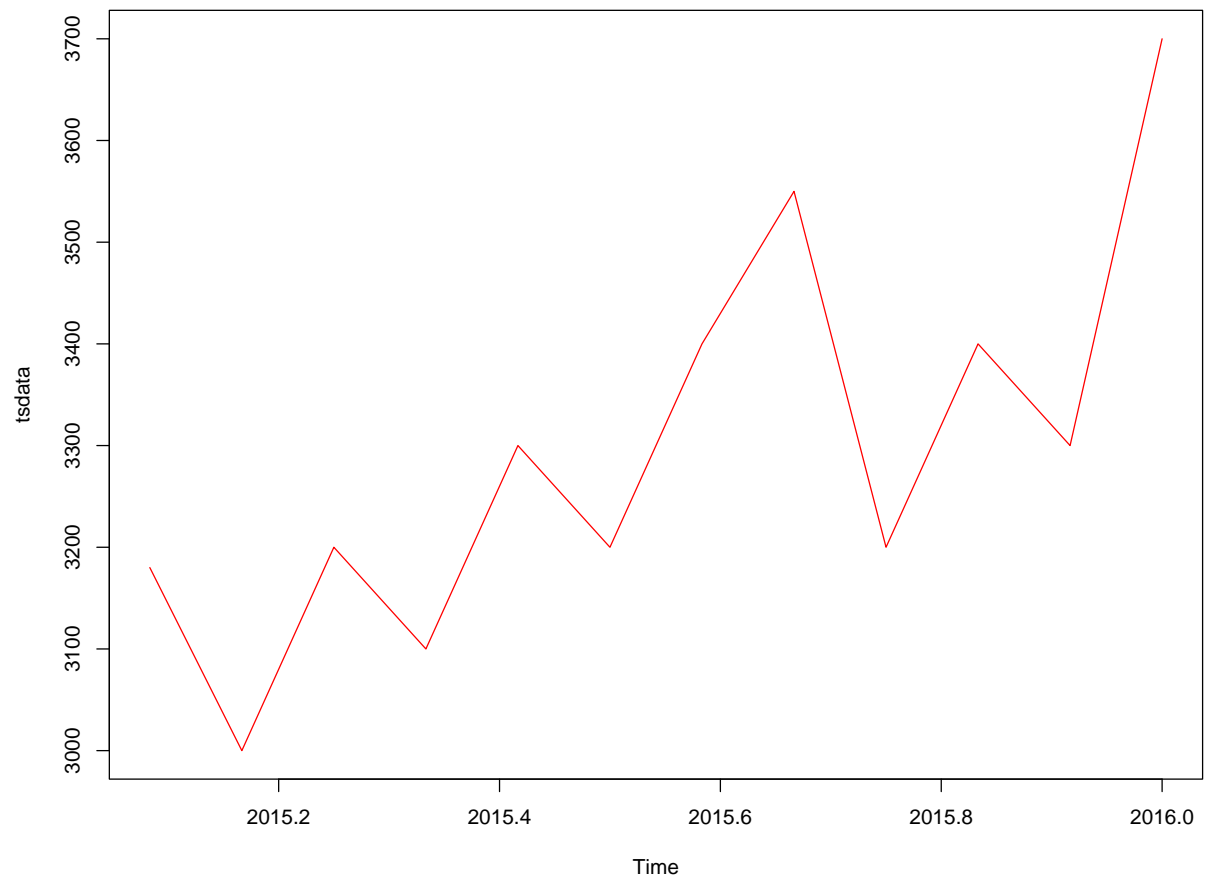
단계 1: 시계열 자료 생성

```
input <- c(3180, 3000, 3200, 3100, 3300, 3200,
           3400, 3550, 3200, 3400, 3300, 3700)
input
```

```
## [1] 3180 3000 3200 3100 3300 3200 3400 3550 3200 3400 3300 3700
```

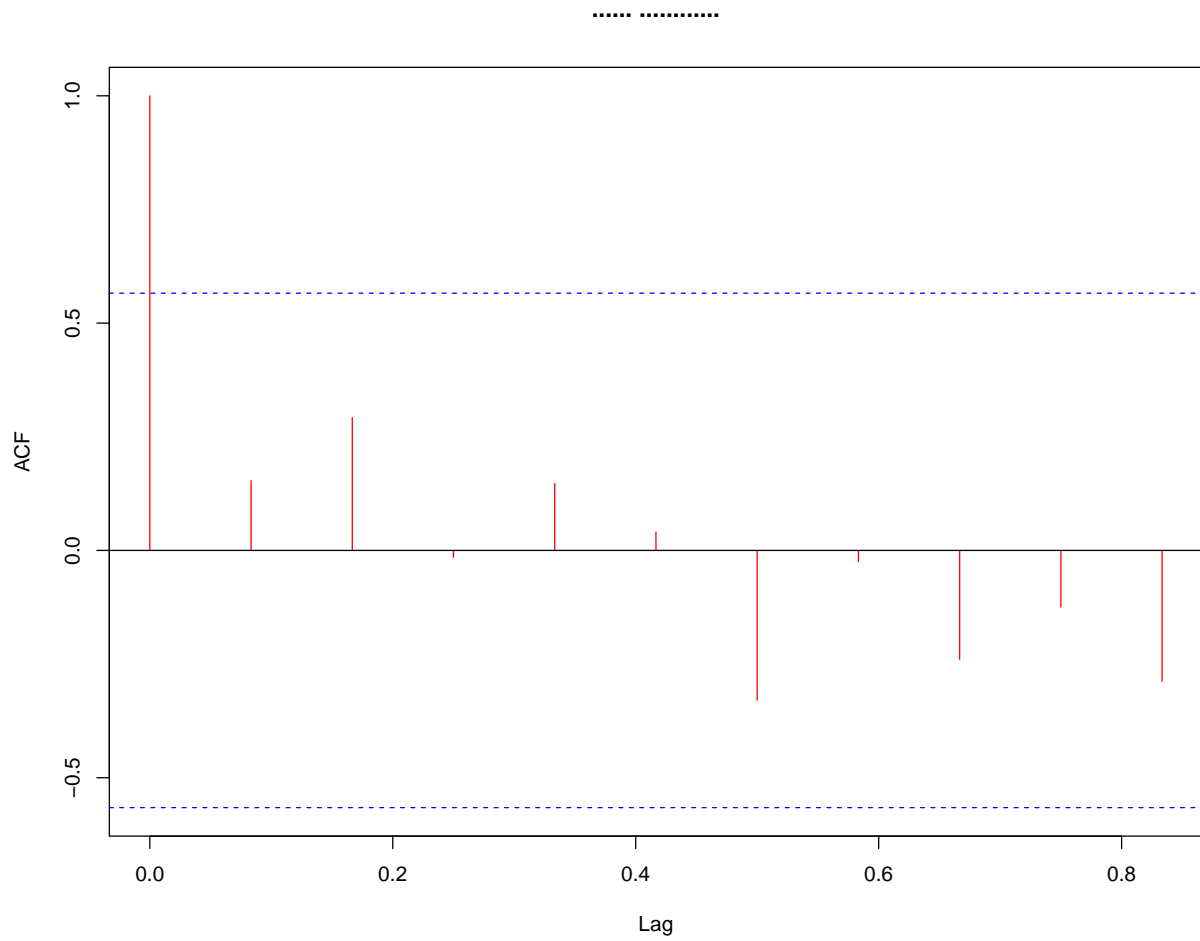
단계 2: 추세선 시각화

```
plot(tsddata, type = "l", col = "red")
```



