HW 03

2021234640 이종현

Table of Contents

Figures

Figure 3-1

중간재 출하지수 데이터에 단순지수평활법을 적용한 예. 이때 평활 지수는 0.9를 사용하였다. 다음 시점의 예측이 전 시점의 결과에 아주 밀접하게 관련되어 있음을 확인할 수 있다.

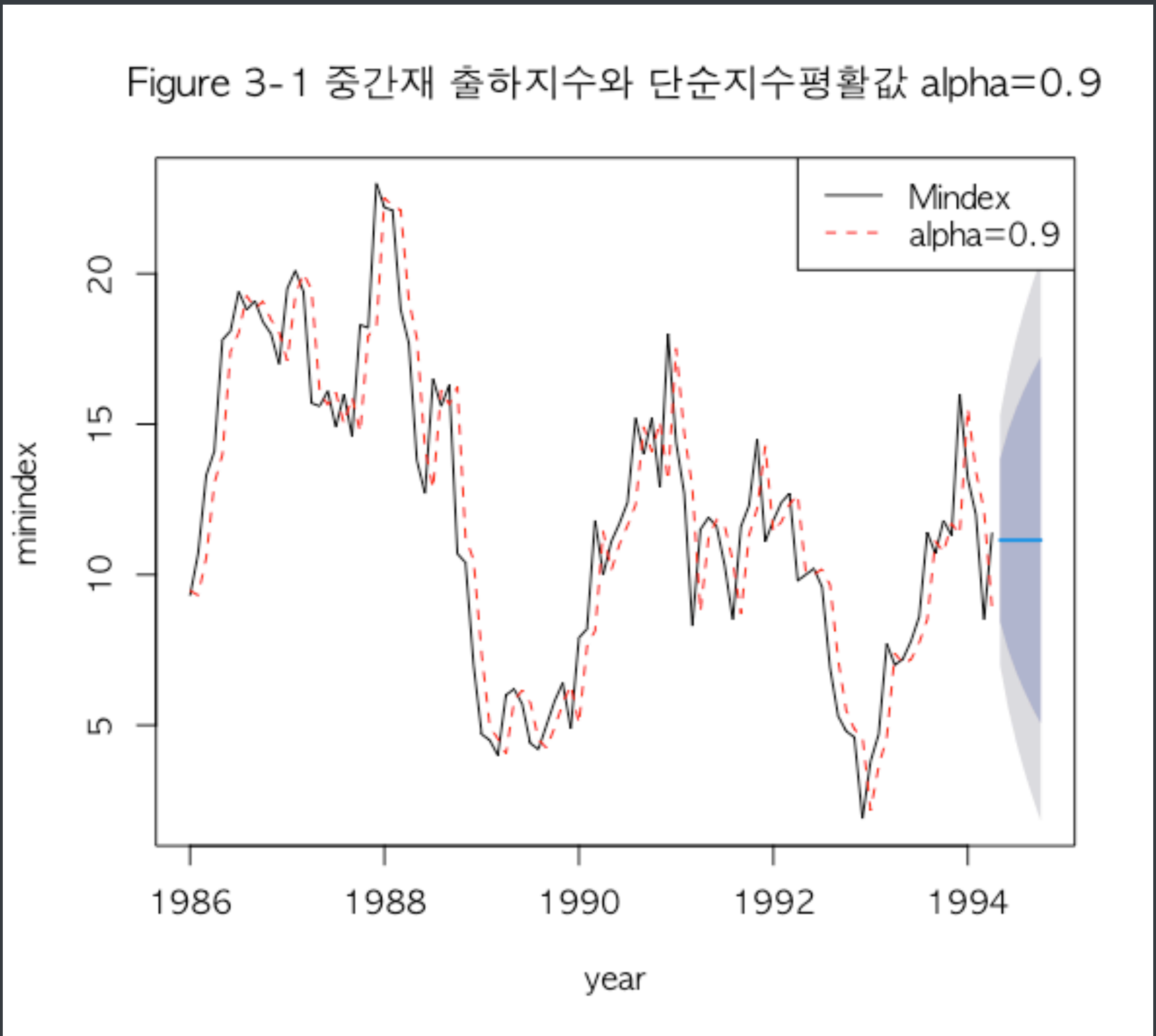


Figure 3-2

앞서 alpha를 0.9로 설정하였으나, 이는 임의적으로 설정된 모수이다. 피팅 후 예측 오차의 제곱합이 가장 작아지는 지점을 골라 alpha 값을 설정할 수 있다. 따라서 X축에는 alpha 값이 변화하는 상황을, Y축에는 설정된 alpha 수준에서 발생하는 오차의 제곱합을 플랏하였다. 이때 가장 오차가 작은 시점은 0.9 로 확인되었다.

