시계열자료분석

HW02 (CH03, Ch04) 풀이

In [1]:
```
setwd("C:\\R-Project\\DAT\\Time Series Data")
options(repr.plot.width = 15, repr.plot.height = 8)
```

1. 중간재출하지수 자료를 이용하여, 초기 평활값 $S_0^{(1)} = 15$를 사용하여 $\omega = 0.6$와 $\omega = 0.2$의 각 경우에 1-시차 후 예측값 $\hat{Z}_{t-1}(1),\ t = 2, 3, 99, 100$을 계산하여라.

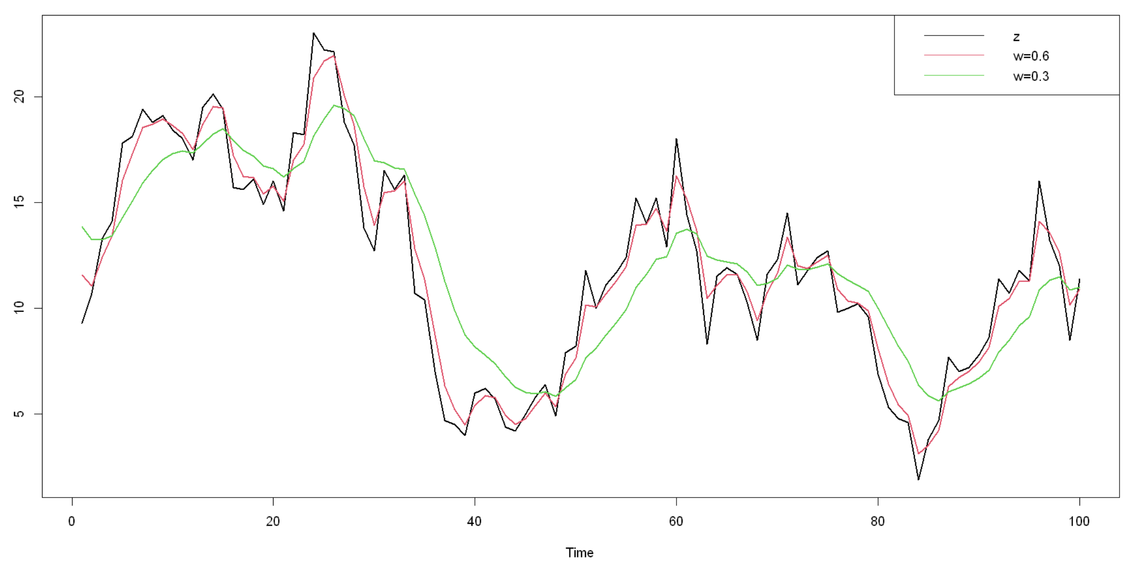
In [2]:
```
# 데이터 불러오기
z <- scan("mindex.txt")
```

$S_n^{(1)} = \omega \sum_{j=0}^{n-1} (1-\omega)^j Z_{n-j} + (1-\omega)^n S_0^{(1)}$

In [3]:
```
simple_exp_smoothing <- function(z,w,s0){
    Sn <- c()
    Sn[1] <- w*z[1] + (1-w)*s0
    for(k in 2:length(z)){
     Sn[k] <- w * z[k] + (1-w)*Sn[k-1]
    }
    return(Sn)
}
```

In [4]:
```
ses_0.6 <- simple_exp_smoothing(z,0.6,15)
ses_0.2 <- simple_exp_smoothing(z,0.2,15)
```

In [5]:
```
ts.plot(ts(z), ses_0.6, ses_0.2, col=1:3, lwd=2)
legend("topright", legend = c("z","w=0.6","w=0.3"), col=1:3, lty=1)
```



- ω 값이 클수록 원 데이터와 비슷하고, 평활 효과가 작아진다.

- ω 값이 작을수록 원데이터를 천천히 따라가고, 평활 효과가 커진다

$\hat{Z}_{t-1}(1) = S^{(1)}_{t-1}$ 이므로

In [6]:
```
point <- c(2,3,99,100)
ses_0.6[point-1]
ses_0.2[point-1]
```

11.58 · 11.052 · 12.6262842848221 · 10.1505137139288

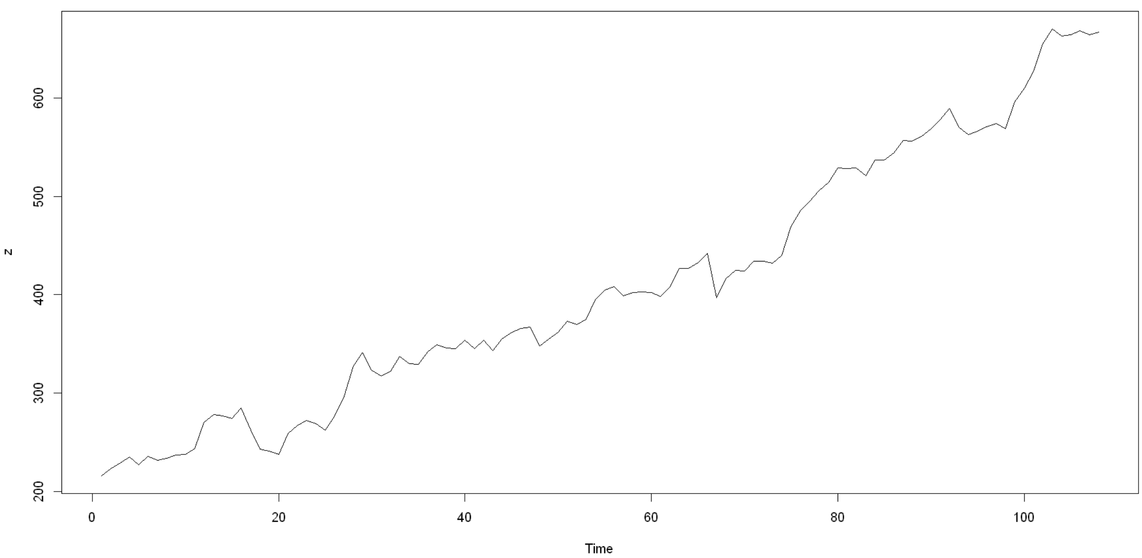
13.86 · 13.228 · 11.4677321254815 · 10.8741857003852

2. 다음의 각 자료에 대해 적절한 평활법을 적용한 후 , 예측오차 분석을 하여 적용한 평활법이 적절했는지 논하여라. 그리고 (2)번의 경우 첫번째 과제 5,6번에서 적합한 모형과 1-시차 후 예측 오차의 제곱합 (SSE) 기준하에서 예측력을 비교하여라.

(1) `female.txt' : 월별 전문기술행정직에 종사하는 여성근로자 수 (단위:명)

In [7]:
```
z <- scan("female.txt")
```

In [8]:
```
plot.ts(z)
```

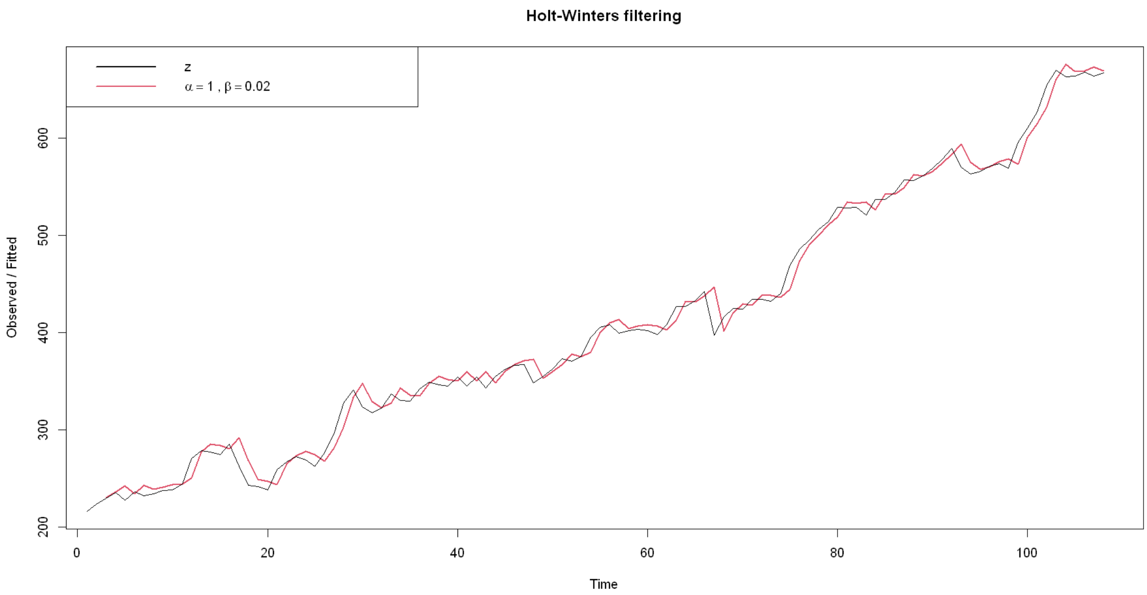


- 추세가 있기 때문에 이중지수평활법을 사용한다.

In [9]:
```
female_smoothing <- HoltWinters(z, gamma = FALSE)
female_smoothing
```

```
Holt-Winters exponential smoothing with trend and without seasonal component.

Call:
HoltWinters(x = z, gamma = FALSE)

Smoothing parameters:
 alpha: 1
 beta : 0.01666631
 gamma: FALSE

Coefficients:
        [,1]
a 667.000000
b   5.145043
```

In [10]:
```
plot(female_smoothing, lwd=2)
legend("topleft", legend=c("z",expression(alpha == 1~","~beta==0.02)), lty=1, lw
```



- 예측오차 분석

In [11]:
```
head(female_smoothing$fitted)
```

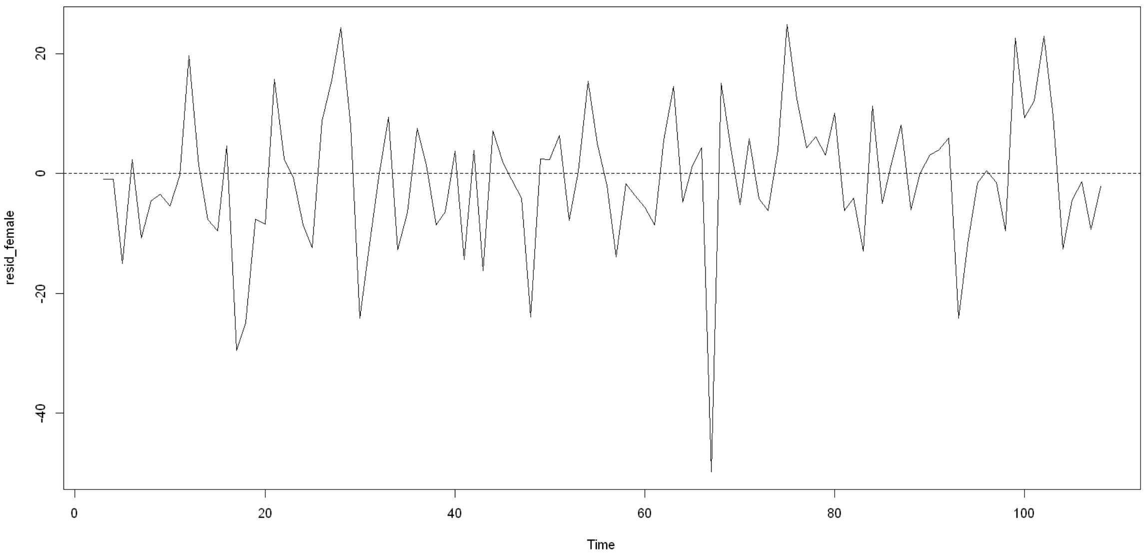
A matrix: 6 × 3 of type dbl

xhat	level	trend
230.0000	223	7.000000
235.9833	229	6.983334
241.9669	235	6.966945
233.7175	227	6.717501
242.7555	236	6.755542
238.5763	232	6.576287

- HoltWinters 함수를 사용하는 경우, 이중지수 평활값이 3번째부터 구해진다. 따라서 예측오차 역시 원데이터의 3번째 값부터 구할 수 있다.