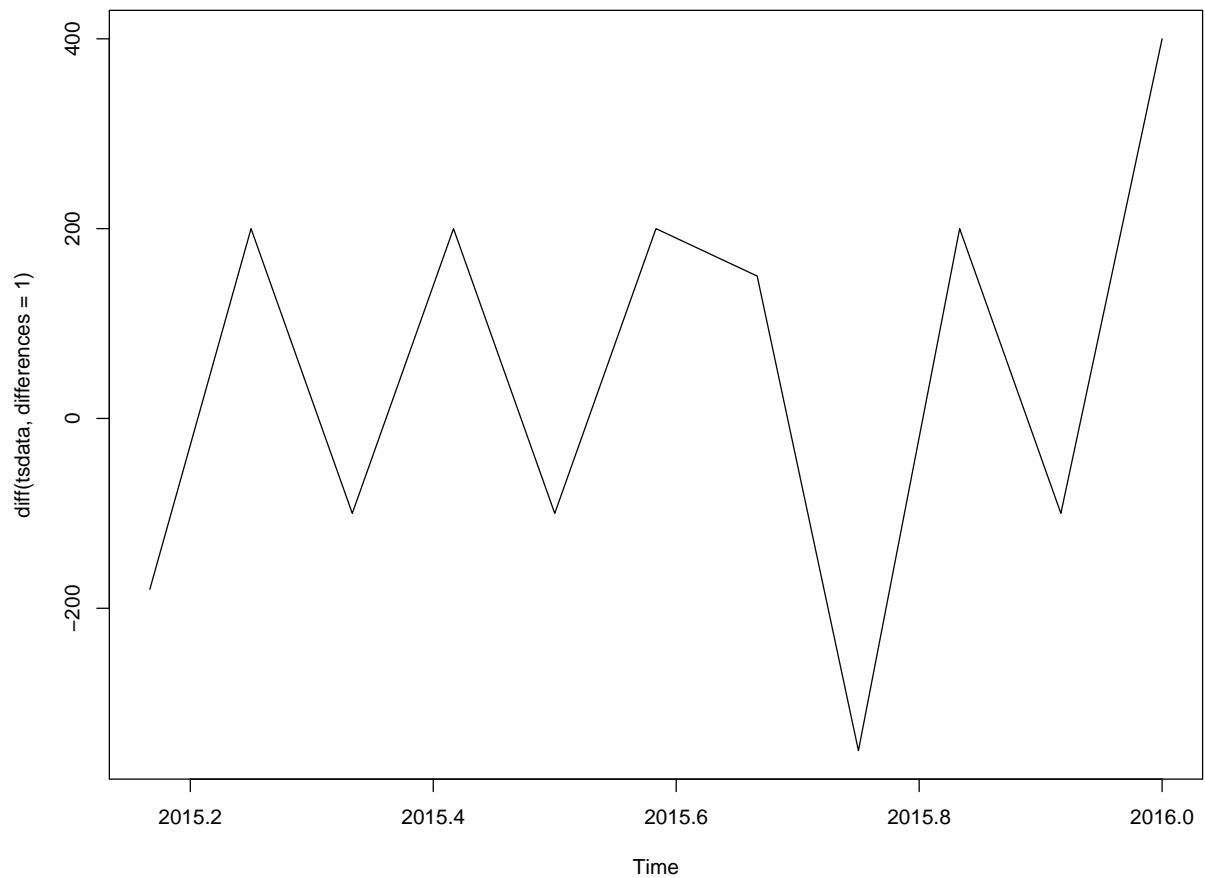
단계 3: 자기 상관 함수 시각화

```
acf(na.omit(tsdata), main = "자기 상환함수", col = "red")
```

단계 4: 차분 시각화

```
plot(diff(tsdata, differences = 1))
```



실습: 이동평균법을 이용한 평활하기

단계 1: 시계열 자료 생성

```
data <- c(45, 56, 45, 43, 69, 75, 58, 59, 66, 64, 62, 65,
          55, 49, 67, 55, 71, 78, 71, 65, 69, 43, 70, 75,
          56, 56, 65, 55, 82, 85, 75, 77, 77, 69, 79, 89)
length(data)
```

```
## [1] 36
```

```
tsdata <- ts(data, start = c(2016, 1), frequency = 12)
tsdata
```