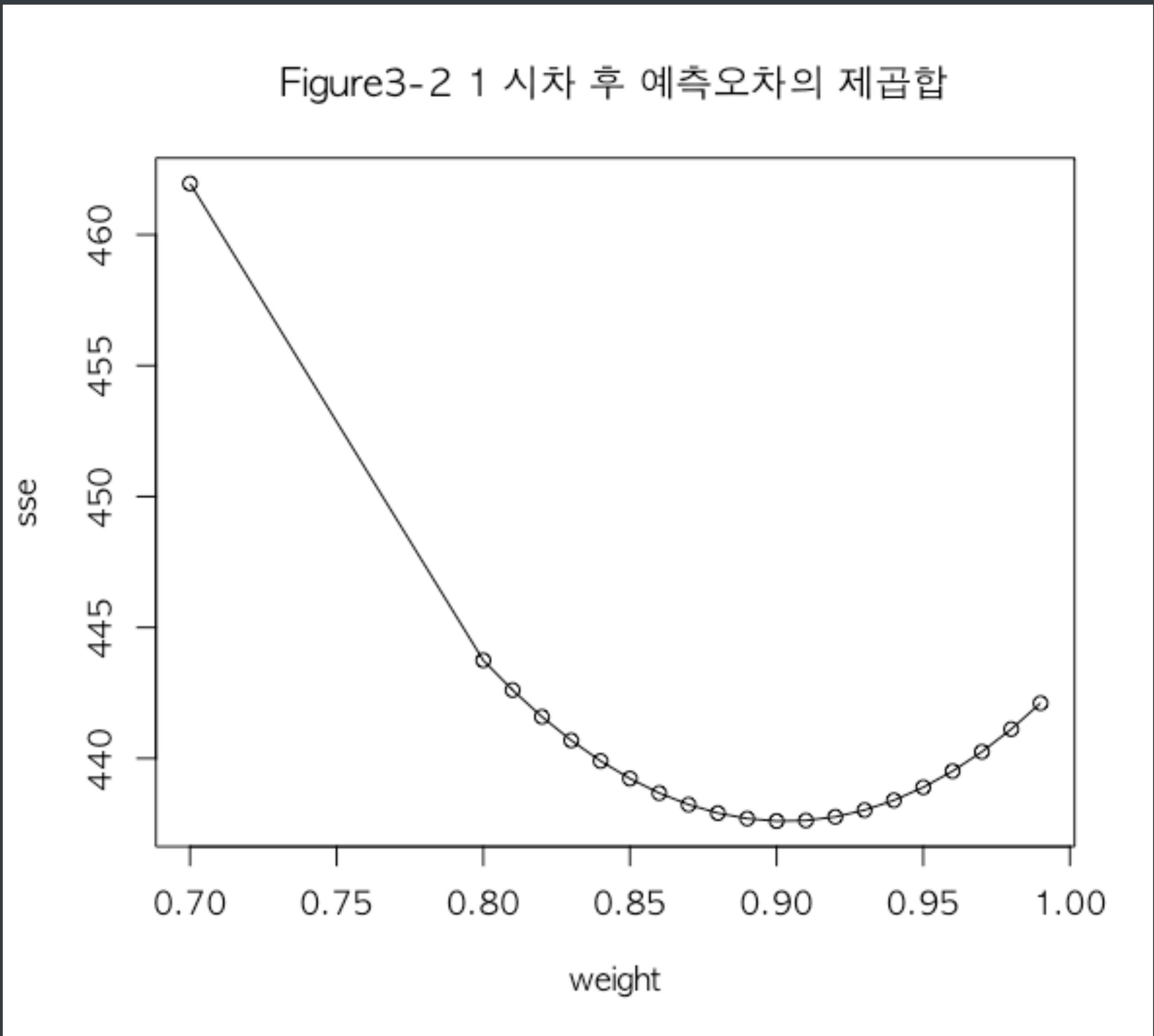


Figure 3-3

만약 alpha 가 최적 모수가 아닌 경우에는 어떻게 될까? 다음 피겨는 alpha를 0.2로 설정한 예시이다. 어느 정도 따라가는 경향성은 보이지만 앞서 0.9에서처럼 정확하게 따르지는 않는 것을 볼 수 있다.

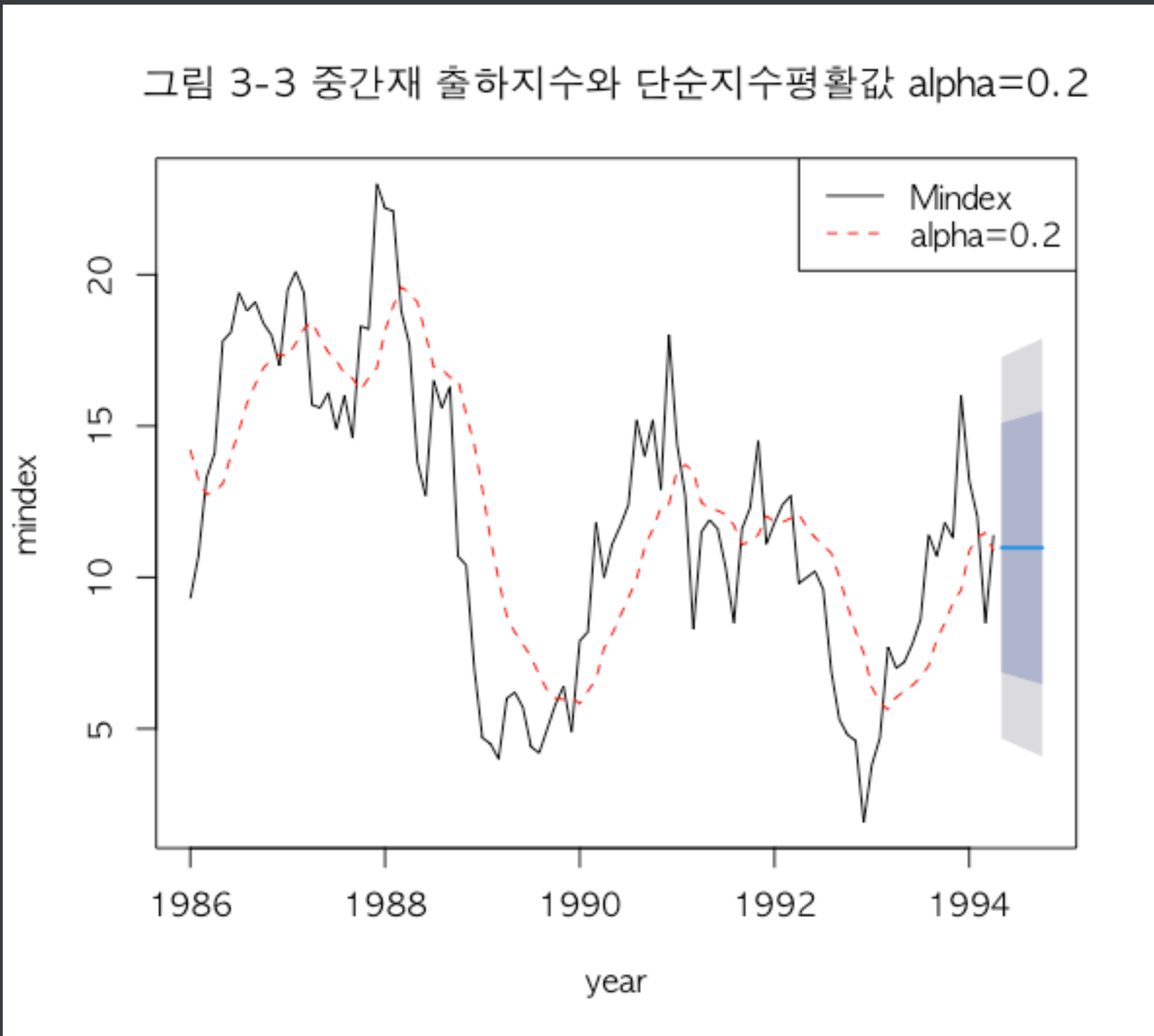
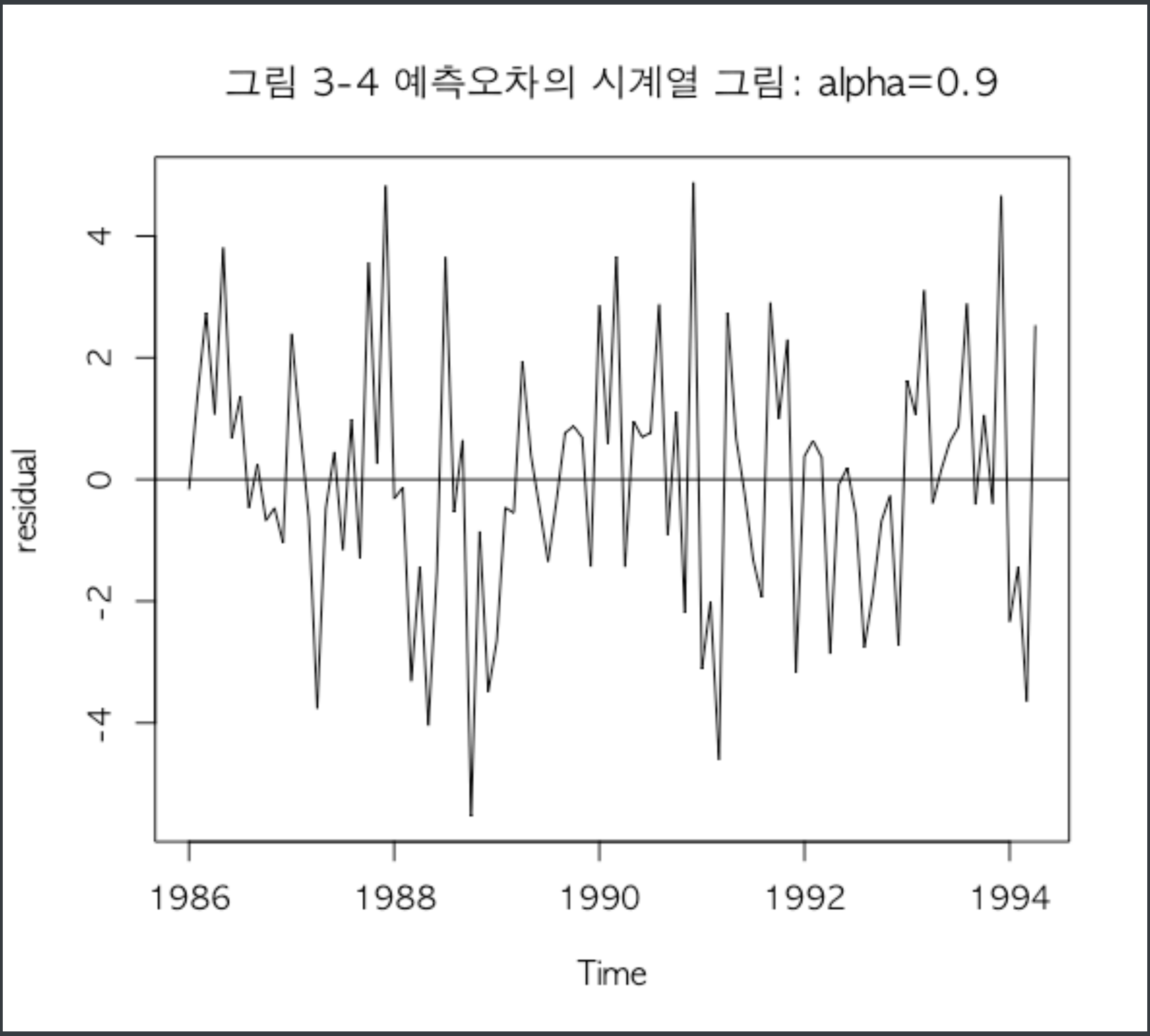