In [12]:
```r
resid_female <- z[-(1:2)] - female_smoothing$fitted[,1]  #또는
resid_female <- resid(female_smoothing) #같은 방법임
```

In [13]:
```r
plot.ts(resid_female)
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [14]:
```r
t.test(resid_female)
```

```
	One Sample t-test

data:  resid_female
t = -0.93497, df = 105, p-value = 0.3519
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -3.276746  1.176749
sample estimates:
mean of x 
-1.049998 
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

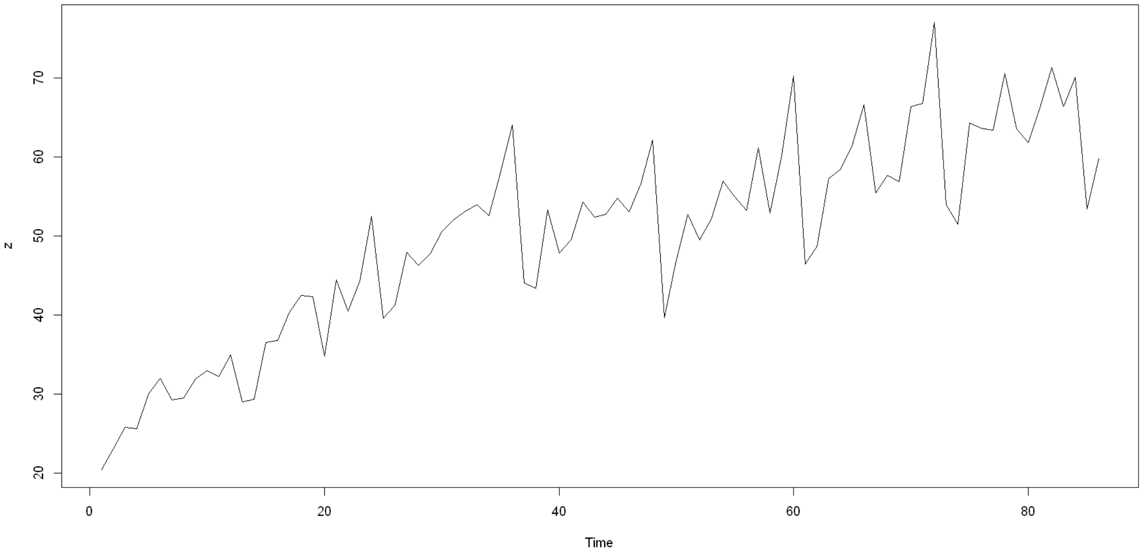
In [15]:
```r
lmtest::dwtest(lm(resid_female~1), alternative="two.sided")
```

```
	Durbin-Watson test

data:  lm(resid_female ~ 1)
DW = 1.7861, p-value = 0.2662
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 예측오차들 간의 자기 상관은 없는 것으로 보인다.
- 따라서 예측오차는 분석 결과 평활 결과는 적합하다고 할 수 있다.

(2) `export.txt' ; 월별수출액(단위:억$)

```
In [16]: z <- scan("export.txt")
plot.ts(z)
```



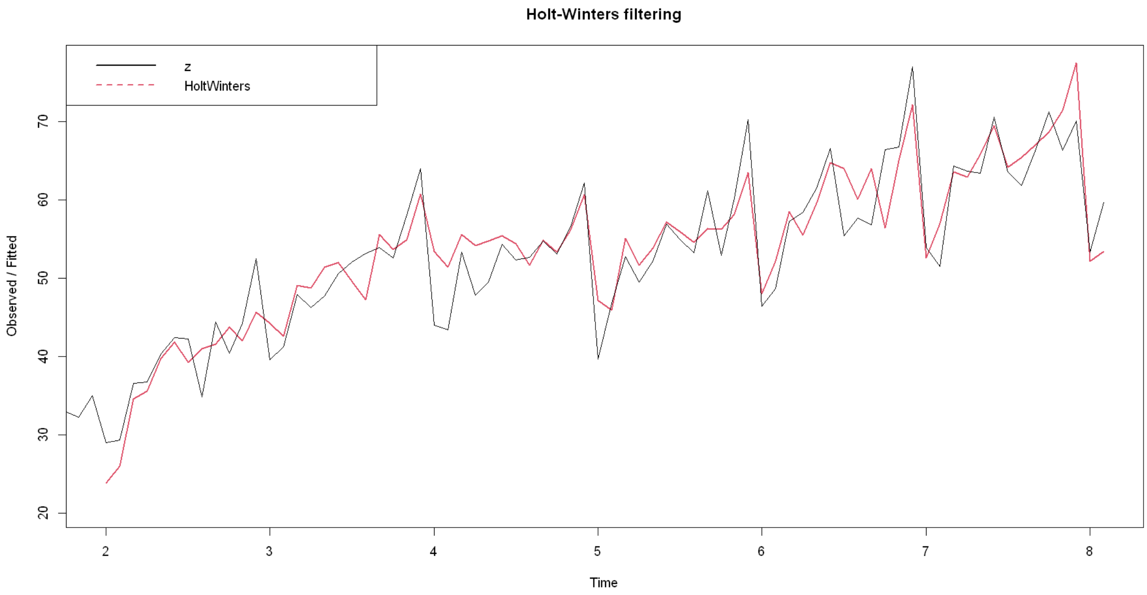
- 지난 과제에 언급한대로 이 시계열에는 이분산성은 없다고 할 수 있다.
- 추세와 계절성분이 있으므로 계절주기가 12, 가법형 계절평활법을 사용하는 것이 적합하다.
- 계절평활을 하기 위해서는 데이터를 ts 객체로 변환해야 한다.

```
In [17]: z_ts <- ts(z, frequency=12)
```

```
In [18]: export_smoothing <- HoltWinters(z_ts)
export_smoothing
```

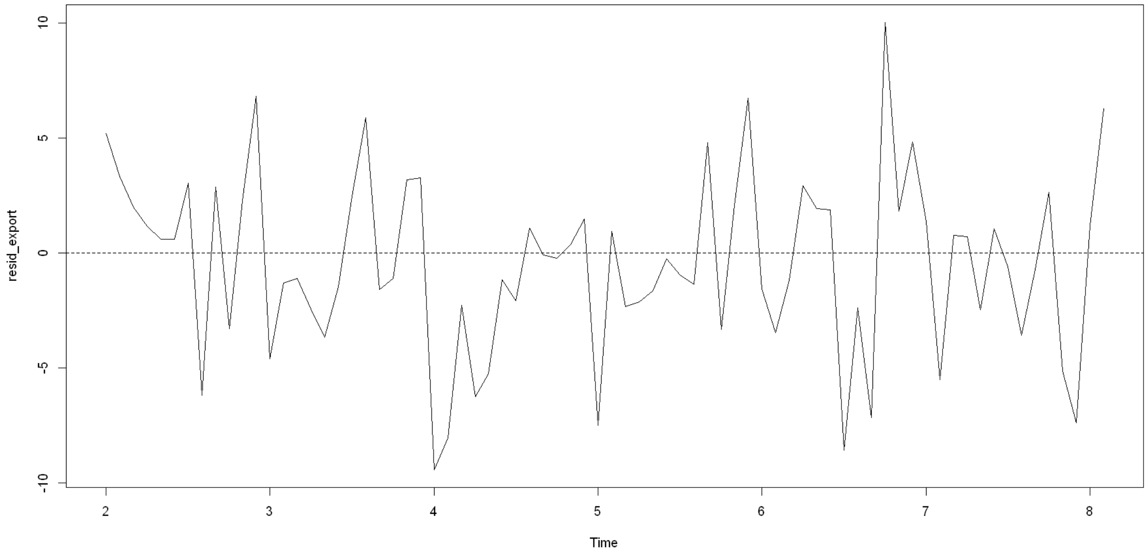
```
Holt-Winters exponential smoothing with trend and additive seasonal component.

Call:
HoltWinters(x = z_ts)

Smoothing parameters:
 alpha: 0.3304767
 beta : 0.04369053
 gamma: 0.6102758

Coefficients:
           [,1]
a    66.9300146
b     0.3670945
s1   -0.9590061
s2   -2.3460160
s3   -1.4388022
s4    4.0020957
s5   -2.8546787
s6   -3.1036803
s7    0.4486017
s8    3.3118493
s9    1.6302355
s10   8.0731659
s11 -11.5480012
s12  -8.8892298
```

In [19]:
```
plot(export_smoothing, lwd=2, lty=1:2)
legend("topleft", legend=c("z", "HoltWinters"), col=1:2, lty=1:2, lwd=2)
```



- 예측오차 분석

In [20]:
```
resid_export <- resid(export_smoothing)
plot.ts(resid_export)
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [21]:
```
t.test(resid_export)
```

```
        One Sample t-test

data:  resid_export
t = -0.99304, df = 73, p-value = 0.324
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -1.359825  0.455374
sample estimates:
 mean of x
-0.4522256
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [22]:
```
lmtest::dwtest(lm(resid_export~1), alternative="two.sided")
```

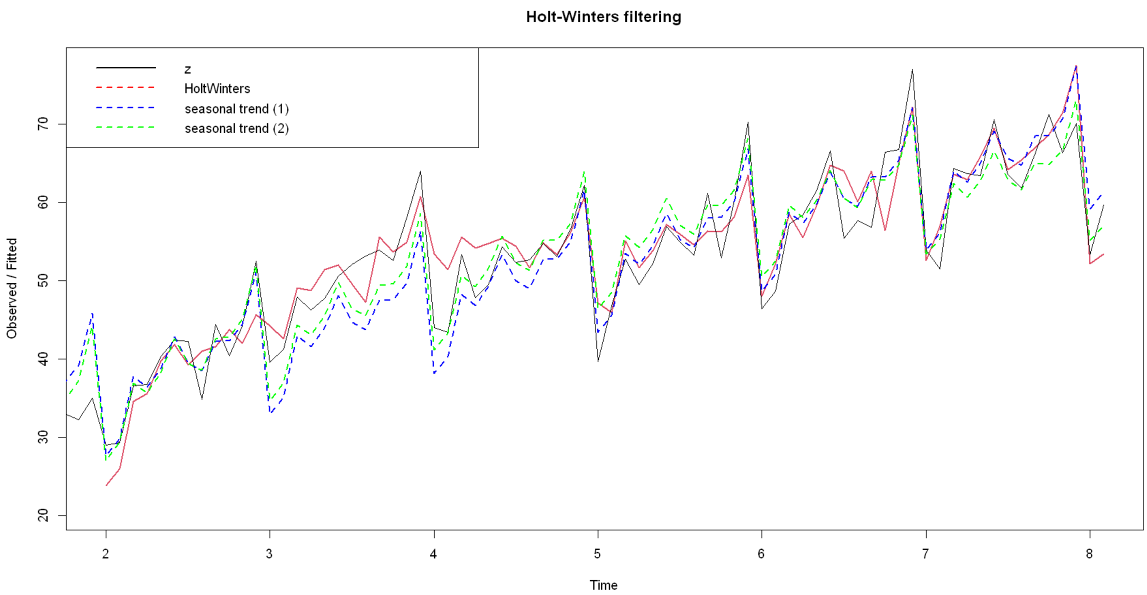
```
        Durbin-Watson test

data:  lm(resid_export ~ 1)
DW = 1.8163, p-value = 0.4267
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 예측오차들 간의 자기 상관은 없는 것으로 보인다.
- 따라서 예측오차는 분석 결과 평활 결과는 적합하다고 할 수 있다.
- 2장에서 계절추세 모형을 적합한 경우에는 예측오차 간에 자기 상관이 있다는 결과를 얻었었는데, 평활을 이용한 결과 자기상관이 제거되었다.

계절추세모형

In [23]:
```
seasonal_I <- as.factor(cycle(z_ts))
t <- 1:length(z)
m2 <- lm(z~t+seasonal_I)
m3 <- lm(z~t+I(t^2)+seasonal_I)
```

In [24]:
```
plot(export_smoothing, lwd=2, lty=1:2)
lines(ts(fitted(m2), frequency=12), col='blue', lty=2, lwd=2)
lines(ts(fitted(m3), frequency=12), col='green', lty=2, lwd=2)
legend("topleft", legend=c("z", "HoltWinters", "seasonal trend (1)", "seasonal t
```



- SSE 비교하기

In [25]:
```
head(export_smoothing$fitted)  ## 평활값은 12시차 이후값부터 생성됨
```

A matrix: 6 × 4 of type dbl

xhat	level	trend	season
23.77825	28.21682	0.8502812	-5.2888542
25.99728	30.78616	0.9253878	-5.7142708
34.53331	32.80963	0.9733636	0.7503125
35.55911	34.44285	1.0021933	0.1140625
39.71058	35.83200	1.0190994	2.8594792
41.81786	37.04258	1.0274655	3.7478125

In [26]:
```
sse_smoothing <- sum((z[-(1:12)]-export_smoothing$fitted[,1])^2) #또는
sse_smoothing <- export_smoothing$SSE
```

In [27]:
```
sse_m2 <- sum(resid(m2)^2)
sse_m3 <- sum(resid(m3)^2)
```

In [28]:
```
paste0("SSE_smomthing = ",sse_smoothing)
paste0("SSE_m2 = ", sse_m2)
paste0("SSE_m3 = ", sse_m3)
```

'SSE_smomthing = 1135.42182593918'

'SSE_m2 = 1371.05731181319'

'SSE_m3 = 934.603221419272'

- 모형을 1시차 예측오차에 대한 제곱합으로 비교하면 2차추세모형을 적합한 m3의 SSE값이 가장 작고, 1차선형추세를 사용한 모형의 SSE가 가장 크다.

In [29]:
```
mse_smoothing <- mean((z[-(1:12)]-export_smoothing$fitted[,1])^2)
mse_m2 <- mean(resid(m2)^2)
mse_m3 <- mean(resid(m3)^2)

paste0("MSE_smomthing = ",mse_smoothing)
paste0("MSE_m2 = ", mse_m2)
paste0("MSE_m3 = ", mse_m3)
```

'MSE_smomthing = 15.3435381883673'

'MSE_m2 = 15.9425268815487'

'MSE_m3 = 10.8674793188287'

- MSE로 비교하여도 결과는 같다.

3. 'data1.csv'는 모의 실험에 의해 생성된 시계열자료(t:시간, z:시계열자료)이다.다음 물음에 답하여라.