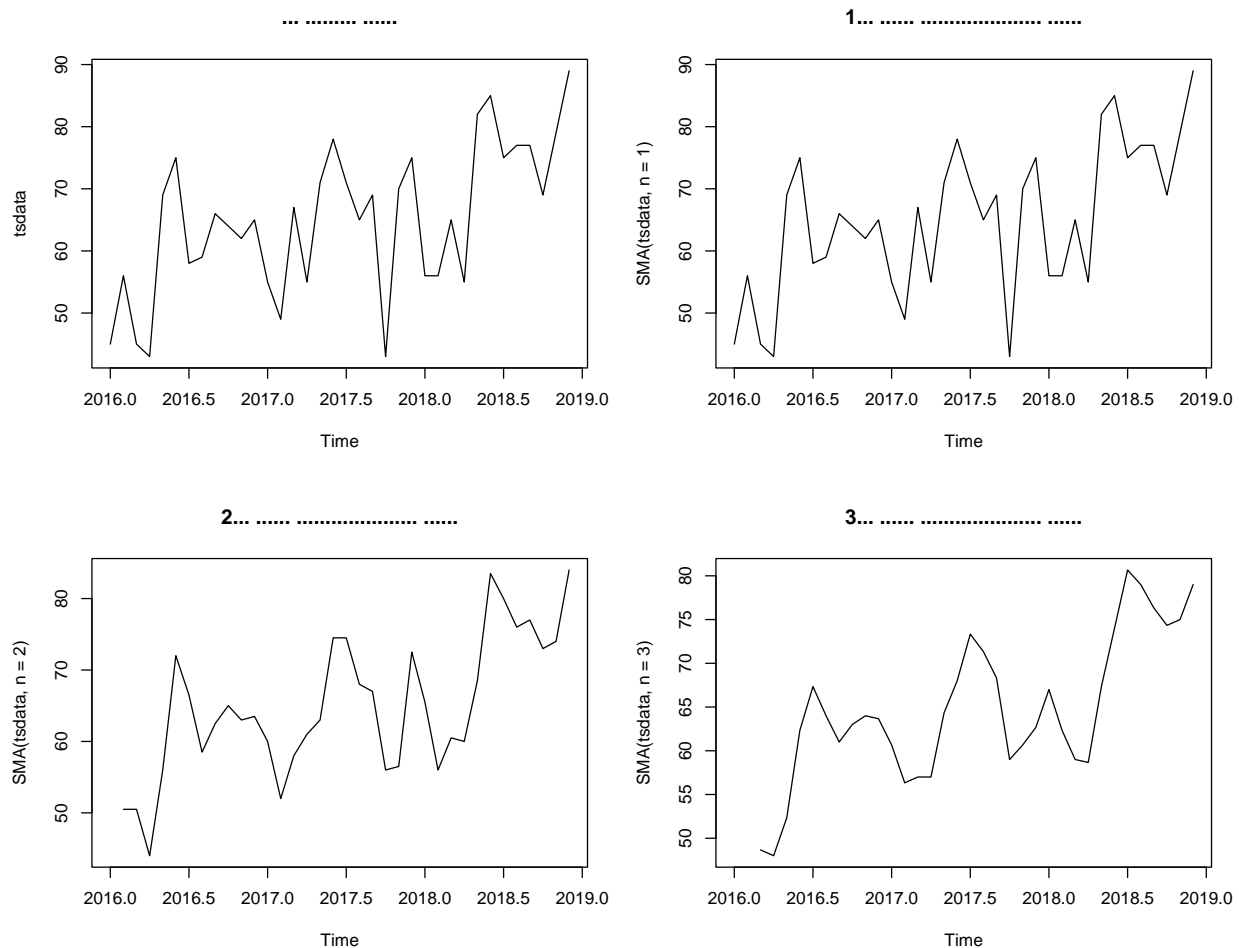
```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 2016  45  56  45  43  69  75  58  59  66  64  62  65
## 2017  55  49  67  55  71  78  71  65  69  43  70  75
## 2018  56  56  65  55  82  85  75  77  77  69  79  89
```

단계 2: 평활 관련 패키지 설치

```
library(TTR)
```

단계 3: 이동평균법으로 평활 및 시각화

```
par(mfrow = c(2, 2))
plot(tsddata, main = "원 시계열 자료")
plot(SMA(tsddata, n = 1), main = "1년 단위 이동평균법으로 평활")
plot(SMA(tsddata, n = 2), main = "2년 단위 이동평균법으로 평활")
plot(SMA(tsddata, n = 3), main = "3년 단위 이동평균법으로 평활")
```



실습: 계절성이 없는 정상성 시계열분석

단계 1: 시계열 자료 특성 분석

단계 1-1: 데이터 준비

```
input <- c(3180, 3000, 3200, 3100, 3300, 3200,
           3400, 3550, 3200, 3400, 3300, 3700)
```

```
input
```

```
## [1] 3180 3000 3200 3100 3300 3200 3400 3550 3200 3400 3300 3700
```