Figure 3-4

앞서 figure 3-1 에서 제시한 alpha를 0.9로 갖는 단순지수평활 모형의 잔차를 다음과 같이 그려보았다. residual이 특정 경향성을 띄지 않는 정상 오차로 보인다.

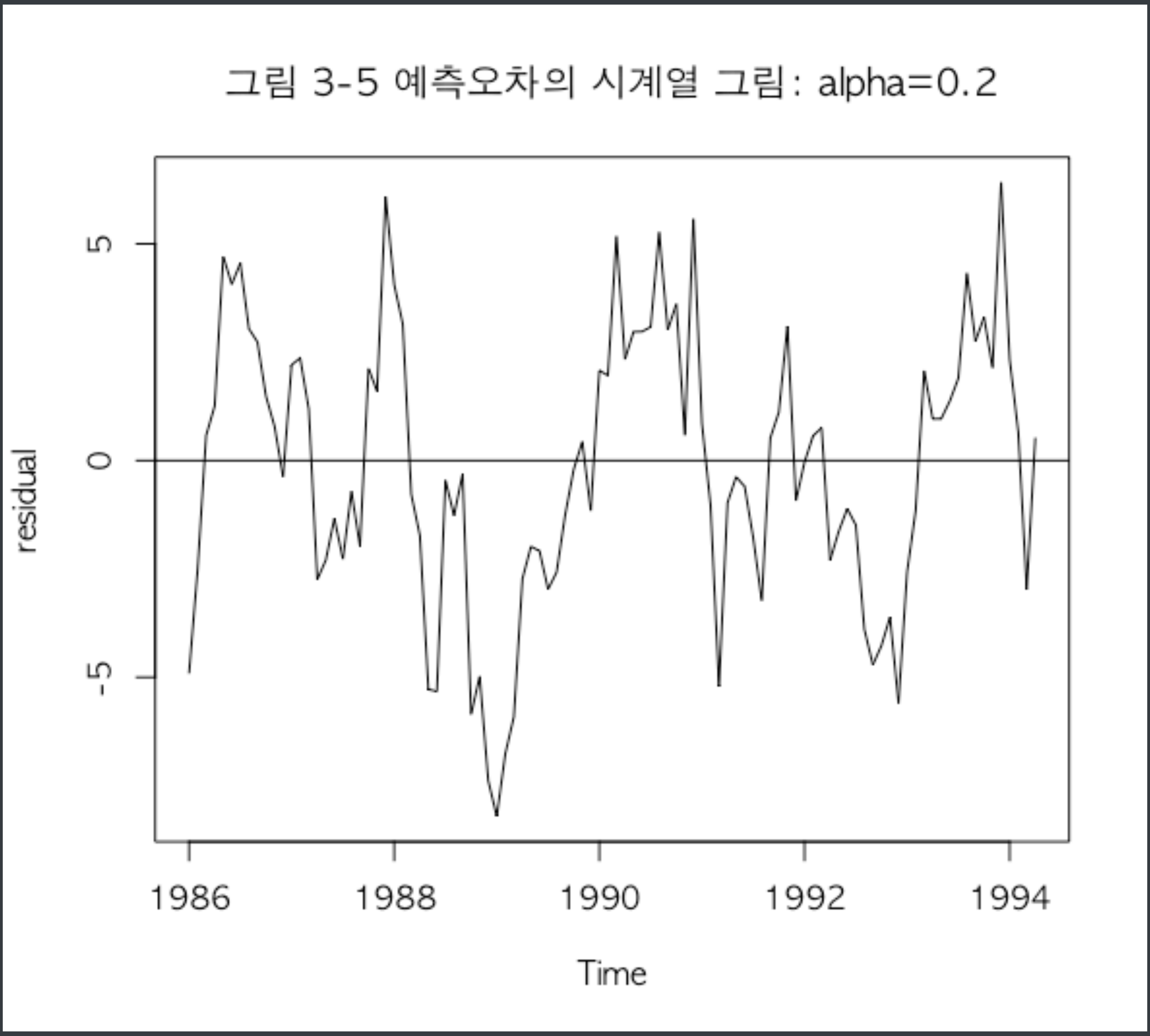


잔차가 0과 차이가 있는지 one-sample t-test를 진행하였으나, 통계적 차이는 없었다.

```
> t.test(resid(fit1), mu=0)

    One Sample t-test

data:  resid(fit1)
t = 0.088676, df = 99,
p-value = 0.9295
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.3985116  0.4357973
sample estimates:
 mean of x
0.01864288
```

Figure 3-5

한편 figure 3-3에서 본 것처럼 alpha가 0.2일 때의 오차는 주기적 움직임이 보인다.



각각 alpha가 0.9, 0.2 인 모델의 성능을 아래와 같이 비교하였다. alpha 0.9인 경우가 우수한 성능을 보임이 확인되었다.

```
> # 두 모형의 비교
> round(rbind(accuracy(fit1), accuracy(fit2)), digit=3)
                  ME  RMSE   MAE     MPE   MAPE  MASE   ACF1
Training set  0.019 2.092 1.616  -2.596 16.589 0.295 -0.015
Training set -0.162 3.178 2.595 -13.104 30.650 0.474  0.736
```

Figure 3-6

중간재 출하지수에 이중지수평활을 이용한 피팅 결과이다. 앞서 단순지수평활은 모수가 alpha 하나인 반면, 이중지수평활은 지수가 alpha, beta로 2개를 사용한다. 여기에서는 alpha, beta를 모두 0.6으로 설정하였다. 후반부에 예측이 상대적으로 부정확해지는 것을 볼 수 있다.