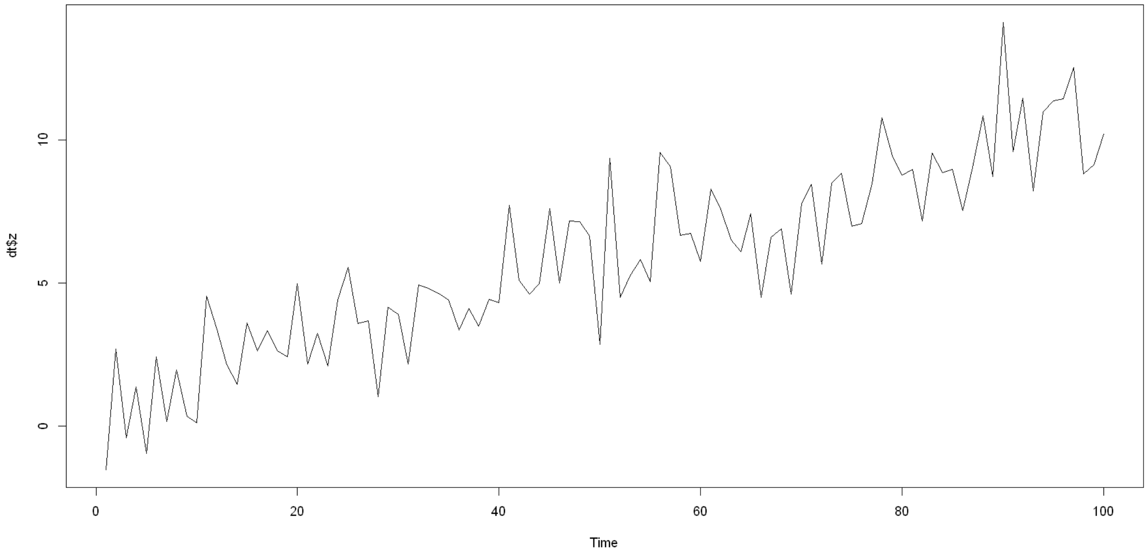
(1) 적절한 추세 모형을 적합시킨 후 잔차분석을 하여라.

In [30]:
```
dt <- read.csv('data1.csv')
head(dt)
```

A data.frame: 6 × 3

	X	t	z
	<int>	<int>	<dbl>
1	1	1	-1.5346871
2	2	2	2.6850469
3	3	3	-0.4288189
4	4	4	1.3724199
5	5	5	-0.9800884
6	6	6	2.4156505

In [31]:
```
plot.ts(dt$z)
```



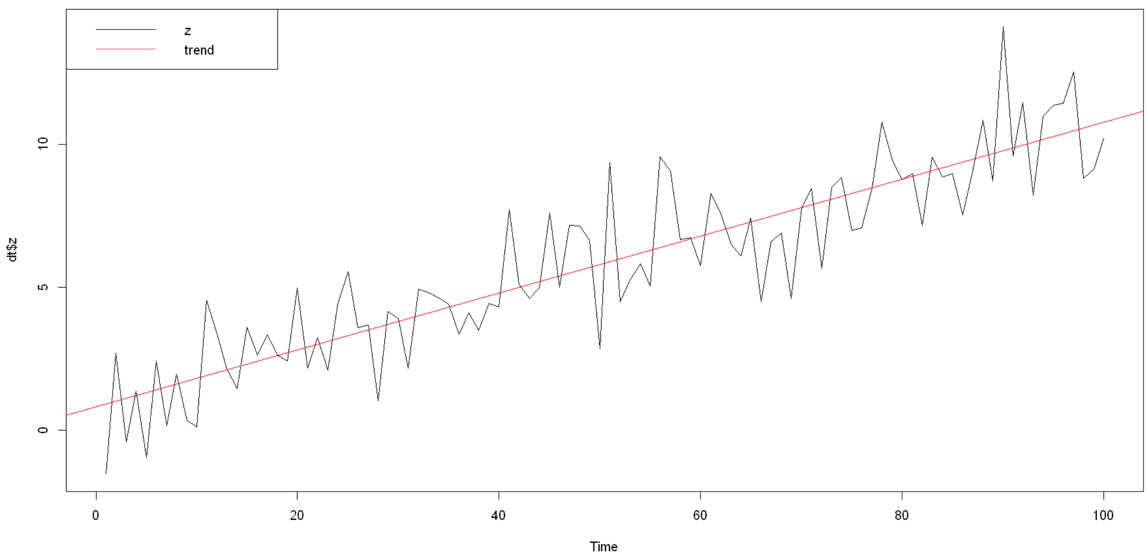
- 추세가 있음, 계절성분은 없음
- 선형추세 모형 적합

In [32]:
```
m_trend <- lm(z~t, dt)
summary(m_trend)
```

```
Call:
lm(formula = z ~ t, data = dt)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.0803 -1.0287  0.0169  0.8426  4.3288 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 0.818932   0.296301   2.764  0.00682 ** 
t           0.099619   0.005094  19.557  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.47 on 98 degrees of freedom
Multiple R-squared:  0.796,	Adjusted R-squared:  0.7939 
F-statistic: 382.5 on 1 and 98 DF,  p-value: < 2.2e-16
```

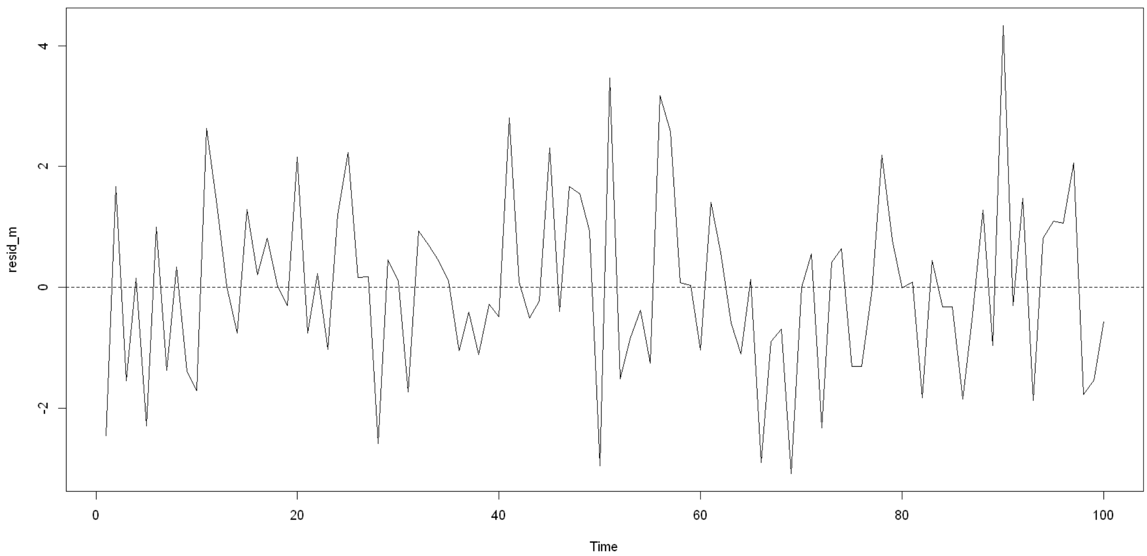
- 모형 적합이 잘 되었음.
- $\hat{z} = 0.82 + 0.10t$

In [33]:
```
plot.ts(dt$z)
abline(m_trend, col='red')
legend("topleft", c("z", "trend"), lty=1, col=1:2)
```



잔차분석

In [34]:
```
resid_m <- resid(m_trend)
plot.ts(resid_m)
abline(h=0, lty=2)
```



- 잔차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [35]:
```
t.test(resid_m)
```

```
	One Sample t-test

data:  resid_m
t = -3.7755e-16, df = 99, p-value = 1
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.2902834  0.2902834
sample estimates:
    mean of x
-5.523414e-17
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [36]:
```
lmtest::dwtest(m_trend)
```

```
	Durbin-Watson test

data:  m_trend
DW = 2.2164, p-value = 0.8389
alternative hypothesis: true autocorrelation is greater than 0
```

- DW test 결과 잔차들 간의 자기 상관은 없는 것으로 보인다.
- 따라서 잔차 분석 결과 추세모형 결과는 적절하다고 할 수 있다.

(2) (1)에서 적합한 추세모형에서 1~ 10 시차 후의 예측값을 구하여라.

In [37]:
```
new_dt <- data.frame(t=100+(1:10))
pred_trend <- predict(m_trend, new_dt)
pred_trend
```

1: 10.8804416292757 **2:** 10.9800605357706 **3:** 11.0796794422655 **4:** 11.1792983487603 **5:** 11.2789172552552 **6:** 11.3785361617501 **7:** 11.4781550682449 **8:** 11.5777739747398 **9:** 11.6773928812347 **10:** 11.7770117877295

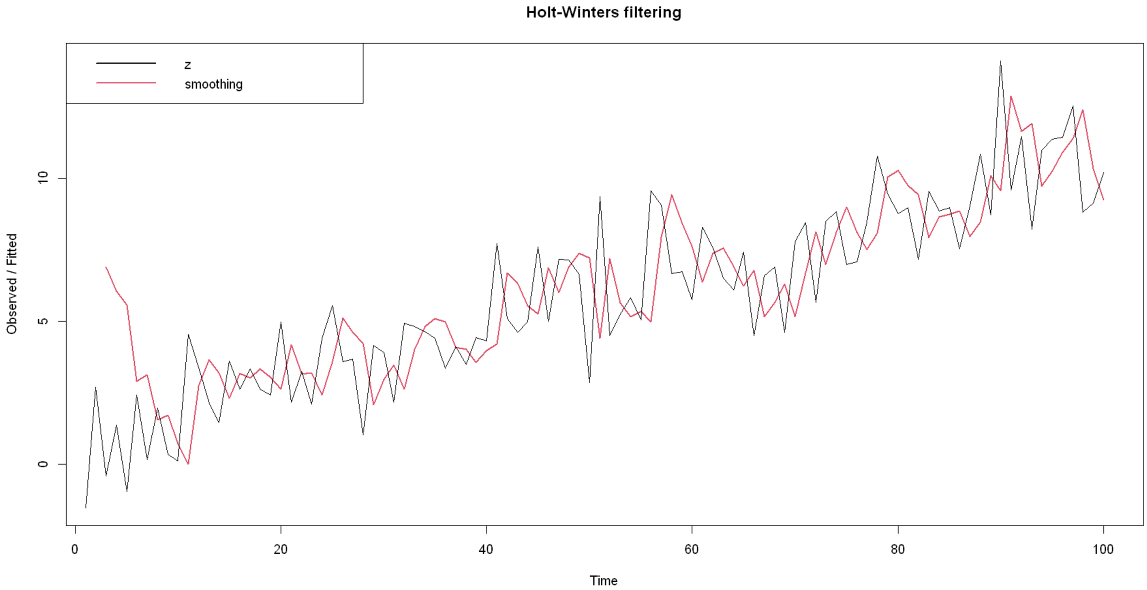
(3) 적절한 평활법을 적용한 후 잔차분석을 하여라.

- 추세만 있으므로 이중지수평활 적용

In [38]:
```
m_smoothing <- HoltWinters(dt$z, gamma=FALSE)
m_smoothing
```

```
Holt-Winters exponential smoothing with trend and without seasonal component.

Call:
HoltWinters(x = dt$z, gamma = FALSE)

Smoothing parameters:
 alpha: 0.496601
 beta : 0.396225
 gamma: FALSE

Coefficients:
        [,1]
a  9.7266353
b -0.3150449
```

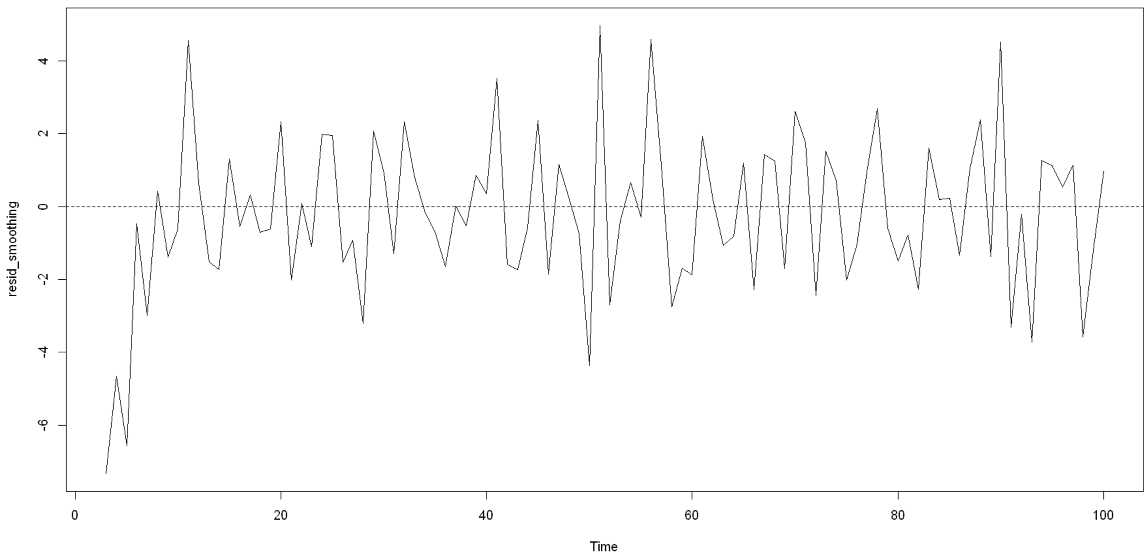
In [39]:
```
plot(m_smoothing, lwd=2)
legend("topleft", legend=c("z","smoothing"), lty=1, lwd=2, col=1:2)
```



예측오차 분석

In [40]:
```
resid_smoothing <- dt$z[-(1:2)] - m_smoothing$fitted[,1]  #또는
resid_smoothing <- resid(m_smoothing) #같은 방법임
```

In [41]:
```
plot.ts(resid_smoothing)
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [42]:
```
t.test(resid_smoothing)
```

```
	One Sample t-test

data:  resid_smoothing
t = -1.0711, df = 97, p-value = 0.2868
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.6709125  0.2005740
sample estimates:
 mean of x 
-0.2351693 
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [43]:
```
lmtest::dwtest(lm(resid_smoothing~1), alternative="two.sided")
```

```
	Durbin-Watson test

data:  lm(resid_smoothing ~ 1)
DW = 1.9664, p-value = 0.8676
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 예측오차들 간의 자기 상관은 없는 것으로 보인다.
- 따라서 예측오차는 분석 결과 평활 결과는 적절하다고 할 수 있다.