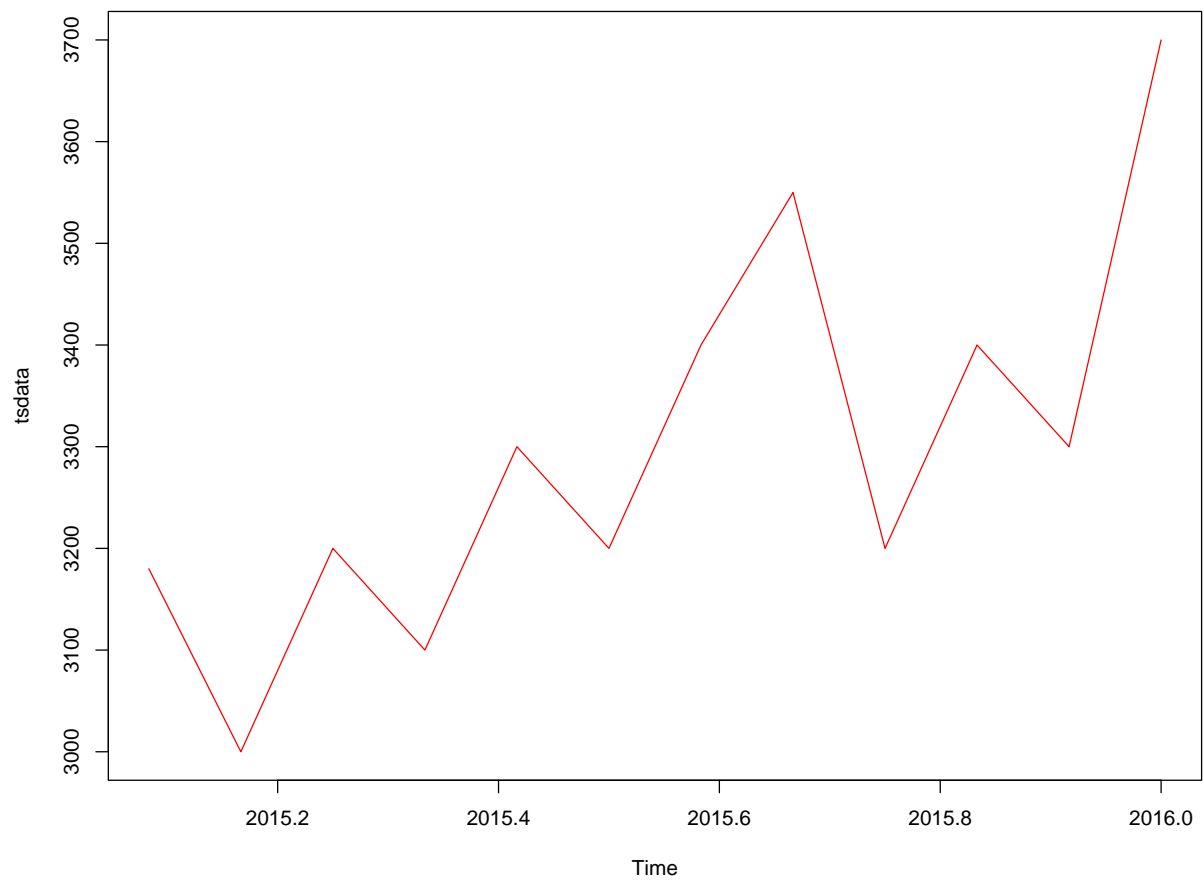
단계 1-2: 시계열 객체 생성(12개월: 2015 2월 ~ 2016년 1월)

```
tsdata <- ts(input, start = c(2015, 2), frequency = 12)
tsdata
```

```
##      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2015      3180 3000 3200 3100 3300 3200 3400 3550 3200 3400 3300
## 2016 3700
```

단계 1-3: 추세선 시각화

```
plot(tsdata, type = "l", col = "red")
```

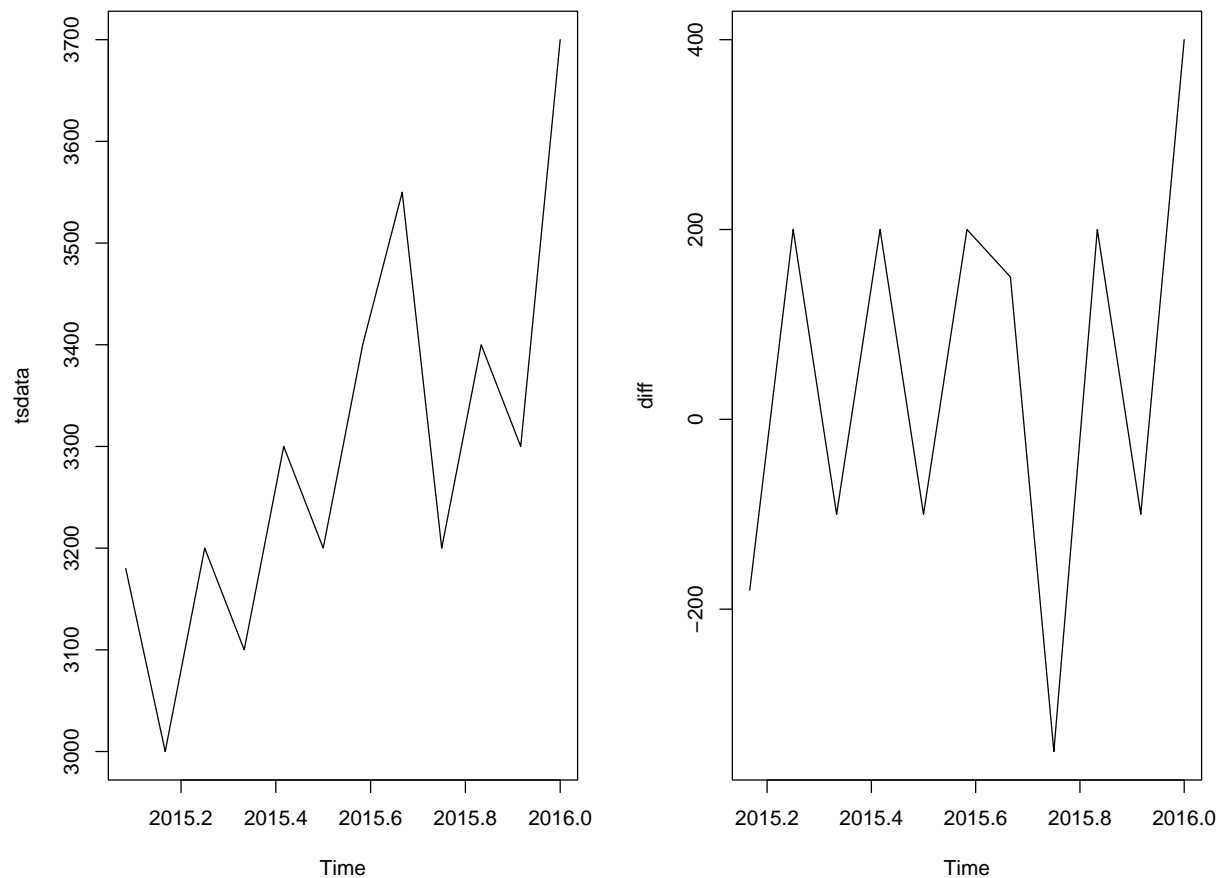


단계 2: 정상성 시계열 변환

```

par(mfrow = c(1, 2))
ts.plot(tsddata)
diff <- diff(tsddata)
plot(diff)