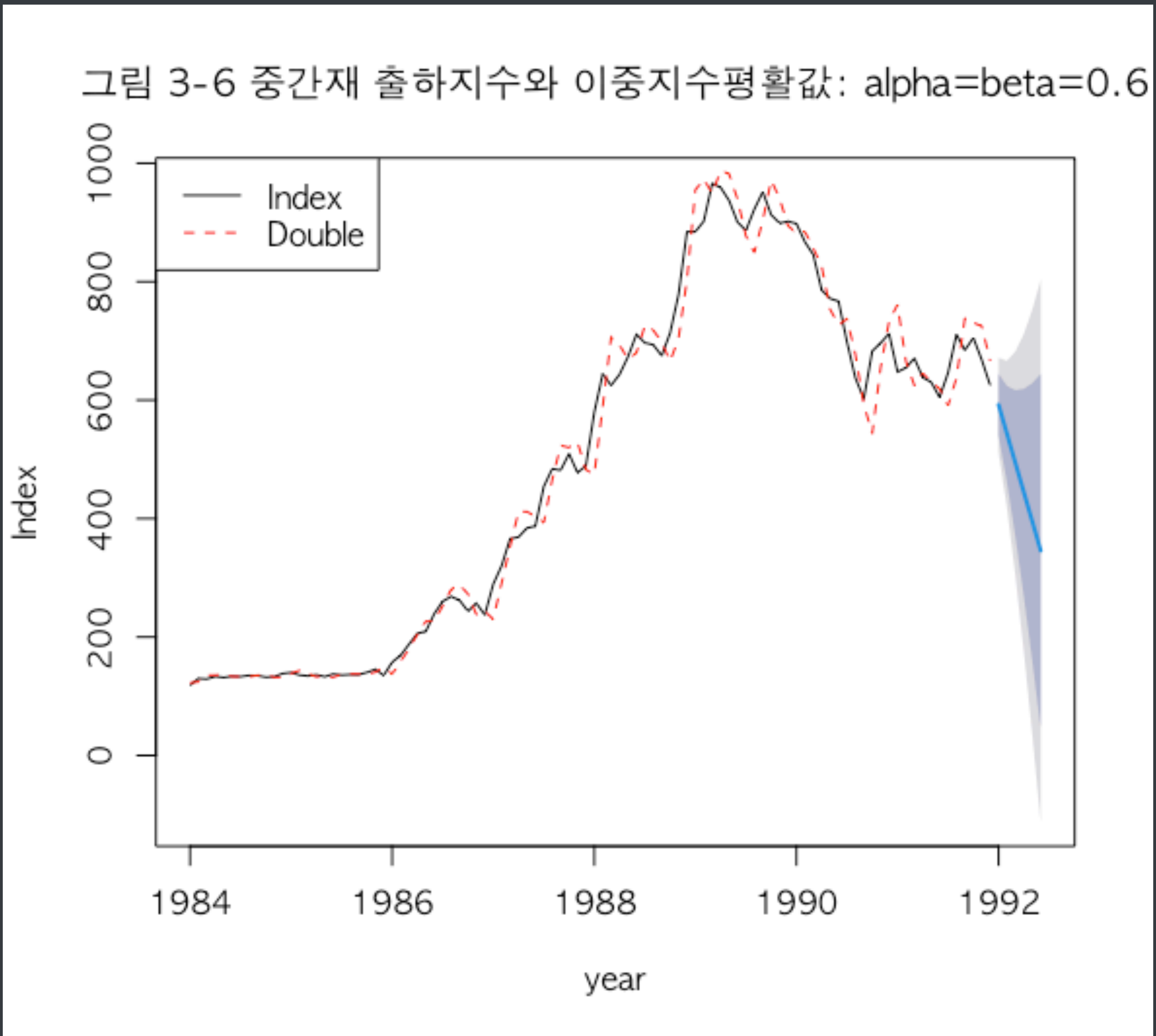
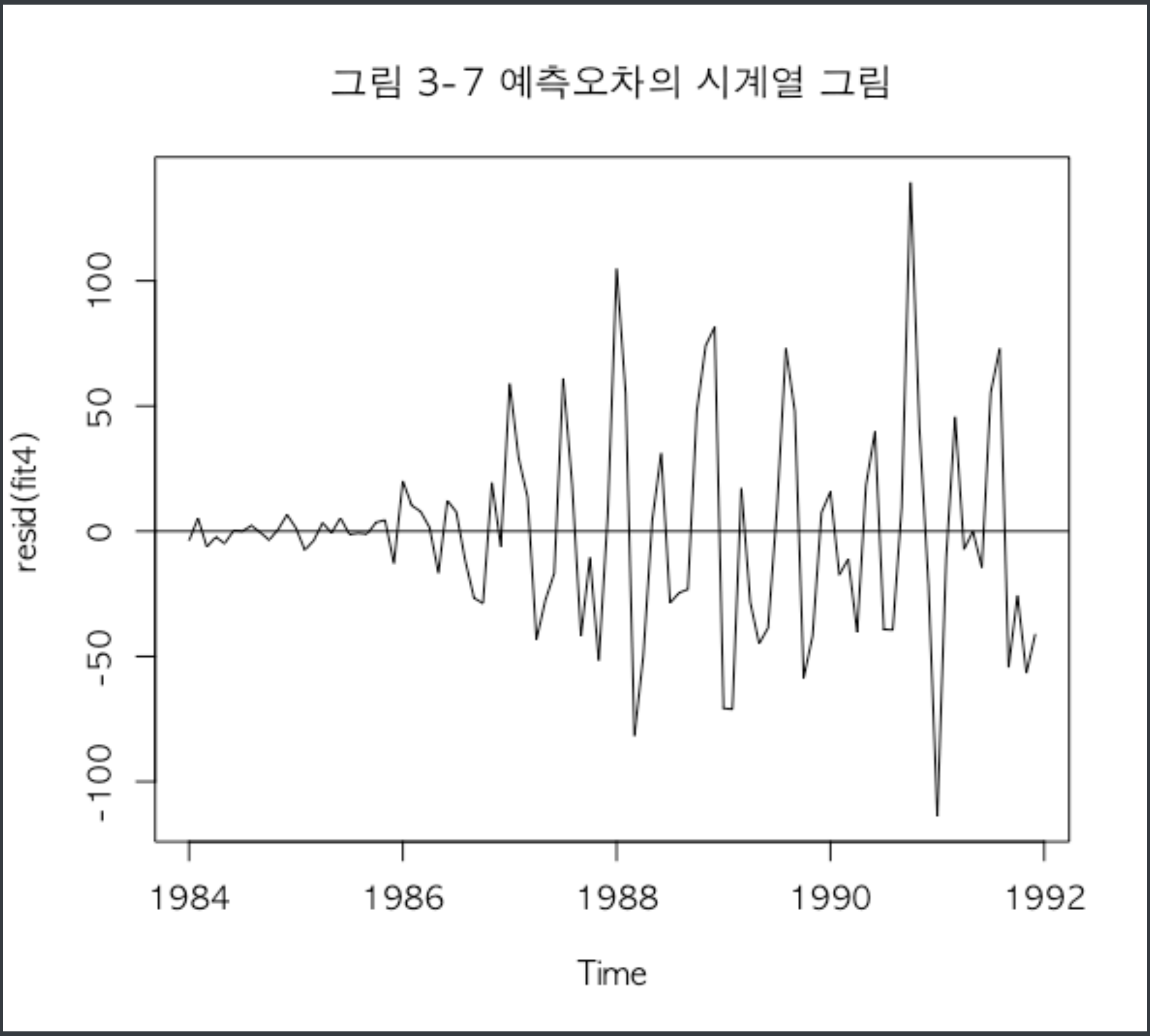


Figure 3-7

figure 3-6의 오차는 다음과 같다. 전반부에서는 차이가 거의 발생하지 않으나, 후반부에서 변동폭이 커지며 오차가 커지는 것을 확인할 수 있다. 이 경우, 분산이 달라지는 시점을 기준으로 다른 모수를 적용하는 방법도 있을 것 같다.