(4) (3)의 결과를 이용하여 1∼10시차 후의 예측값을 구하여라.

In [44]:
```
pred_smoothing <- predict(m_smoothing, n.ahead=10, prediction.interval = T, leve
pred_smoothing
```

A Time Series:

9.41159036810392 · 9.09654542365521 · 8.78150047920649 · 8.46645553475778 · 8.15141059030907 · 7.83636564586035 · 7.52132070141164 · 7.20627575696293 · 6.89123081251421 · 6.5761858680655

In [45]:
```
forecast::holt(dt$z, alpha = m_smoothing$alpha, beta=m_smoothing$beta, h=10, ini
```

```
Registered S3 method overwritten by 'quantmod':
  method            from
  as.zoo.data.frame zoo 
```

```
    Point Forecast       Lo 80    Hi 80      Lo 95    Hi 95
101       9.411590   6.7026406 12.12054   5.268609 13.55457
102       9.096545   5.4649981 12.72809   3.542573 14.65052
103       8.781500   3.7434431 13.81956   1.076457 16.48654
104       8.466456   1.6676142 15.26530  -1.931475 18.86439
105       8.151411  -0.6814246 16.98425  -5.357245 21.66007
106       7.836366  -3.2573443 18.93008  -9.130000 24.80273
107       7.521321  -6.0316379 21.07428 -13.206141 28.24878
108       7.206276  -8.9850805 23.39763 -17.556267 31.96882
109       6.891231 -12.1036780 25.88614 -22.158975 35.94144
110       6.576186 -15.3766386 28.52901 -26.997762 40.15013
```

(5) 실제 $1, 2, \ldots, 10$ 시차 후의 관측값이 ‘data1_new.csv’일 때, (2), (4)의 결과를 이용하여 어느 모형이 더 적합했는지에 대해 비교하여라.

In [46]:
```
real_new_dt <- read.csv("data1_new.csv")
real_new_dt
```

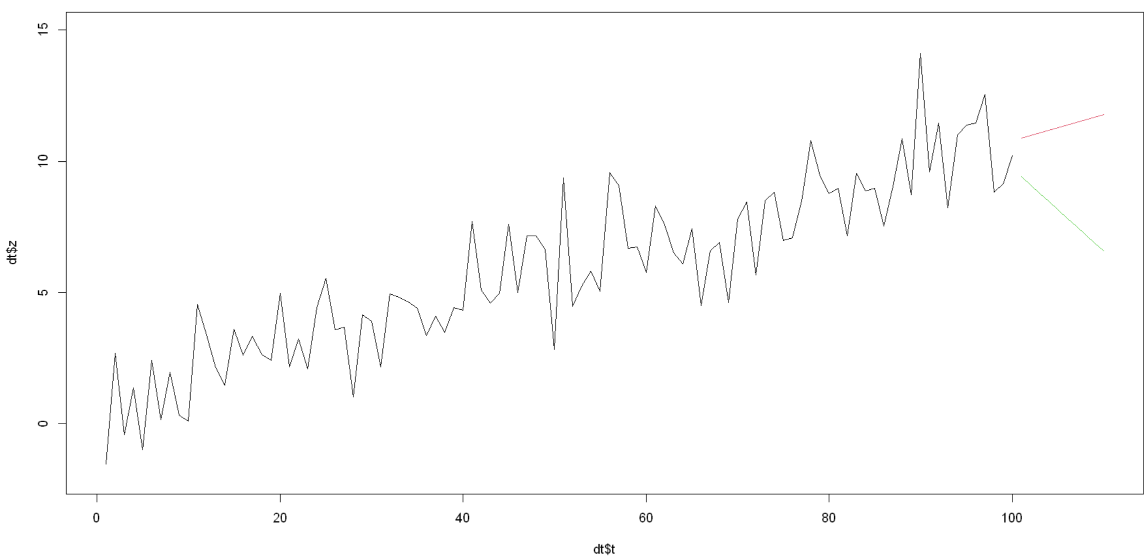
A data.frame: 10 × 3

X	t	z
<int>	<int>	<dbl>
1	101	11.577471
2	102	13.171004
3	103	12.730258
4	104	12.242375
5	105	10.877077
6	106	13.297355
7	107	14.632224
8	108	15.049088
9	109	9.671734
10	110	12.216420

In [47]:
```
min(dt$z)
```

-1.53468707477591

In [48]:
```
plot(dt$t,dt$z, xlim=c(1,110), ylim=c(-2, 15), type='l')
lines(101:110, pred_trend, col=2)
lines(101:110, pred_smoothing, col=3)
```



- MSE, MAPE, MAE 값을 이용하여 비교할 수 있다.
- $MSE = \frac{1}{m}\sum_{i=1}^{m}\hat{e}_{n-1+t}(1)^2$, $MAPE = \frac{100}{m}\sum_{i=1}^{m}\left|\frac{\hat{e}_{n-1+t}(1)}{Z_{n+t}}\right|$, $MAE = \frac{1}{m}\sum_{i=1}^{m}|\hat{e}_{n-1+t}(1)|$

In [49]:
```r
e_trend <- real_new_dt$z - pred_trend
e_smoothing <- real_new_dt$z - pred_smoothing
```

In [50]:
```r
(MSE_trend = mean(e_trend^2))
(MSE_smoothing = mean(e_smoothing^2))

(MAPE_trend = 100 * sum(abs(e_trend/ real_new_dt$z)) / 10)
(MAPE_smoothing = 100 * sum(abs(e_smoothing/ real_new_dt$z)) / 10)

(MAE_trend = mean(abs(e_trend)))
(MAE_smoothing = mean(abs(e_smoothing)))
```

3.91978903404577

24.0012264541186

13.1385437579375

35.3262009444052

1.69927387507187

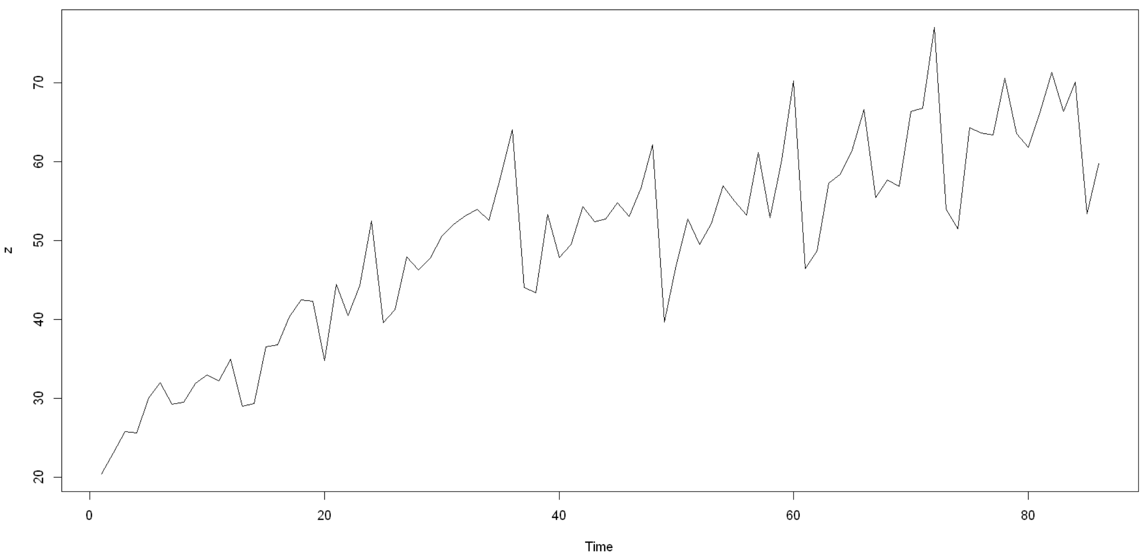
4.55261253925276

Model	MSE	MAPE	MAE
trend	3.9198	13.1385	1.6993
smoothing	24.0012	35.3262	4.5526

- 어떤 값을 보나 1차선형 추세 모형이 예측을 더 잘했다고 할 수 있다.
- 추세가 전체적으로 변하지 않는 경우에는 평활법보다는 선형추세 모형이 더 적합하다고 할 수 있다.

4. `export.txt'자료에 대하여 각가 다음 물음에 답하여라.

In [51]:
```r
z <- scan('export.txt')
plot.ts(z)
```



(1) 추세분석을 이용한 분해법에 의한 각 성분의 시계열 그림을 그려라.

- 위에서 2차선형추세 모형이 더 적절했다고 했으므로, 2차 추세 모형을 적합한다.

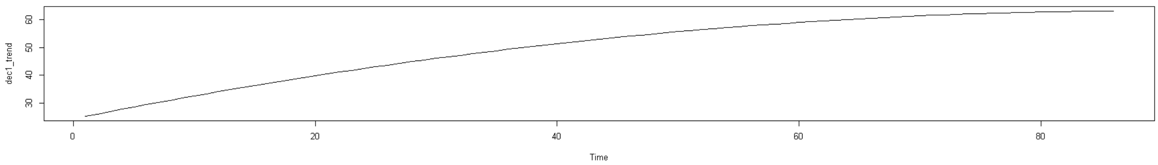
In [52]:
```
z_ts <- ts(z, frequency=12)
seasonal_I <- as.factor(cycle(z_ts))
t <- 1:length(z)
```

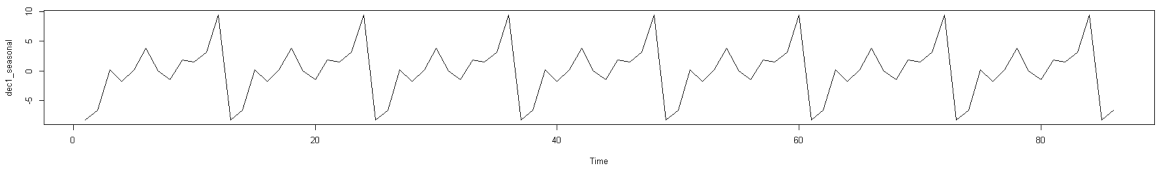
In [53]:
```
dec1_trend <- fitted(lm(z~t+I(t^2)))  ## 추세성분
```

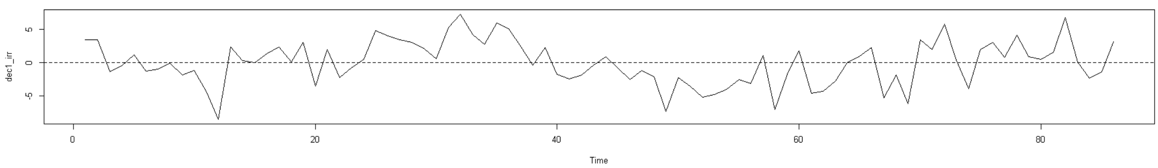
In [54]:
```
adjtrend = z-dec1_trend
dec1_seasonal <- fitted(lm(adjtrend ~ 0+seasonal_I)) ## 계절성분
```

In [55]:
```
dec1_irr = z-dec1_trend-dec1_seasonal ## 불규칙 성분
```

In [56]:
```
par(mfrow=c(3,1))
plot.ts(dec1_trend)
plot.ts(dec1_seasonal)
plot.ts(dec1_irr)
abline(h=0, lty=2)
par9mfrow=c(1,1)
```

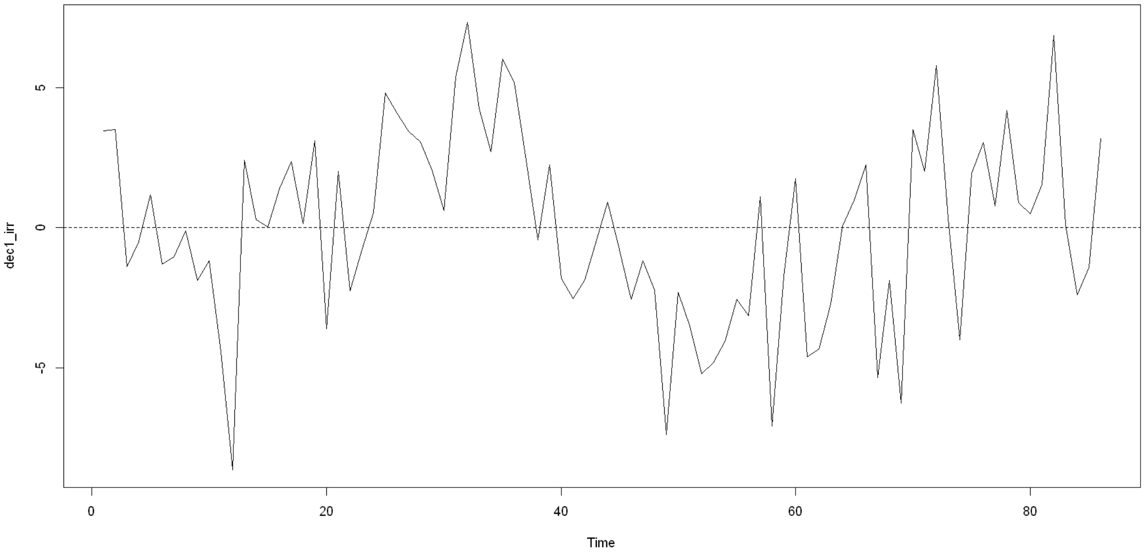






(2) 추정된 불규칙성분의 분석을 통해 적용된 분해법이 적절했는지 논하여라.