```



단계 3: 모형 식별과 추정

```

library(forecast)
arima <- auto.arima(tsddata)
arima

```

```

## Series: tsdata
## ARIMA(1,1,0)
##
## Coefficients:
##      ar1
##      -0.6891
## s.e.    0.2451
##
## sigma^2 = 31644: log likelihood = -72.4

```

```
## AIC=148.8   AICc=150.3   BIC=149.59
```

단계 4: 모형 생성

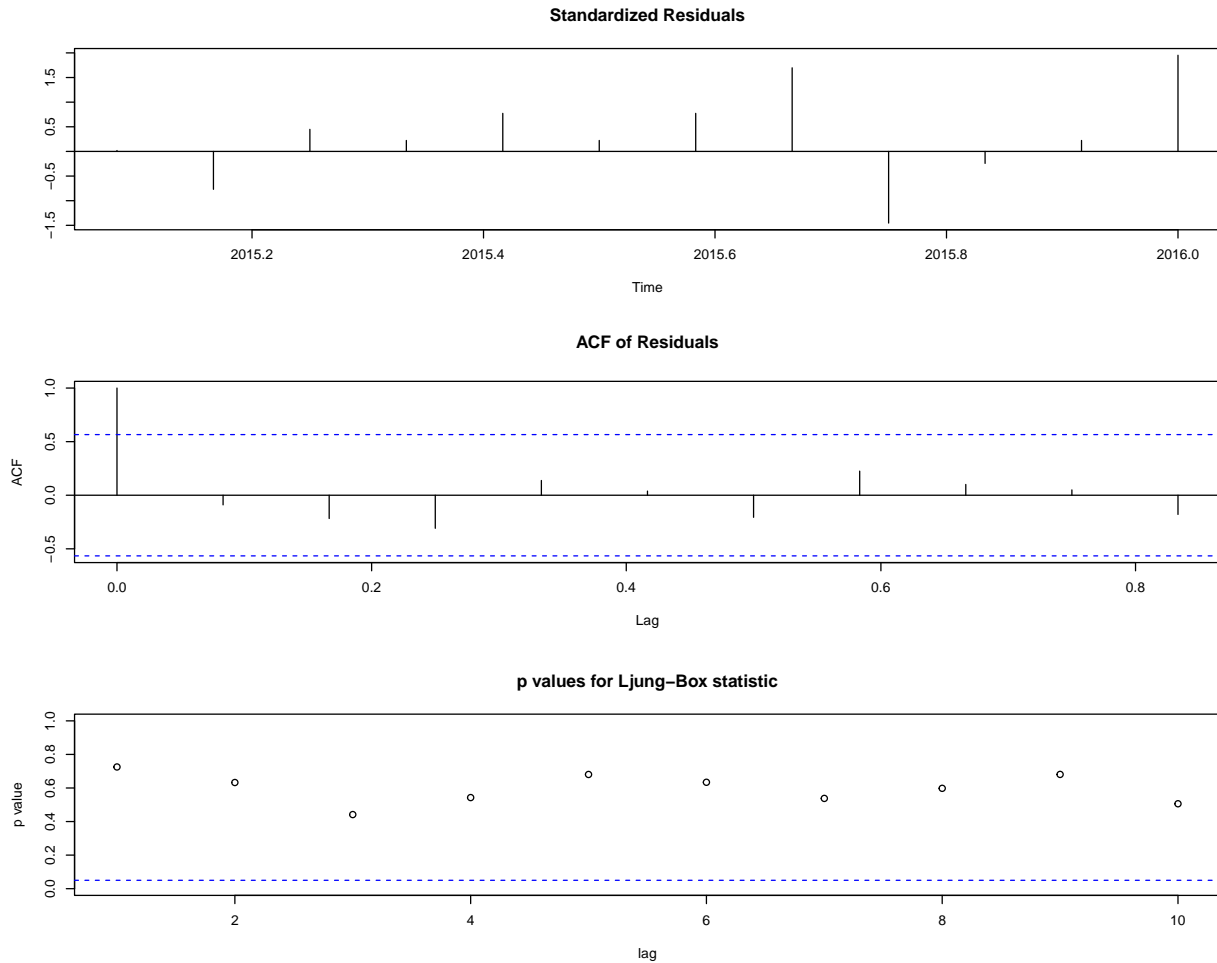
```
model <- arima(tsdata, order = c(1, 1, 0))  
model
```

```
##  
## Call:  
## arima(x = tsdata, order = c(1, 1, 0))  
##  
## Coefficients:  
##          ar1  
##       -0.6891  
## s.e.    0.2451  
##  
## sigma^2 estimated as 28767:  log likelihood = -72.4,  aic = 148.8
```

단계 5: 모형 진단(모형의 타당성 검토)

단계 5-1: 자기 상관 함수에 의한 모형 진단

```
tsdiag(model)
```



단계 5-2: Box_Ljung에 의한 잔차항 모형 진단

```
Box.test(model$residuals, lag = 1, type = "Ljung")
```

```
##
## Box-Ljung test
##
## data: model$residuals
## X-squared = 0.12353, df = 1, p-value = 0.7252
```