오차의 평균은 0과 차이는 없다.

Figure 3-8

관광객 방문 데이터를 Winter의 가법모형으로 피팅한 결과이다. 전반적인 예측 결과는 실제 값을 잘 따르는 것으로 보인다. 주기성과 선형 추세가 모두 보이므로 Winter의 가법모형은 적절한 선택으로 판단된다.

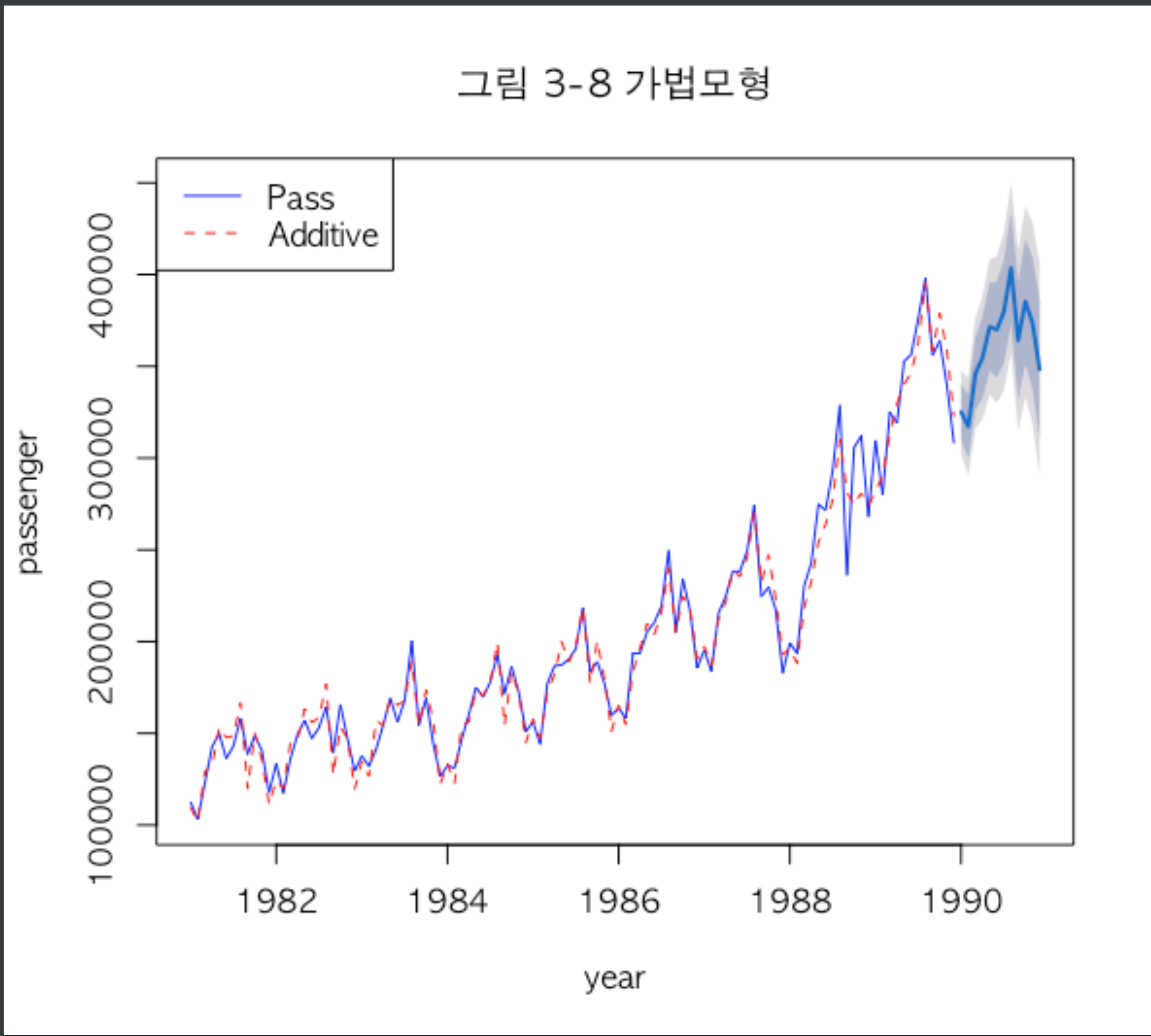


Figure 3-9

이번에는 가법모형이 아닌 승법모형으로 모델을 피팅하였다. 전반적인 결과가 앞서 가법모형보다 정교해진 것을 볼 수 있다.