In [57]:
```
plot.ts(dec1_irr)
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 보이지 않는다.

In [58]:
```
t.test(dec1_irr)
```

```
	One Sample t-test

data:  dec1_irr
t = 1.7925e-16, df = 85, p-value = 1
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.7179805  0.7179805
sample estimates:
   mean of x
6.472939e-17
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [59]:
```
lmtest::dwtest(lm(dec1_irr~1), alternative="two.sided")
```

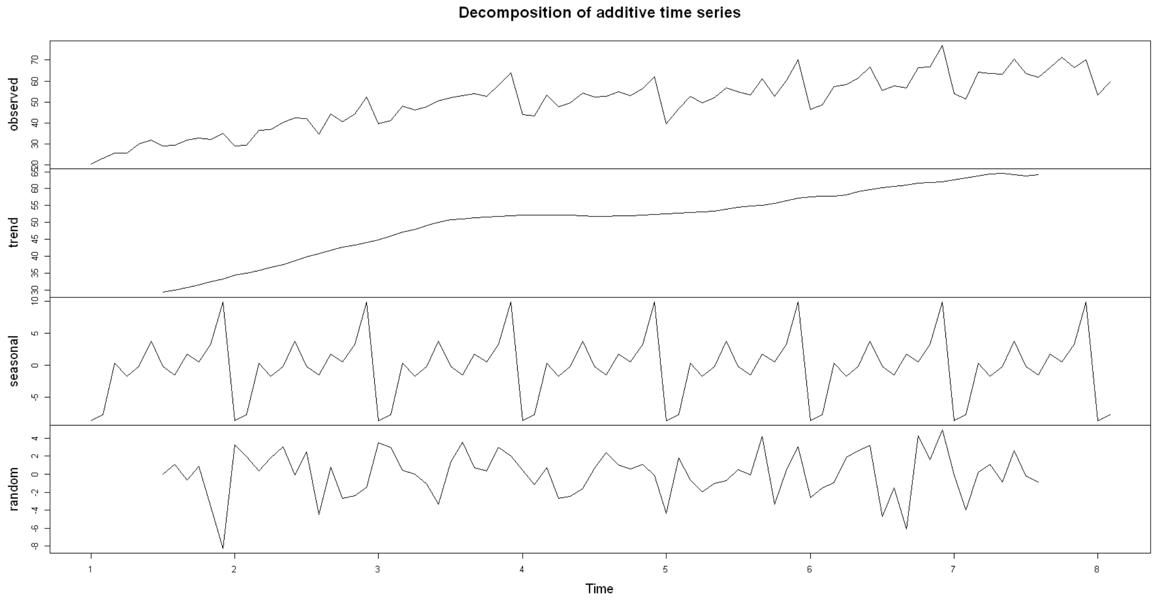
```
	Durbin-Watson test

data:  lm(dec1_irr ~ 1)
DW = 1.1429, p-value = 2.78e-05
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 예측오차들 간의 자기 상관은 있는 것으로 보인다.
- 오차들간의 상관관계가 있는 것을 제외하고는 적절하다고 볼 수 있다.

(3) 이동평균을 이용한 분해법에 의한 각 성분의 시계열 그림을 그려라.

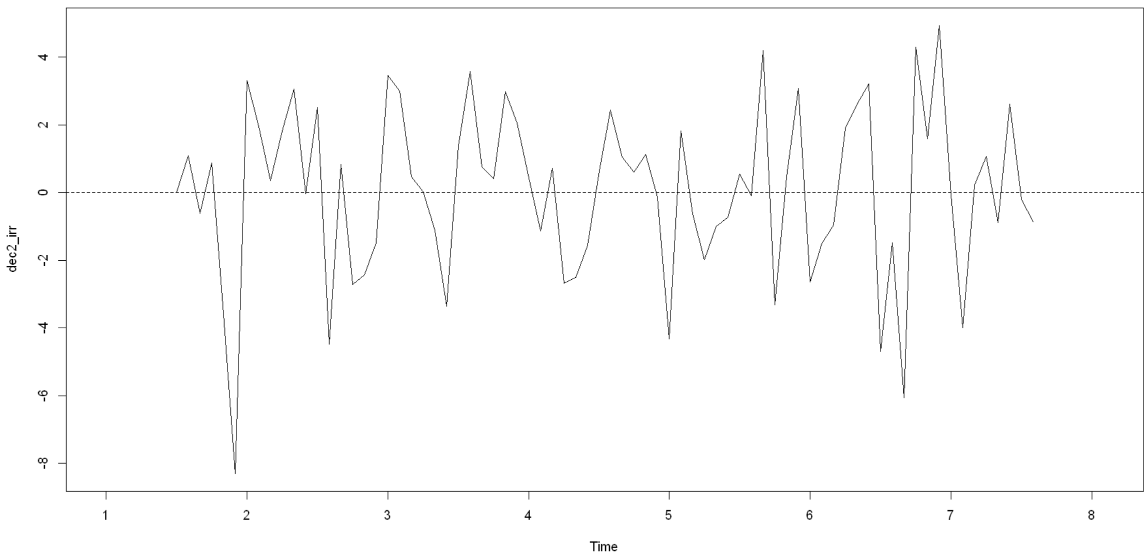
- 이분산성을 보이지 않기 때문에 가법 모형을 사용
- decompose 함수 사용

In [60]:
```
dec_fit <- decompose(z_ts, 'additive')
plot(dec_fit)
```



(4) 추정된 불규칙성분의 분석을 통해 적용된 분해법이 적절했는지 논하여라.

In [61]:
```
dec2_irr <- dec_fit$random
plot.ts(dec2_irr)
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [62]:
```
t.test(dec2_irr)
```

```
	One Sample t-test

data:  dec2_irr
t = 0.074013, df = 73, p-value = 0.9412
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.5674456  0.6112171
sample estimates:
 mean of x
0.02188575
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [63]:
```
lmtest::dwtest(lm(dec2_irr~1), alternative="two.sided")
```

```
	Durbin-Watson test

data:  lm(dec2_irr ~ 1)
DW = 1.9469, p-value = 0.8188
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 예측오차들 간의 자기 상관은 없는 것으로 보인다.
- 따라서 예측오차는 분석 결과 평활 결과는 적절하다고 할 수 있다.
- 추세를 이용한 분해법에서 예측 오차는 상관관계가 있었지만, 평활법을 사용하면서 예측오차들의 상관관계를 제거할 수 있었다.

(5) 각 분해법에 의한 결과를 1-시작 후 예측오차의 제곱합 (SSE) 기준하에서 2번의 결과와 예측력을 비교하여라.

In [64]:
```
sse_1 = sum(dec1_irr^2)
sse_2 = sum(dec2_irr^2, na.rm=T)

paste0("SSE_1 = ", sse_1)
paste0("SSE_2 = ", sse_2)
```

'SSE_1 = 953.219486362055'

'SSE_2 = 472.381499188028'

- SSE 값을 비교하였을 때, 평활에 의한 분해법의 값이 더 작다.

In [65]:
```
mse_1 = mean(dec1_irr^2)
mse_2 = mean(dec2_irr^2, na.rm=T)

paste0("MSE_1 = ", mse_1)
paste0("MSE_2 = ", mse_2)
```

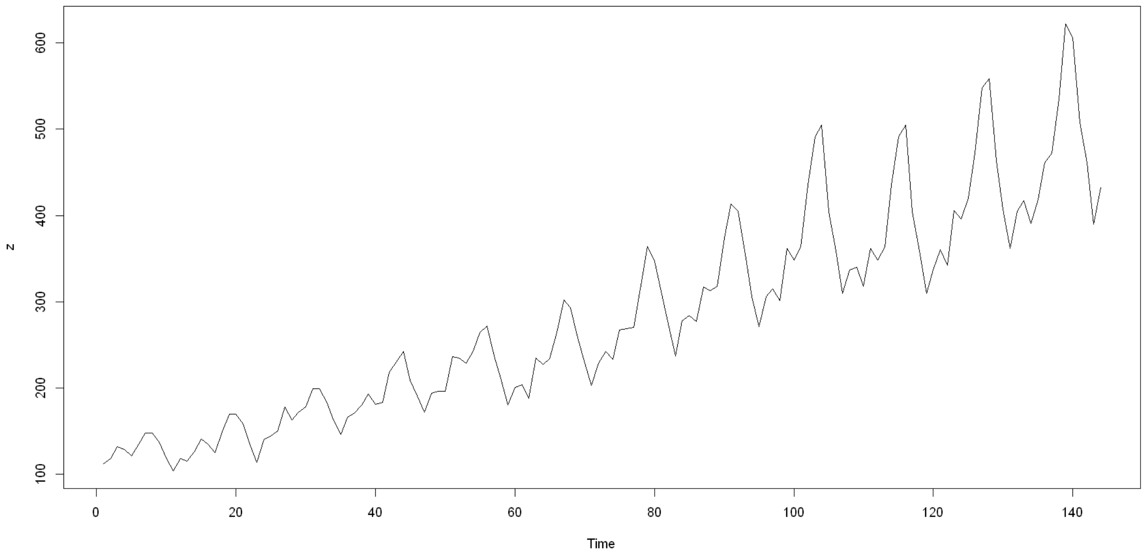
'MSE_1 = 11.0839475158378'

'MSE_2 = 6.38353377281119'

- MSE 값을 비교하여도 결과가 같다.
- 따라서 불규칙성분을 분석한 결과와 더불어 평활에 의한 분해법이 더 적절하다고 할 수 있다.

5. ‘usapass.txt’는 미국 월별 비행기 승객 수(단위 : 천 명)의 시계열자료이다. log 변환 후 아래의 분석을 수행하시오. ((2),(3),(4)번에 대하여 모형을 적합한 후 실제데이터와 각 모형에서 구해진 추정값을 비교하는 그림도 포함)

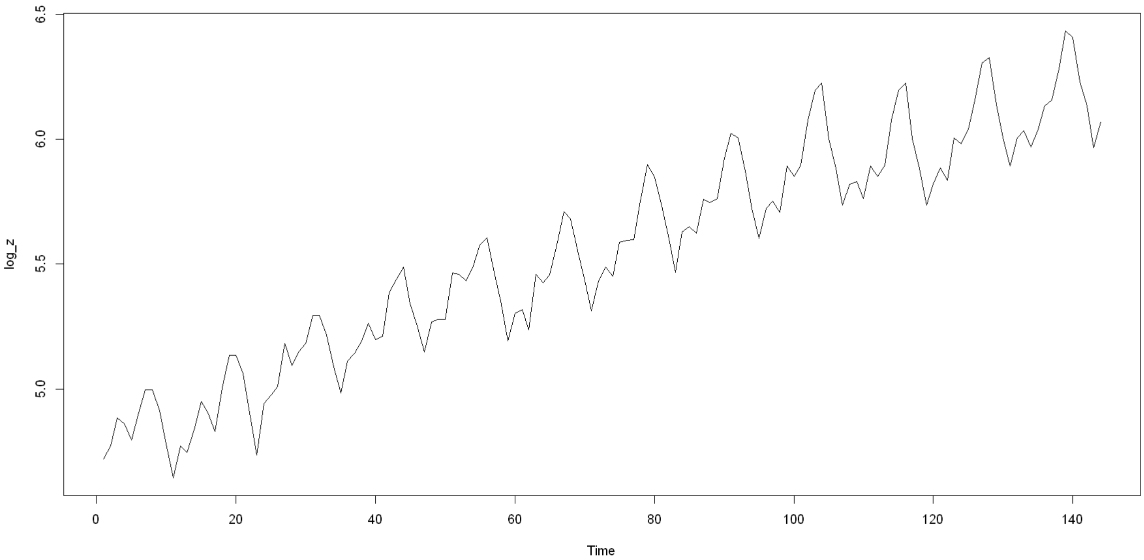
In [66]:
```
z <- scan('usapass.txt')
plot.ts(z)
```



(1) 왜 log 변환이 필요한지에 대해 간단히 설명하여라.

- 계절성분의 진폭이 시간이 흐름에 따라 증가하고 있는 것을 볼 수 있다.
- 따라서 이분산성을 제거하기 위해 로그변환이 필요하다.

In [67]:
```
log_z <- log(z)
plot.ts(log_z)
```



- 로그변환을 통해 이분산성이 제거된 것을 확인할 수 있다.
- 이 자료에는 추세성분과 계절성분(주기 12)이 있다.