단계 6: 미래 예측(업무 적용)

```
fore <- forecast(model)
fore
```

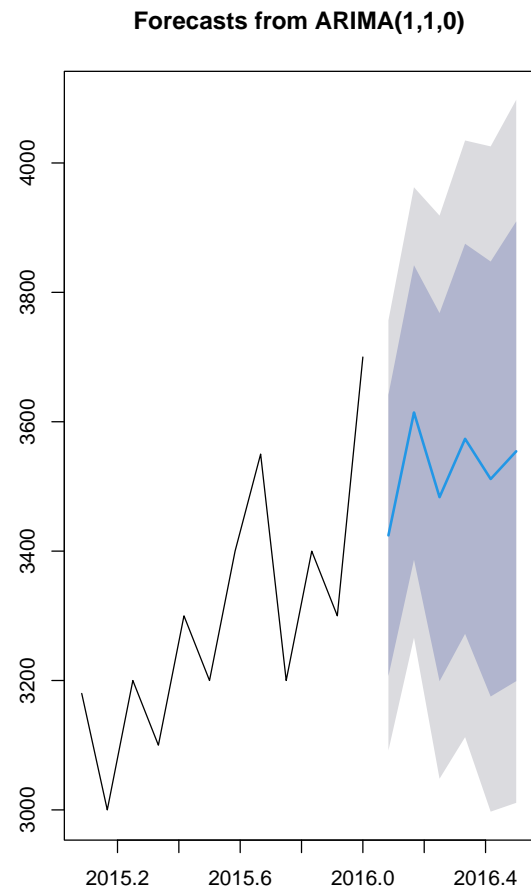
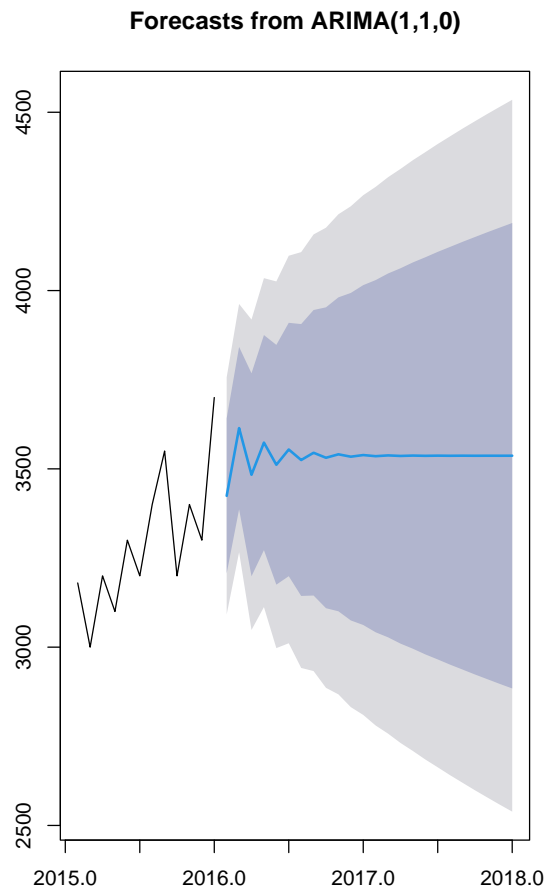
```
##           Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Feb 2016      3424.367 3207.007 3641.727 3091.944 3756.791
## Mar 2016      3614.301 3386.677 3841.925 3266.180 3962.421
## Apr 2016      3483.421 3198.847 3767.995 3048.203 3918.639
## May 2016      3573.608 3272.084 3875.131 3112.467 4034.748
## Jun 2016      3511.462 3175.275 3847.649 2997.308 4025.615
## Jul 2016      3554.286 3199.003 3909.568 3010.928 4097.643
```

## Aug 2016	3524.776	3143.569	3905.984	2941.770	4107.783
## Sep 2016	3545.111	3144.813	3945.408	2932.908	4157.313
## Oct 2016	3531.099	3109.224	3952.974	2885.897	4176.301
## Nov 2016	3540.754	3100.585	3980.923	2867.574	4213.934
## Dec 2016	3534.101	3074.901	3993.300	2831.816	4236.385
## Jan 2017	3538.685	3062.192	4015.179	2809.951	4267.420
## Feb 2017	3535.526	3041.695	4029.357	2780.277	4290.775
## Mar 2017	3537.703	3027.557	4047.849	2757.502	4317.904
## Apr 2017	3536.203	3009.958	4062.448	2731.381	4341.025
## May 2017	3537.237	2995.565	4078.908	2708.822	4365.651
## Jun 2017	3536.524	2979.724	4093.325	2684.972	4388.077
## Jul 2017	3537.015	2965.573	4108.457	2663.070	4410.960
## Aug 2017	3536.677	2950.901	4122.453	2640.809	4432.545
## Sep 2017	3536.910	2937.181	4136.639	2619.704	4454.116
## Oct 2017	3536.749	2923.359	4150.140	2598.650	4474.849
## Nov 2017	3536.860	2910.124	4163.596	2578.350	4495.371
## Dec 2017	3536.784	2896.968	4176.600	2558.270	4515.298
## Jan 2018	3536.836	2884.211	4189.462	2538.732	4534.941

```

par(mfrow = c(1, 2))
plot(fore)
model2 <- forecast(model, h = 6)
plot(model2)

```

실습: 계절성을 갖는 정상성 시계열분석

단계 1: 시계열 자료 특성 분석

단계 1-1: 데이터 준비

```
data <- c(55, 56, 45, 43, 69, 75, 58, 59, 66, 64, 62, 65,
          55, 49, 67, 55, 71, 78, 61, 65, 69, 53, 70, 75,
          56, 56, 65, 55, 68, 80, 65, 67, 77, 69, 79, 82,
          57, 55, 63, 60, 68, 70, 58, 65, 70, 55, 65, 70)
length(data)
```

```
## [1] 48
```