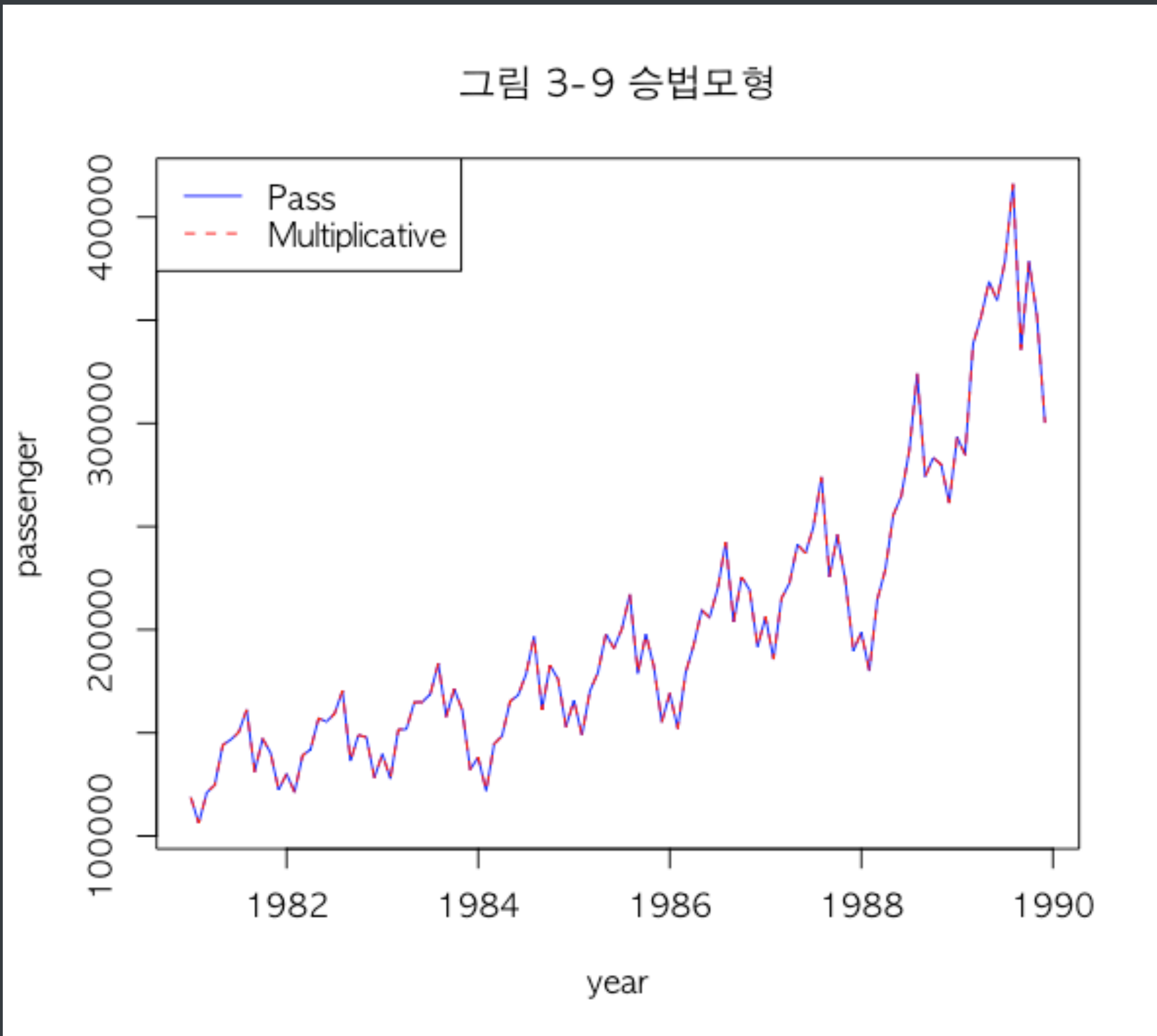
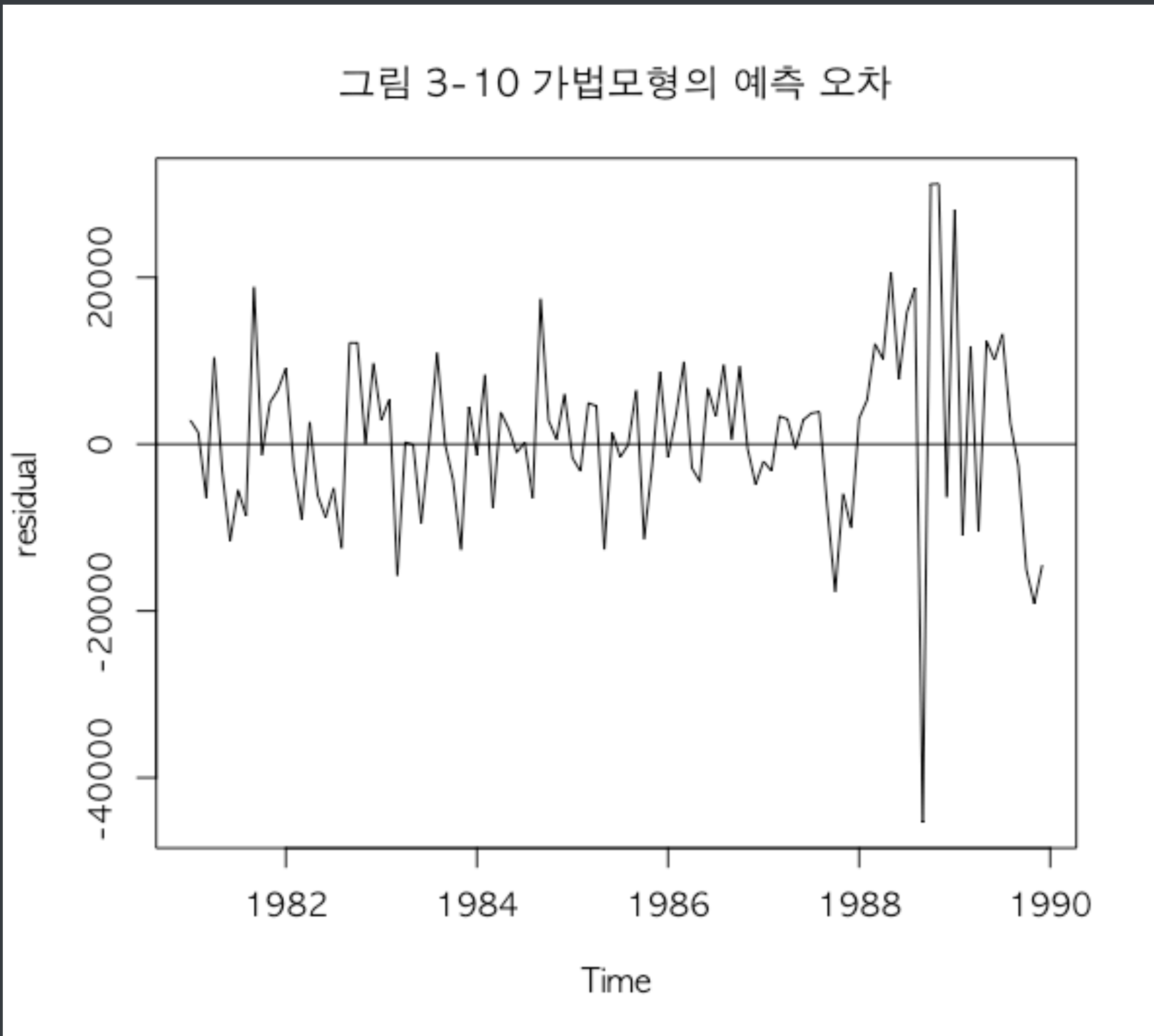


Figure 3-10

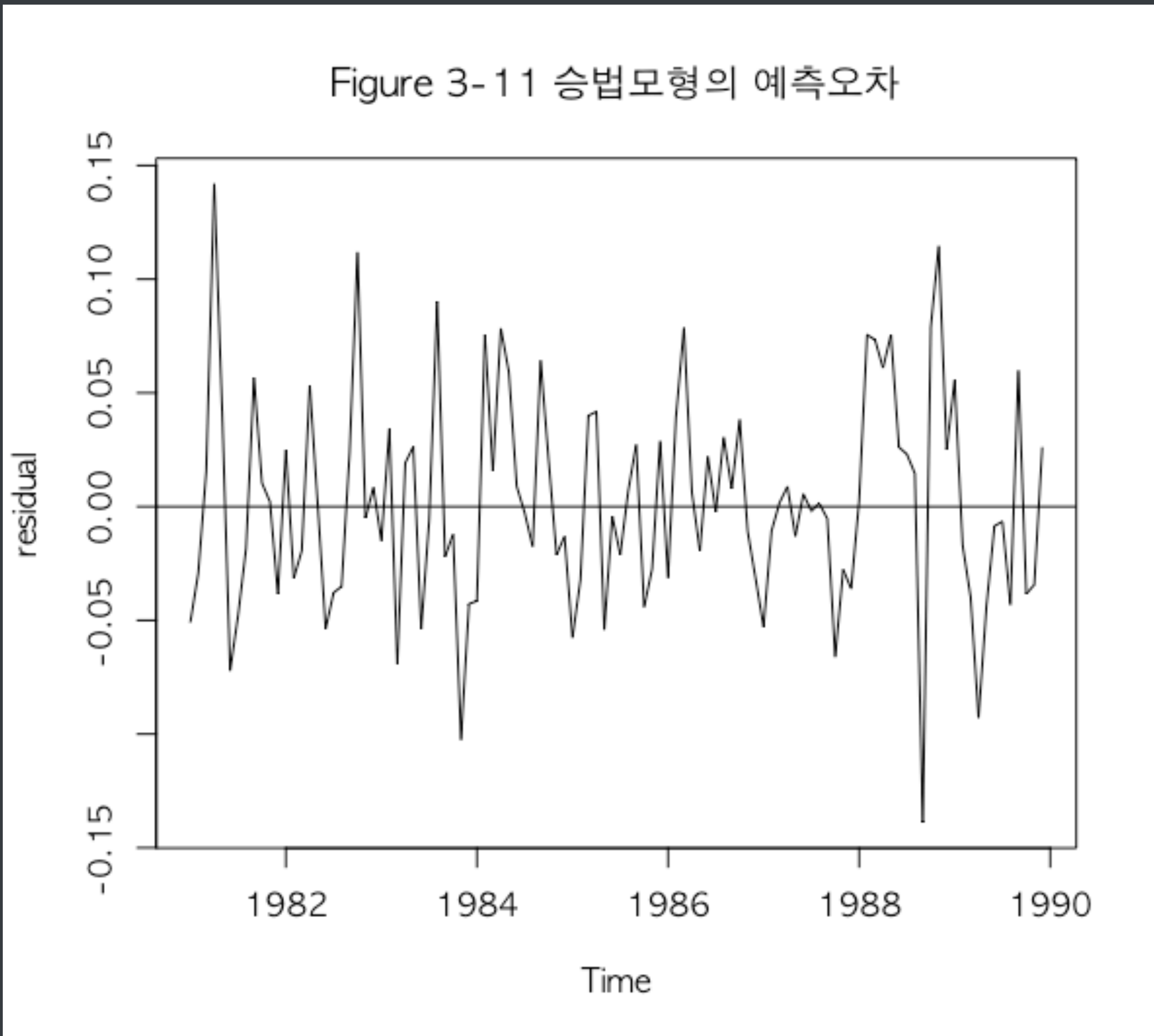
가법모형의 오차 그림이다. 1988~1990 사이에서 큰 오차가 발생하며 이때의 오차 폭은 점차 커지는 것으로 보인다.