(2) 적절한 추세 모형을 적합시킨 후 잔차분석을 하여라.

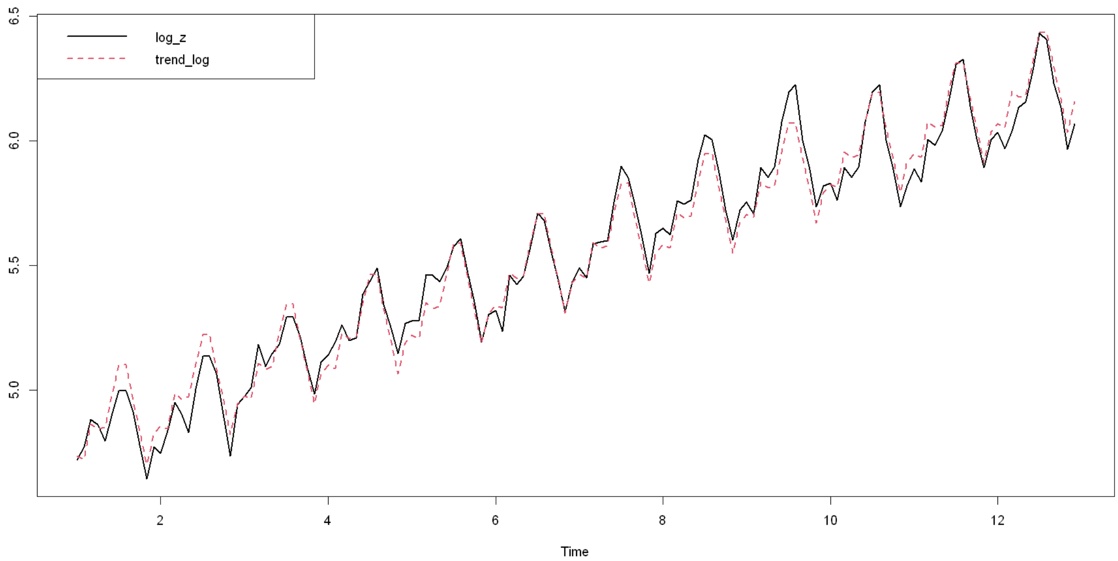
- 지수함수를 사용한 계절형 추세모형을 적합한다.

In [68]:
```r
z_ts <- ts(z, frequency=12)
log_z_ts = log(z_ts)
seasonal_I <- as.factor(cycle(log_z_ts))
t <- 1:length(z)
m_trend <- lm(log_z_ts~0+t+seasonal_I)

summary(m_trend)
```

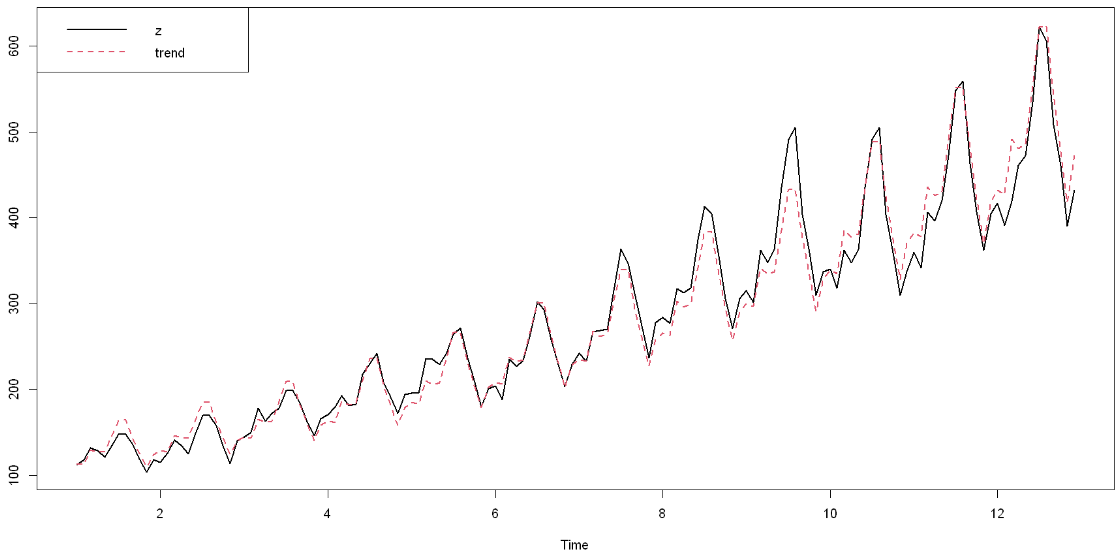
```
Call:
lm(formula = log_z_ts ~ 0 + t + seasonal_I)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.159814 -0.044426  0.000623  0.045572  0.151846 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
t            0.0100999  0.0001252   80.67   <2e-16 ***
seasonal_I1  4.7246982  0.0198289  238.27   <2e-16 ***
seasonal_I2  4.7026123  0.0198822  236.52   <2e-16 ***
seasonal_I3  4.8342011  0.0199361  242.49   <2e-16 ***
seasonal_I4  4.8015084  0.0199907  240.19   <2e-16 ***
seasonal_I5  4.8009618  0.0200459  239.50   <2e-16 ***
seasonal_I6  4.9237482  0.0201017  244.94   <2e-16 ***
seasonal_I7  5.0296649  0.0201582  249.51   <2e-16 ***
seasonal_I8  5.0223242  0.0202153  248.44   <2e-16 ***
seasonal_I9  4.8708171  0.0202729  240.26   <2e-16 ***
seasonal_I10 4.7357833  0.0203312  232.93   <2e-16 ***
seasonal_I11 4.5905564  0.0203901  225.14   <2e-16 ***
seasonal_I12 4.7032829  0.0204496  229.99   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.06224 on 131 degrees of freedom
Multiple R-squared:  0.9999,	Adjusted R-squared:  0.9999 
F-statistic: 8.844e+04 on 13 and 131 DF,  p-value: < 2.2e-16
```

- $\hat{Z}_t = 0.01t + 4.72I_1 + 4.70I_2 + \cdots + 4.70I_{12}$

In [69]:
```r
ts.plot(log_z_ts, ts(fitted(m_trend), frequency=12), col=1:2, lty=1:2, lwd=2)
legend("topleft", c("log_z", "trend_log"), col=1:2, lty=1:2, lwd=2)
```

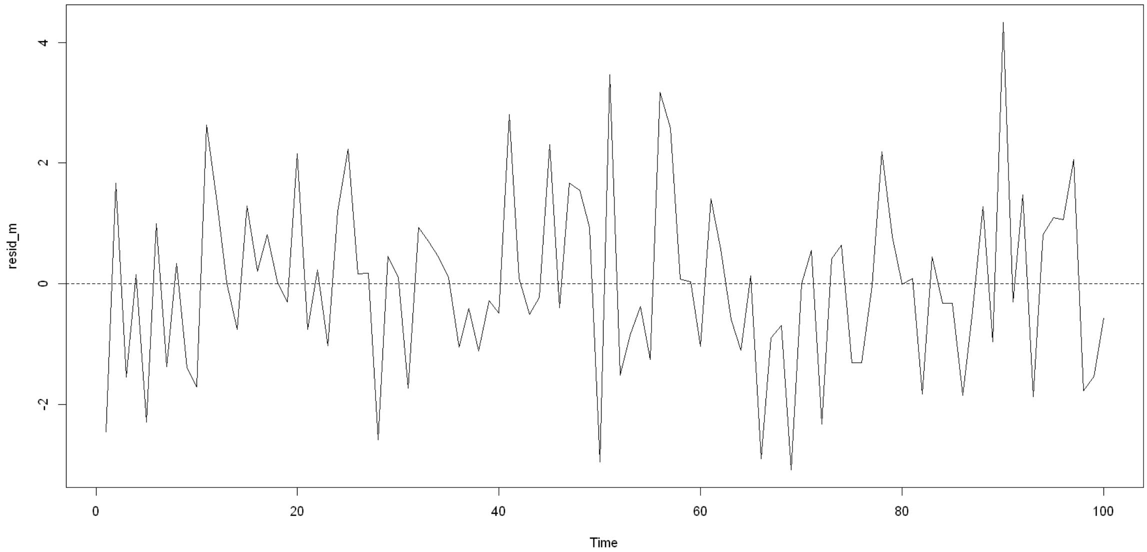


In [70]:
```
ts.plot(z_ts, ts(exp(fitted(m_trend)), frequency=12), col=1:2, lty=1:2, lwd=2)
legend("topleft", c("z", "trend"), col=1:2, lty=1:2, lwd=2)
```



잔차분석

In [71]:
```
resid_trend <- resid(m_trend)
plot.ts(resid_m)
abline(h=0, lty=2)
```



- 잔차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [72]:
```
t.test(resid_m)
```

```
	One Sample t-test

data:  resid_m
t = -3.7755e-16, df = 99, p-value = 1
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.2902834  0.2902834
sample estimates:
    mean of x
-5.523414e-17
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [73]:
```
lmtest::dwtest(m_trend, alternative="two.sided")
```

```
	Durbin-Watson test

data:  m_trend
DW = 0.40831, p-value < 2.2e-16
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 잔차들 간의 자기 상관이 있는 것으로 보인다.
- 따라서 잔차 분석 결과 추세모형 결과는 오차간의 자기 상관이 있는 것을 제외하면 적절해 보인다. 하지만 오차간의 자기상관을 제거할 필요가 있다.

(3) 적절한 평활법을 적용한 후 잔차분석을 하여라.

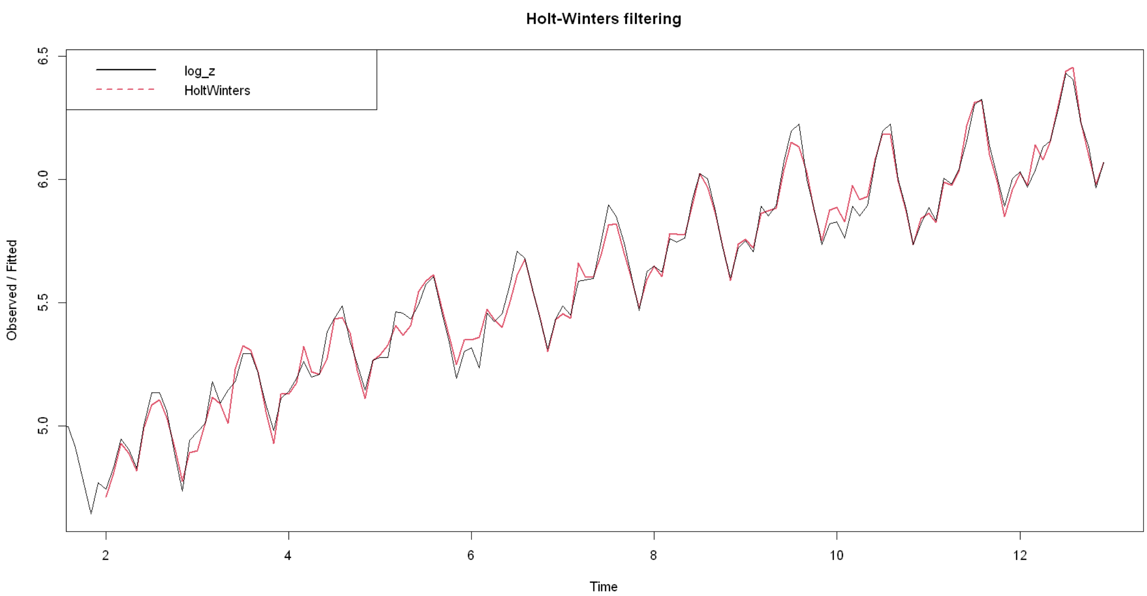
- 추세와 계절성분이 있으므로 계절평활평활을 사용한다.
- 로그변환한 데이터에 대해 가법 모형을 적용할 수도 있고, 원래 데이터에 대한 승법 모형을 적합할 수도 있다.