단계 1-2: 시계열 자료 생성

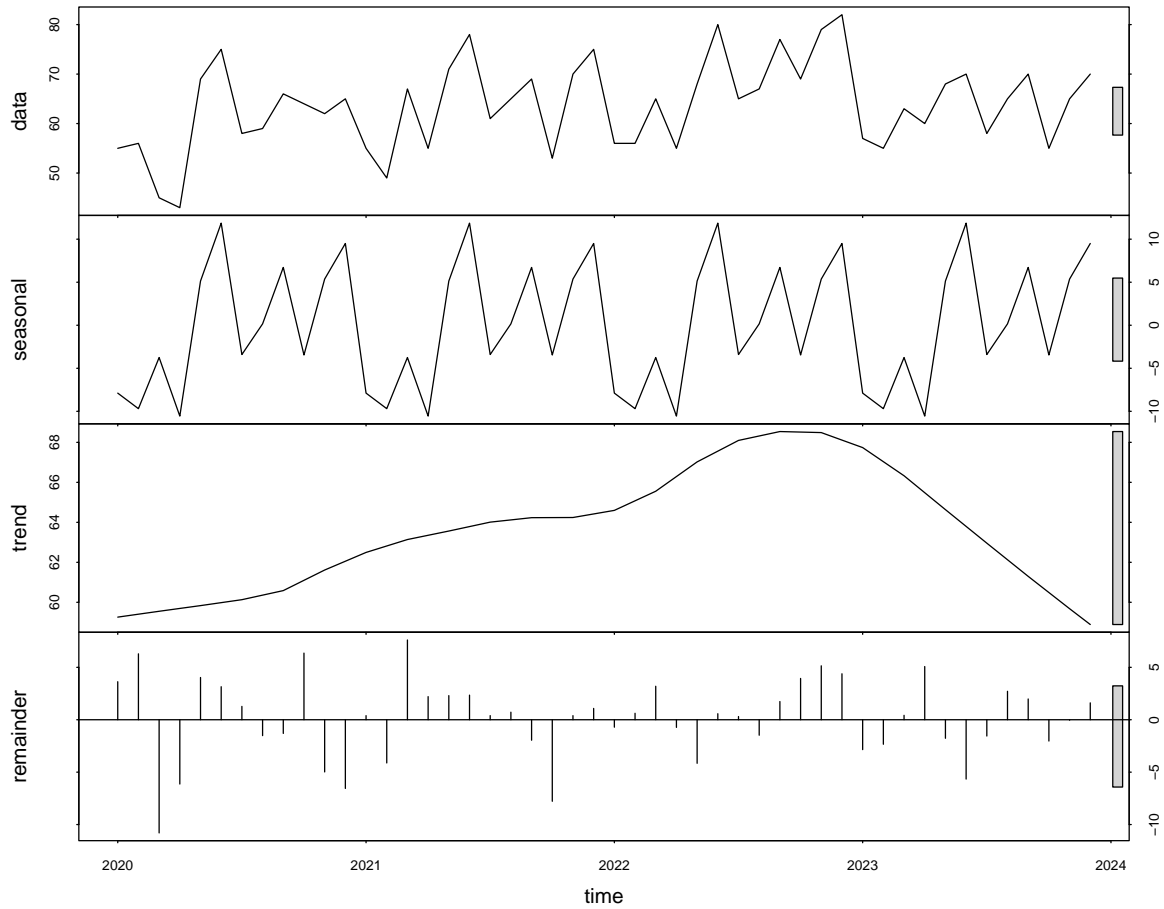
```
tsdata <- ts(data, start = c(2020, 1), frequency = 12)
tsdata
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 2020  55  56  45  43  69  75  58  59  66  64  62  65
## 2021  55  49  67  55  71  78  61  65  69  53  70  75
```

```
## 2022 56 56 65 55 68 80 65 67 77 69 79 82
## 2023 57 55 63 60 68 70 58 65 70 55 65 70
```

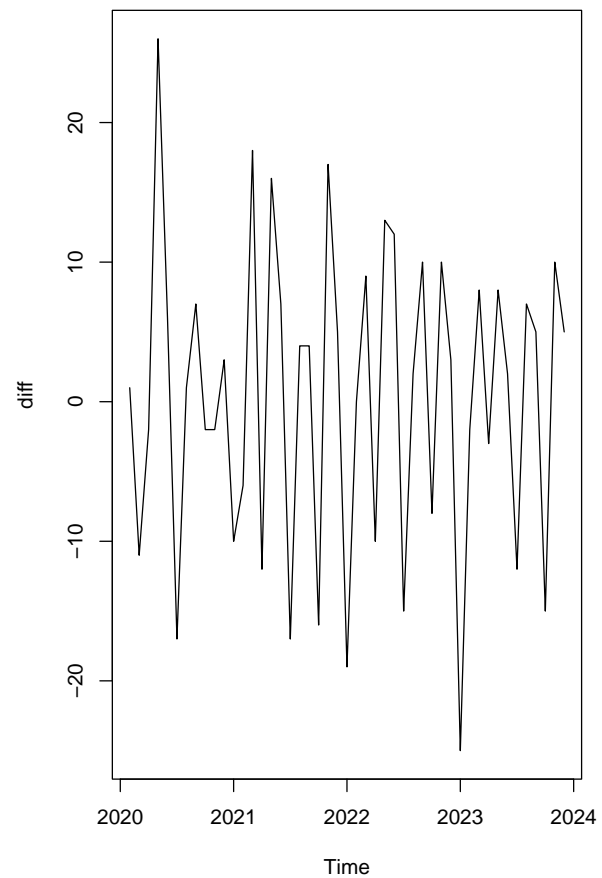
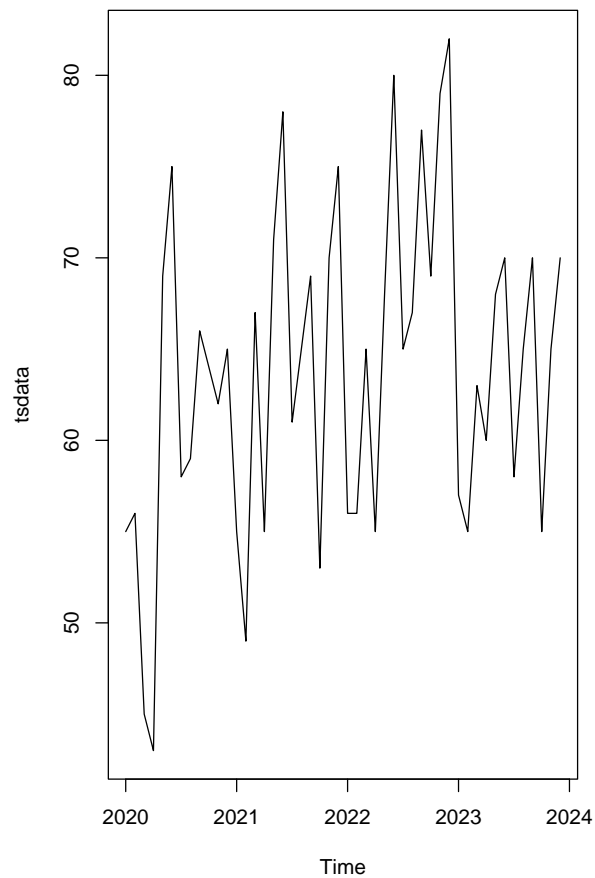
단계 1-3: 시계열 요소 분해 시각화

```
ts_feature <- stl(tsdata, s.window = "periodic")
plot(ts_feature)
```