오차가 유의하게 0과 차이를 보이지는 않는다.

Figure 3-11

승법모형의 오차 그림이다. 앞서 가법모형보다 오차의 폭이 작으며 상당히 잘 따라가고 있다는 것을 알 수 있다. 시간의 흐름에 따라 오차의 변동 폭의 변화도 잘 캐치해내고 있어 오차의 폭도 일정한 것으로 보인다.



오차가 유의하게 0과 차이를 보이지는 않는다.

이중지수평활의 기댓값

$$S_n^{(1)} = w z_n + (1-w) S_{n-1}^{(1)}$$

$$S_n^{(2)} = w S_n^{(1)} + (1-w) S_{n-1}^{(2)}$$

$$= \underbrace{w \sum (1-w)^j S_{n-j}^{(1)}}_{i)} + \underbrace{(1-w)^n S_0^{(2)}}_{ii)} \quad - ⓐ$$

$n \to \infty$ 이면 ⓐ식은

i) $\lim_{n \to \infty} w \sum (1-w)^j = \frac{w}{1-(1-w)} = 1$

ii) $\lim_{n \to \infty} (1-w)^n S_0^{(2)} = 0$

∴ ⓐ는 $S_n^{(2)} = w \sum_{j=0}^{n-1} (1-w)^j S_{n-j}^{(1)} \quad - ⓑ$

$E[S_{n-j}^{(1)}] = \beta_0 + \beta_1 (n+1) - \frac{1}{w}\beta_1$

ⓑ에 기댓값, $E\left[w \sum_{j=0}^{n-1} (1-w)^j S_{n-j}^{(1)}\right] = w \sum_{j=0}^{n-1} (1-w)^j E[S_{n-j}^{(1)}]$

$$= w \sum_{j=0}^{n-1} (1-w)^j E\left[\beta_0 + \beta_1 (n-j+1) - \frac{1}{w}\beta_1\right]$$

$$= \underbrace{w \sum (1-w)^j E\left[\beta_0 + \beta_1 (n+1) - \frac{1}{w}\beta_1\right]}_{①} \underbrace{- w\beta_1 \sum (1-w)^j j}_{②}$$

$E[S_n^{(1)}]$

$① = E[S_n^{(1)}]$

$$A \overset{let}{=} \sum_{j=0}^{\infty} (1-w)^j j$$

$$= 0 + (1-w) + (1-w)^2 \cdot 2 + \cdots$$

$$- \quad (1-w)A = \cancel{0 \cdot (1-w)} \cdot (1-w)^2 \cdot 1 + \cdots$$

$$wA = (1-w) + (1-w)^2 + (1-w)^3 + \cdots$$

$$= \frac{1-w}{1-(1-w)} = \frac{1-w}{w}$$

$$wA = \frac{(1-w)}{w} \rightsquigarrow A = \frac{(1-w)}{w^2}$$

②는 $w\beta_1 \cdot \frac{(1-w)}{w^2} = \frac{\beta_1(1-w)}{w}$

$$\therefore ① - ② = E[S_n^{(1)}] - \frac{(1-w)}{w}\beta_1 = E[S_n^{(2)}]$$

$$E[S_n^{(1)}] = \beta_{0,n} + \beta_{1,n}(n+1) - \frac{1}{w}\beta_1 \quad -①$$

$$E[S_n^{(2)}] = E[S_n^{(1)}] - \frac{(1-w)}{w}\beta_{1,n} \quad -②$$

$$E[S_n^{(1)}] - E[S_n^{(2)}] = \beta_{0,n} + \beta_{1,n}(n+1) - \frac{1}{w}\beta_1 - E[S_n^{(1)}] + \frac{(1-w)}{w}\beta_{1,n}$$

$$\beta_{0,n} = 2E[S_n^{(1)}] - E[S_n^{(2)}] - n \cdot \beta_{1,n}$$

$$\beta_{1,n} = \frac{w}{1-w}\left(E[S_n^{(1)}] - E[S_n^{(2)}]\right)$$

Exercise 3-4

3-4)

ⓑ $\hat{Z}_n(\ell) = \hat{\beta}_{0,n} + \hat{\beta}_{1,n}(n+\ell) + \hat{\beta}_{2,n}\frac{(n+\ell)^2}{2}$

이때 $\hat{\beta}_{0,n} = 3S_n^{(1)} - 3S_n^{(2)} + S_n^{(3)} - \frac{n}{2}\left(\frac{w}{(1-w)^2}\right) \times \left[(6-5w)S_n^{(1)} - 2(5-4w)S_n^{(2)} + (4-3w)S_n^{(3)}\right]$

$\qquad + \frac{n^2}{2}\left(\frac{w}{1-w}\right)^2 \left[S_n^{(1)} - 2S_n^{(2)} + S_n^{(3)}\right]$

$\hat{\beta}_{1,n} = \frac{w}{2(1-w)^2}\left[(6-5w)S_n^{(1)} - 2(5-4w)S_n^{(2)} + (4-3w)S_n^{(3)}\right] - n\left(\frac{w}{1-w}\right)^2\left[S_n^{(1)} - 2S_n^{(2)} + S_n^{(3)}\right]$

$\hat{\beta}_{2,n} = \frac{w^2}{(1-w)^2}\left[S_n^{(1)} - 2S_n^{(2)} + S_n^{(3)}\right]$ 이므로