로그변환한 데이터에 대한 가법 모형

In [74]:
```
m_smoothing <- HoltWinters(log_z_ts)
m_smoothing
```

```
Holt-Winters exponential smoothing with trend and additive seasonal component.

Call:
HoltWinters(x = log_z_ts)

Smoothing parameters:
 alpha: 0.3447498
 beta : 0.003585913
 gamma: 0.879488

Coefficients:
            [,1]
a     6.165298459
b     0.008744433
s1   -0.073696849
s2   -0.142822821
s3   -0.039964334
s4    0.015968620
s5    0.033673237
s6    0.157016140
s7    0.300626468
s8    0.285698068
s9    0.098491289
s10  -0.021898771
s11  -0.190036243
s12  -0.096734648
```

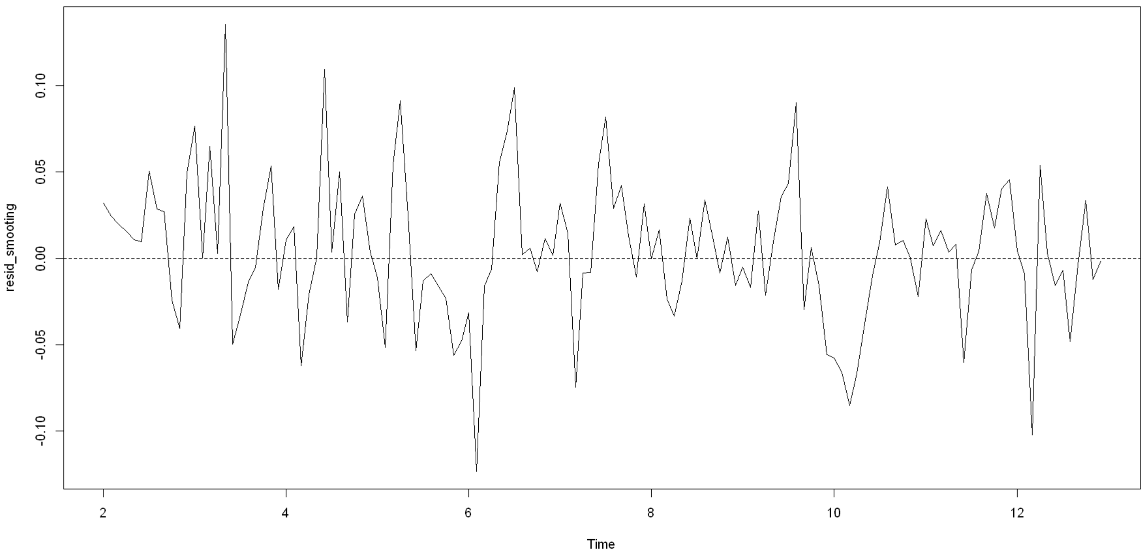
In [75]:
```
plot(m_smoothing, lwd=2, lty=1:2)
legend("topleft", legend=c("log_z", "HoltWinters"), col=1:2, lty=1:2, lwd=2)
```



잔차분석

In [76]:
```
resid_smooting <- resid(m_smoothing)
plot.ts(resid_smooting)
```

```
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [77]:
```
t.test(resid_smooting)
```

```
        One Sample t-test

data:  resid_smooting
t = 1.2026, df = 131, p-value = 0.2313
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.002766709  0.011346227
sample estimates:
  mean of x
0.004289759
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [78]:
```
lmtest::dwtest(lm(resid_smooting~1), alternative="two.sided")
```

```
        Durbin-Watson test

data:  lm(resid_smooting ~ 1)
DW = 1.4549, p-value = 0.001604
alternative hypothesis: true autocorrelation is not 0
```

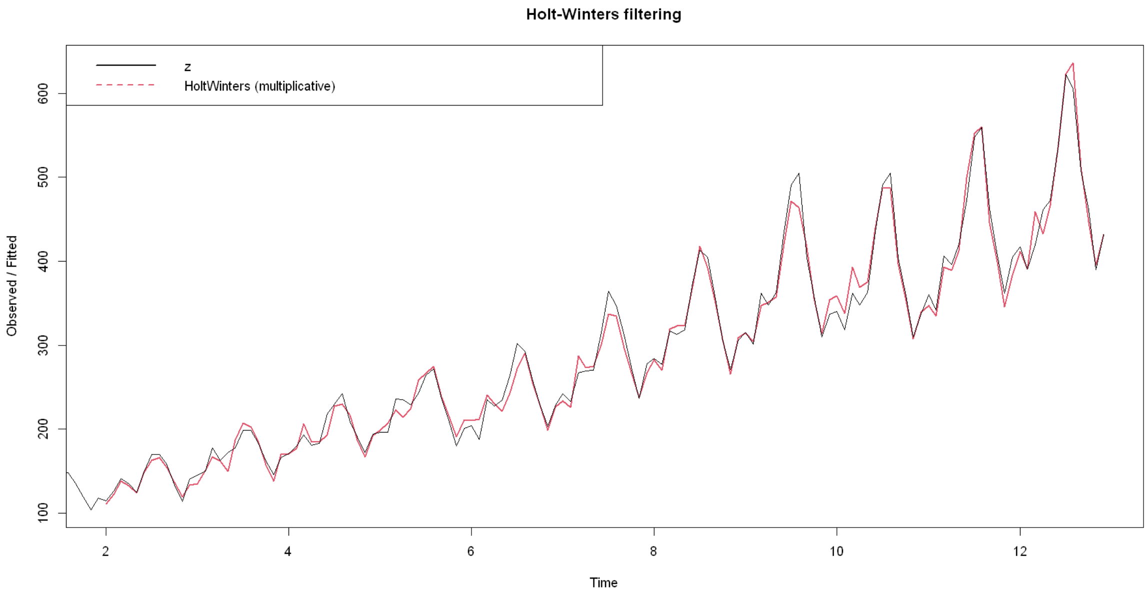
- DW test 결과 예측오차들 간의 자기 상관이 있는 것으로 보인다.
- 하지만, 추세모형을 적합하였을 때 얻어진 잔차에 비해 유의확률값이 크다.
- 예측오차는 분석 결과 추세모형 적합 결과보다는 평활 결과가 더 적절하다고 할 수 있다.

원 데이터에 대한 승법모형 적용

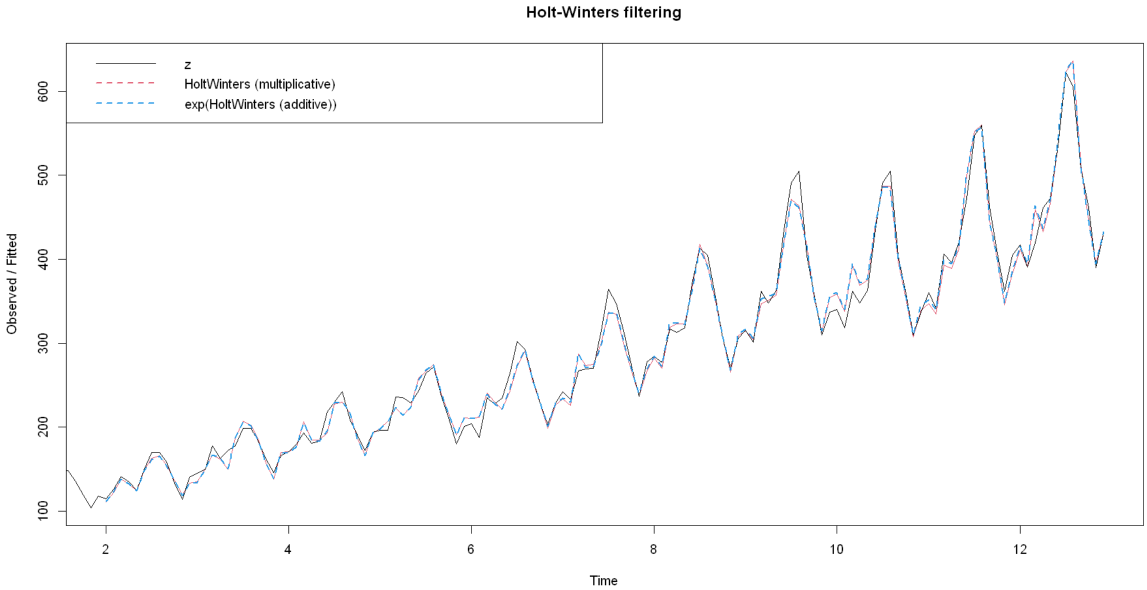
In [79]:
```
m_smoothing_multi <- HoltWinters(z_ts, seasonal = "multiplicative")
m_smoothing_multi
```

```
Holt-Winters exponential smoothing with trend and multiplicative seasonal compone
nt.

Call:
HoltWinters(x = z_ts, seasonal = "multiplicative")

Smoothing parameters:
 alpha: 0.3204565
 beta : 0.02388265
 gamma: 1

Coefficients:
           [,1]
a   464.9721064
b     2.7859138
s1    0.9464640
s2    0.8810274
s3    0.9648716
s4    1.0334898
s5    1.0470250
s6    1.1793118
s7    1.3631491
s8    1.3399425
s9    1.1195987
s10   0.9993794
s11   0.8438225
s12   0.9290880
```

In [80]:
```
plot(m_smoothing_multi, lwd=2, lty=1:2)
legend("topleft", legend=c("z", "HoltWinters (multiplicative)"), col=1:2, lty=1:
```

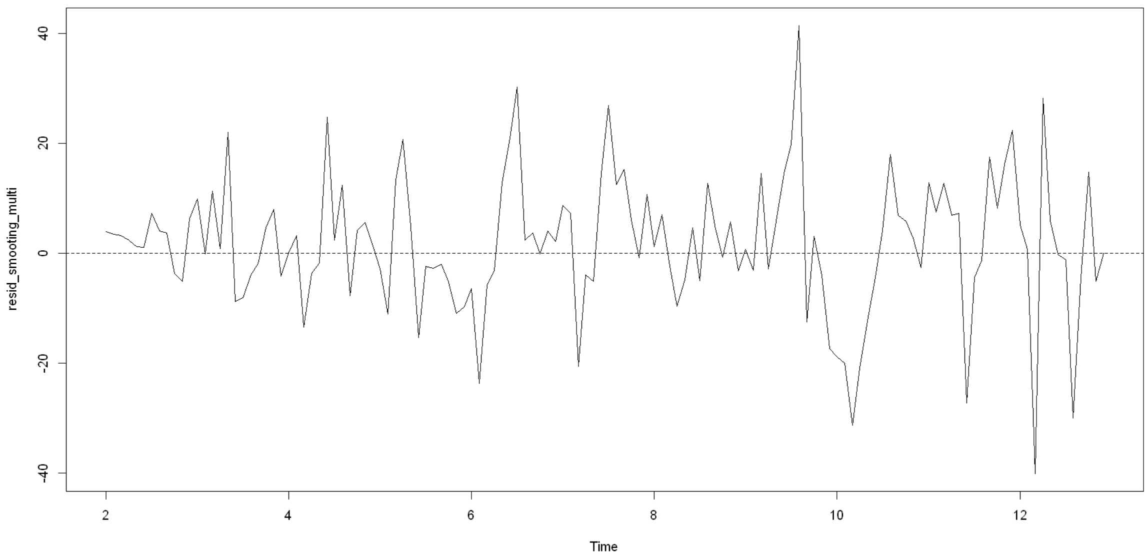


In [81]:
```
plot(m_smoothing_multi, lwd=1:2, lty=1:2)
lines(exp(m_smoothing$fitted[,1]), col=4, lty=2, lwd=2)
legend("topleft", legend=c("z", "HoltWinters (multiplicative)", "exp(HoltWinters
```



승법모형에 대한 잔차분석

In [82]:
```
resid_smooting_multi <- resid(m_smoothing_multi)
plot.ts(resid_smooting_multi)
abline(h=0, lty=2)
```



- 예측오차의 경우 0을 중심으로 대칭이고, 이분산성은 없어 보인다.

In [83]:
```
t.test(resid_smooting_multi)
```

```
	One Sample t-test

data:  resid_smooting_multi
t = 1.5746, df = 131, p-value = 0.1178
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.4315512  3.7983414
sample estimates:
mean of x 
 1.683395 
```

- t-test 결과 예측오차의 평균은 0이라고 할 수 있다.

In [84]:
```
lmtest::dwtest(lm(resid_smooting_multi~1), alternative="two.sided")
```