단계 2: 정상성 시계열 변환

```
par(mfrow = c(1, 2))
ts.plot(tsdata)
diff <- diff(tsdata)
plot(diff)
```



단계 3: 모형 식별과 추정

```
ts_model2 <- auto.arima(tsdata)
ts_model2
```

```
## Series: tsdata
## ARIMA(0,1,1)(1,1,0)[12]
##
## Coefficients:
##          ma1      sar1
##      -0.6580 -0.5317
## s.e.   0.1421  0.1754
##
## sigma^2 = 41.97: log likelihood = -116.31
## AIC=238.62  AICc=239.4  BIC=243.29
```

단계 4: 모형 생성

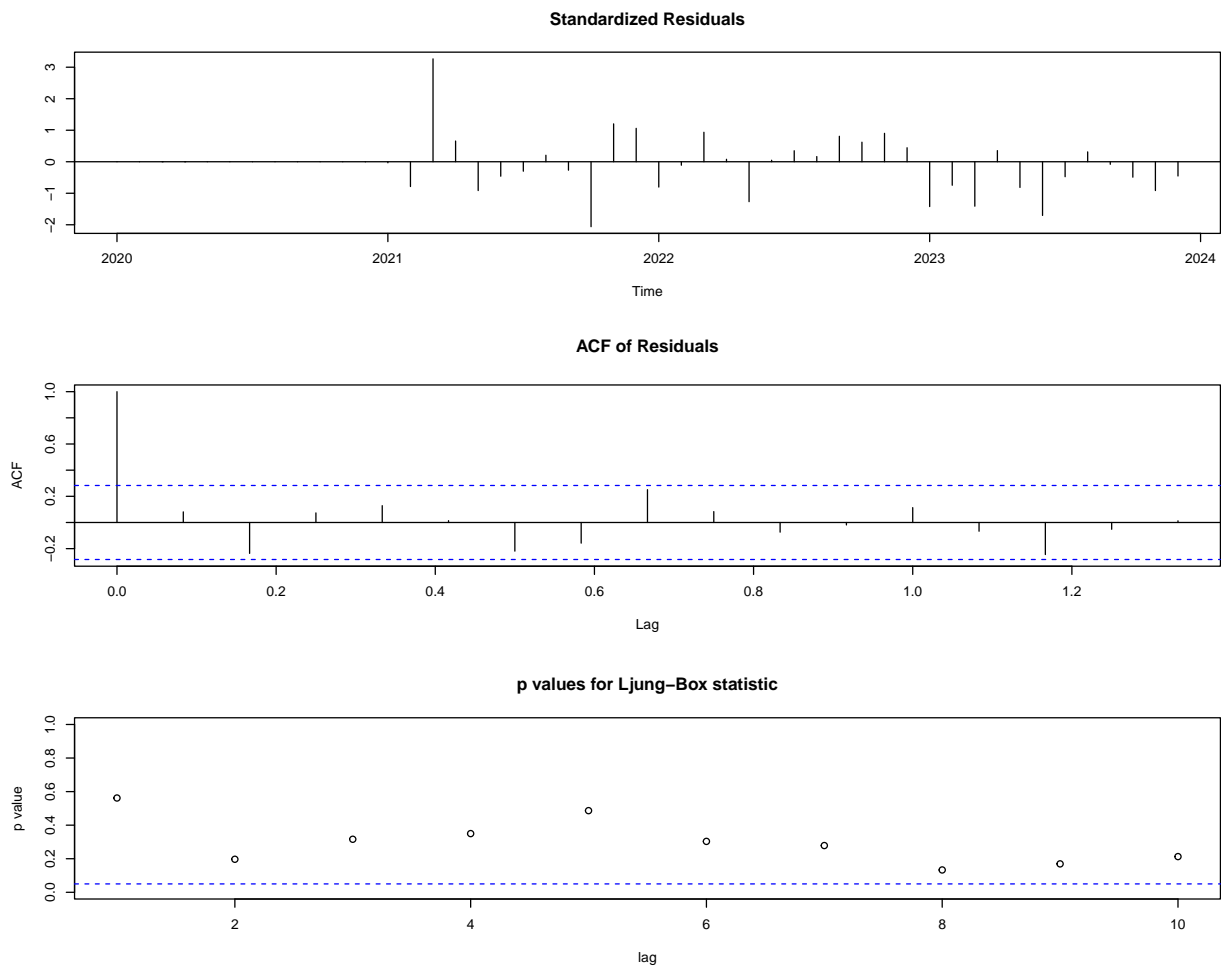
```
model <- arima(tsdata, c(0, 1, 1), seasonal = list(order = c(1, 1, 0)))
model
```

```
##
## Call:
## arima(x = tsdata, order = c(0, 1, 1), seasonal = list(order = c(1, 1, 0)))
##
## Coefficients:
##          ma1      sar1
##      -0.6580 -0.5317
## s.e.   0.1421  0.1754
##
## sigma^2 estimated as 39.57:  log likelihood = -116.31,  aic = 238.62
```

단계 5: 모형 진단(모형 타당성 검정)

단계 5-1: 자기 상관 함수에 의한 모형 진단

```
tsdiag(model)
```



단계 5-2: Box-Ljung에한 잔차항 모형 진단

```
Box.test(model$residuals, lag = 1, type = "Ljung")
```

```
##
## Box-Ljung test
##
## data: model$residuals
## X-squared = 0.33656, df = 1, p-value = 0.5618
```

단계 6: 미래 예측(업무 적용)

```
par(mfrow = c(1, 2))
fore <- forecast(model, h = 24)
plot(fore)
fore2 <- forecast(model, h = 6)
plot(fore2)
```