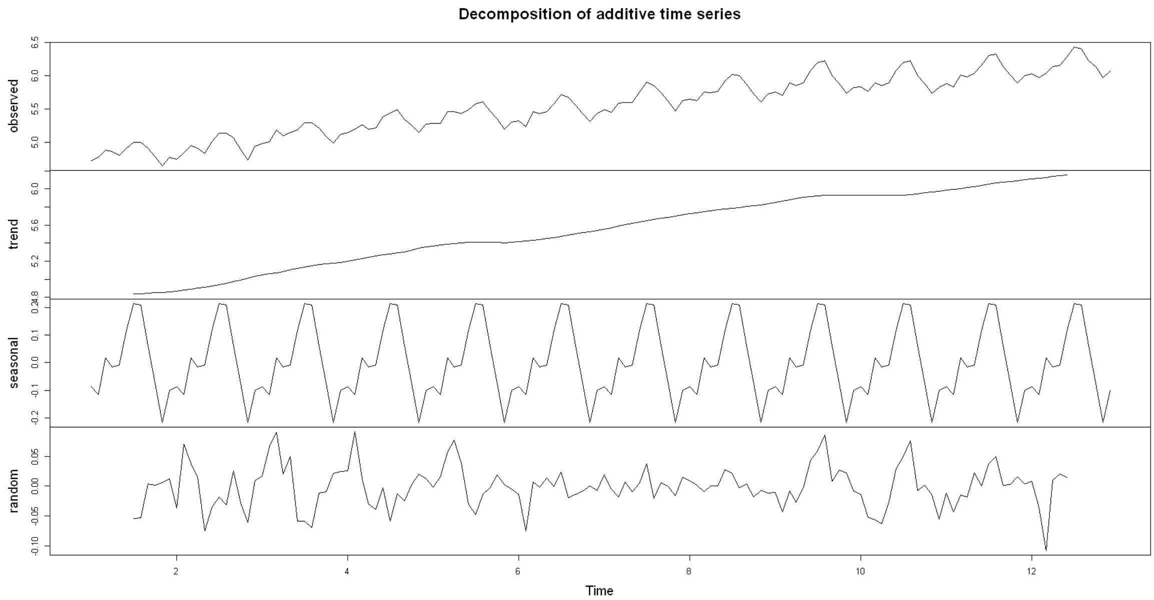
```
	Durbin-Watson test

data:  lm(resid_smooting_multi ~ 1)
DW = 1.4492, p-value = 0.001431
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 예측오차들 간의 자기 상관이 있는 것으로 보인다.
- 하지만, 추세모형을 적합하였을 때 얻어진 잔차에 비해 유의확률값이 크지만, 가법 모형을 사용한 평활법에 비하면 조금 크다.
- 차이가 크지 않지만, 가법 모형이 더 적절하다고 할 수 있다. 따라서 가법 모형 사용.

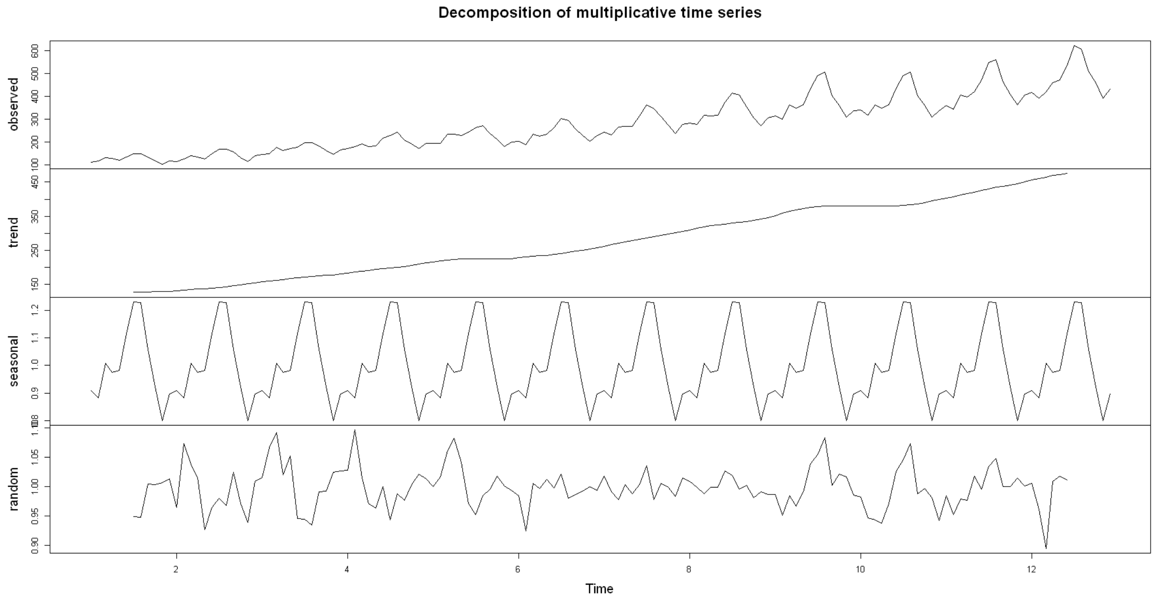
(4) 적절한 분해법에 의해 각 성분을 분해해여 시계열 그림을 그려라.

- 로그변환한 데이터에 대한 가법모형을 사용하거나, 원래 데이터에 대한 승법 모형 적합 가능
- decompose 함수 사용

In [85]:
```
m_decompose_add <- decompose(log_z_ts, 'additive')
plot(m_decompose_add)
```

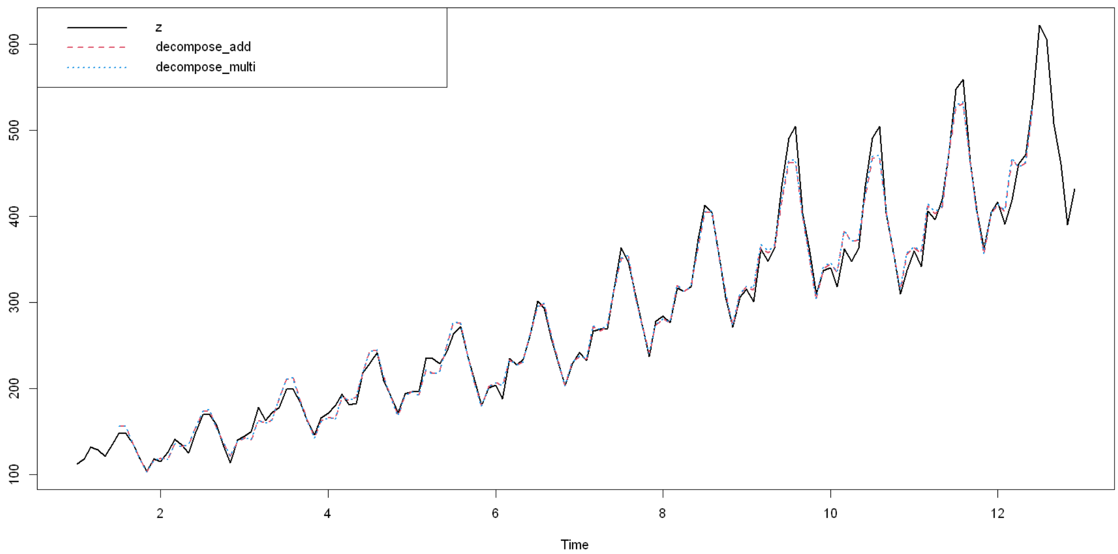


In [86]:
```
m_decompose_multi <- decompose(z_ts, 'multiplicative')
plot(m_decompose_multi)
```



In [87]:
```
fitted_dec_add <- m_decompose_add$trend + m_decompose_add$seasonal #가법모형은 더
fitted_dec_multi <- m_decompose_multi$trend * m_decompose_multi$seasonal  #승법도
```

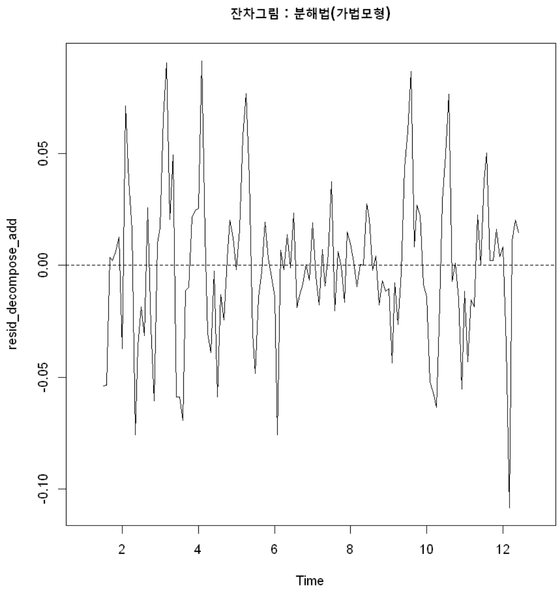
In [88]:
```
ts.plot(z_ts, exp(fitted_dec_add), fitted_dec_multi, col=c(1,2,4), lwd=2, lty=1:
legend("topleft", c("z","decompose_add", "decompose_multi"), col=c(1,2,4), lty=1
```

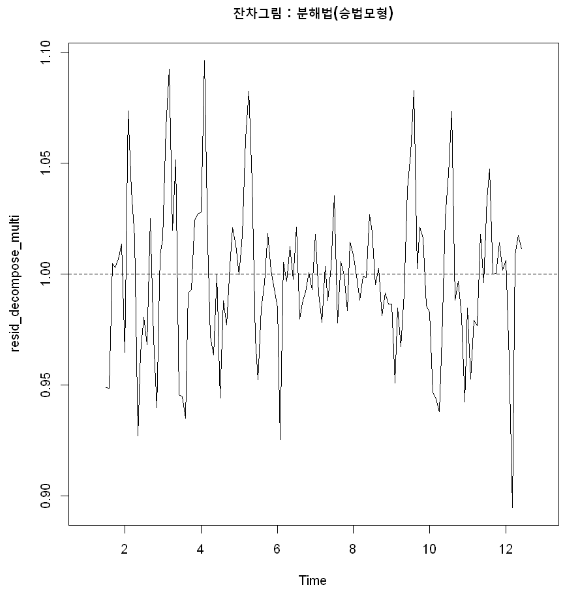


- 가법모형과 승법 모형은 차이가 거의 없어 보임

잔차분석을 하라는 문제는 없었지만 한 번 해보겠음.

In [89]:
```
resid_decompose_add <- m_decompose_add$random
resid_decompose_multi <- m_decompose_multi$random

par(mfrow=c(1,2))
plot.ts(resid_decompose_add, main="잔차그림 : 분해법(가법모형)")
abline(h=0, lty=2)
plot.ts(resid_decompose_multi, main="잔차그림 : 분해법(승법모형)")
abline(h=1, lty=2)
```





- 가법모형의 경우 0을 중심으로 대칭이고, 승법모형의 경우 1을 중심으로 대칭이다.
- 두 경우 모두 이분산성은 없어 보인다.

In [90]:
```
t.test(resid_decompose_add)
t.test(resid_decompose_multi, mu=1)
```

```
	One Sample t-test

data:  resid_decompose_add
t = -0.28335, df = 131, p-value = 0.7774
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.006909609  0.005178249
sample estimates:
  mean of x 
-0.00086568 
	One Sample t-test

data:  resid_decompose_multi
t = -0.59098, df = 131, p-value = 0.5556
alternative hypothesis: true mean is not equal to 1
95 percent confidence interval:
 0.992186 1.004219
sample estimates:
mean of x 
0.9982026 
```

- t-test 결과 가법모형에서 예측오차의 평균은 0이라고 할 수 있고, 승법모형에서 예측오차의 평균은 1이라고 할 수 있다.

In [91]:
```
lmtest::dwtest(lm(resid_decompose_add~1), alternative="two.sided")
lmtest::dwtest(lm(resid_decompose_multi~1), alternative="two.sided")
```

```
	Durbin-Watson test

data:  lm(resid_decompose_add ~ 1)
DW = 1.1441, p-value = 7.258e-07
alternative hypothesis: true autocorrelation is not 0
```

```
        Durbin-Watson test

data:  lm(resid_decompose_multi ~ 1)
DW = 1.1426, p-value = 6.946e-07
alternative hypothesis: true autocorrelation is not 0
```

- DW test 결과 가법,승법 모두 예측오차들 간의 자기 상관이 있는 것으로 보인다.

(5) 각 분석방법에 의한 결과를 1-시착 후 예측오차의 제곱합 (SSE) 기준하에서 비교하여라.

- 주의 : 예측오차의 제곱합을 구하기 위해서는 로그변환한 자료를 사용한 분석 방법의 경우 지수함수를 이용한 역변환을 한 후 계산해야 한다.

In [92]:
```
#예측오차
e_trend <- z-exp(fitted(m_trend))  #추세모형을 적합한 경우의 예측오차
e_smoothing_add <- z[-(1:12)]-exp(m_smoothing$fitted[,1]) #평활(가법)을 적합한 경
e_smoothing_multi <- z[-(1:12)]-m_smoothing_multi$fitted[,1] #평활(가법)을 적합한
e_decompose_add <- z - exp(fitted_dec_add) #분해법(가법)을 적합한 경우의 예측오차
e_decompose_multi <- z - fitted_dec_multi #분해법(승법)을 적합한 경우의 예측오차_
```

In [93]:
```
#SSE
sse1 <- sum(e_trend^2, na.rm=T)
sse2 <- sum(e_smoothing_add^2, na.rm=T)
sse3 <- sum(e_smoothing_multi^2, na.rm=T)
sse4 <- sum(e_decompose_add^2, na.rm=T)
sse5 <- sum(e_decompose_multi^2, na.rm=T)
```

In [94]:
```
paste0("추세분석에서의 SSE = ", sse1)
paste0("가법 계절평활법에 의한 SSE = ", sse2)
paste0("승법 계절평활법에 의한 SSE = ", sse3)
paste0("가법 분해법에 의한 SSE = ", sse4)
paste0("승법 분해법에 의한 SSE = ", sse5)
```

'추세분석에서의 SSE = 49268.5277996132'

'가법 계절평활법에 의한 SSE = 20103.3932111226'

'승법 계절평활법에 의한 SSE = 20138.5987482274'

'가법 분해법에 의한 SSE = 15284.0239261845'

'승법 분해법에 의한 SSE = 14748.922939659'

- 분해법에 의해 얻어진 예측오차의 제곱합이 가작 작았으며, 그 중 승법 모형을 적합한 결과의 SSE 값이 더 작았다.
- 추세분으로 얻어진 SSE가 가장 큰 값을 갖는다.

In [95]:
```
#SSE
mse1 <- mean(e_trend^2, na.rm=T)
mse2 <- mean(e_smoothing_add^2, na.rm=T)
mse3 <- mean(e_smoothing_multi^2, na.rm=T)
mse4 <- mean(e_decompose_add^2, na.rm=T)
mse5 <- mean(e_decompose_multi^2, na.rm=T)
```

In [96]:
```
paste0("추세분석에서의 MSE = ", mse1)
paste0("가법 계절평활법에 의한 MSE = ", mse2)
```

```
paste0("승법 계절평활법에 의한 MSE = ", mse3)
paste0("가법 분해법에 의한 MSE = ", mse4)
paste0("승법 분해법에 의한 MSE = ", mse5)
```

'추세분석에서의 MSE = 342.142554163981'

'가법 계절평활법에 의한 MSE = 152.298433417595'

'승법 계절평활법에 의한 MSE = 152.565142032026'

'가법 분해법에 의한 MSE = 115.788060046852'

'승법 분해법에 의한 MSE = 111.734264694387'

- MSE 역시 승법 분해법을 이용했을 때 가장 작은 값을 갖는다.

In [ ]:

```
```