$\hat{Z}_n(\ell) = S_n^{(1)}\Big(3 - \frac{n}{2}\left(\frac{w}{(1-w)^2}\right)(6-5w) + \frac{n^2}{2}\left(\frac{w}{1-w}\right)^2 + (n+\ell)\cdot\frac{w}{2(1-w)^2}\cdot(6-5w) - n(n+\ell)\left(\frac{w}{1-w}\right)^2$

$\qquad + \frac{(n+\ell)^2}{2}\cdot\frac{w^2}{(1-w)^2}\Big) + S_n^{(2)}\Big(-3 - \frac{n}{2}\left(\frac{w}{(1-w)^2}\right)\cdot(-2(5-4w)) - \frac{n^2}{2}\left(\frac{w}{1-w}\right)^2\cdot 2$

$\qquad + (n+\ell)\cdot\frac{w}{2(1-w)^2}\cdot -2(5-4w) - n(n+\ell)\left(\frac{w}{1-w}\right)^2\cdot -2 + \frac{(n+\ell)^2}{2}\cdot\frac{w^2}{(1-w)^2}\cdot -2\Big)$

$\qquad + S_n^{(3)}\Big(1 - \frac{n}{2}\left(\frac{w}{(1-w)^2}\right)\cdot(4-3w) + \frac{n^2}{2}\left(\frac{w}{1-w}\right)^2 + \frac{(n+\ell)}{2}\cdot\frac{w}{2(1-w)^2}\cdot(4-3w)$

$\qquad - n(n+\ell)\left(\frac{w}{1-w}\right)^2 + \frac{(n+\ell)^2}{2}\cdot\frac{w^2}{(1-w)^2}\Big)$

Appdendix: R code

```
rm(list=ls())
setwd("~/Workspace/2022-Fall_TimeSeriesAnalysis/")
par(family="AppleGothic")

# Example 3.1
library(forecast)
z = scan("data/mindex.txt")
mindex = ts(z, start=c(1986, 1), frequency=12)

w = c(seq(0.1, 0.8, 0.1), seq(0.81, 0.99, 0.01))
sse = sapply(w, function(x)
  return(sum(ses(mindex, alpha=x, h=6)$residuals^2)))
w1 = w[-c(1:6)]
sse1 = sse[-c(1:6)]

plot(w1, sse1, type="o", xlab="weight", ylab="sse",
     main="Figure3-2 1 시차 후 예측오차의 제곱합")
w[which.min(sse)]

fit1 = ses(mindex, alpha=0.9, h=6)
acf(resid(fit1), lag=12)
t.test(resid(fit1), mu=0)

plot(fit1, xlab="year", ylab="minindex",
     main="Figure 3-1 중간재 출하지수와 단순지수평활값 alpha=0.9",
     lty=1, col="black")
lines(fitted(fit1), col="red", lty=2)
legend("topright", legend=c("Mindex", "alpha=0.9"),
       lty=1:2, col=c("black", "red"))

plot(fit1$residuals, ylab="residual",
     main="그림 3-4 예측오차의 시계열 그림: alpha=0.9")
abline(h=0)
```

```r
fit2 = ses(mindex, alpha=0.2, h=6)
t.test(resid(fit2), mu=0)
acf(resid(fit2), lag.max=12)
plot(fit2, xlab="year", ylab="mindex",
     main="그림 3-3 중간재 출하지수와 단순지수평활값 alpha=0.2",
     lty=1, col="black")
lines(fitted(fit2), col="red", lty=2)
legend("topright", legend=c("Mindex", "alpha=0.2"),
       lty=1:2, col=c("black", "red"))

plot(fit2$residuals, ylab="residual",
     main="그림 3-5 예측오차의 시계열 그림: alpha=0.2")
abline(h=0)

# 두 모형의 비교
round(rbind(accuracy(fit1), accuracy(fit2)), digit=3)

fit3 = ses(mindex, h=6)
fit3$model
plot(fit3, xlab="year", ylab="mindex",
     main="중간재 출하지수와 단순지수평활값 alpha estimated",
     lty=1, col="black")
lines(fit3$fitted, col="red", lty=2)
legend("topright", legend=c("Mindex", "estimated_alpha"),
       lty=1:2, col=c("black", "red"))

plot(fit3$residuals, ylab="residual",
     main="예측오차의 시계열그림: 추정된 alpha")
abline(h=0)

# Example 3.2
library(forecast)
z = scan("data/stock.txt")
stock = ts(z, start=c(1984, 1), frequency=12)
```

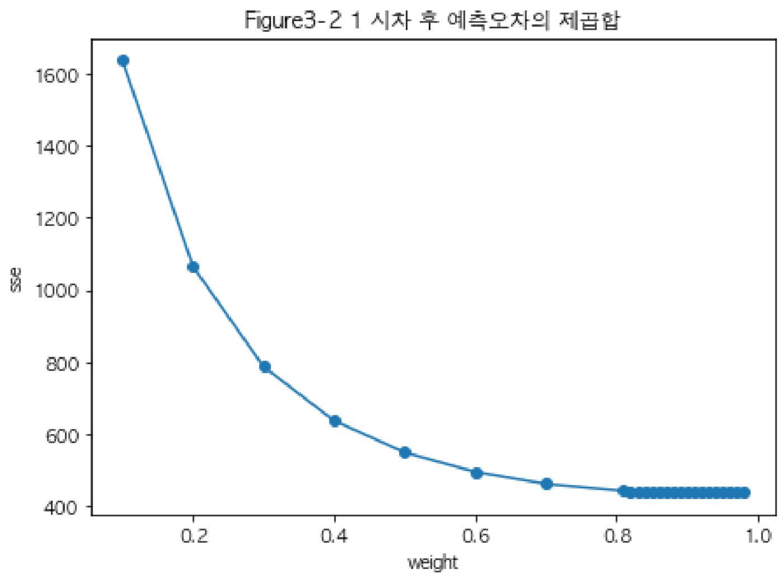
```r
# 1모수 이중지수평활
fit4 = holt(stock, alpha=0.6, beta=0.6, h=6)
fit4$model

plot(fit4, ylab="Index", xlab="year",
     lty=1, col="black",
     main="그림 3-6 중간재 출하지수와 이중지수평활값: alpha=beta=0.6")
lines(fitted(fit4), col="red", lty=2)
legend("topleft", lty=1:2, col=c("black", "red"), c("Index", "Double"))

plot(resid(fit4), main="그림 3-7 예측오차의 시계열 그림")
abline(h=0)

acf(resid(fit4))
t.test(resid(fit4), mu=0)


# 2모수 이중지수평활 alpha, beta 추정
fit5 = holt(stock, h=6)
fit5$model
plot(fit5, ylab="Index", xlab="year", lty=1, col="black",
     main="중간재 출하지수와 이중지수평활값: alpha, beta estimated")
lines(fitted(fit5), col="red", lty=2)
legend("topright", lty=1:2, col=c("black", "red"), c("Index", "Double"))

plot(resid(fit5), main="예측오차의 시계열그림: alpha, beta estimated")
abline(h=0)
acf(resid(fit5))
t.test(resid(fit5), mu=0)

## Example 3.3
library(forecast)
library(astsa)
z = scan("data/koreapass.txt")
pass = ts(z, start=c(1981, 1), frequency=12)
```

```
fit6 = hw(pass, seasonal="additive", h=12)
fit6$model

plot(fit6, ylab="passenger", xlab="year", lty=1, col="blue",
     main="그림 3-8 가법모형")
lines(fit6$fitted, col="red", lty=2)
legend("topleft", lty=1:2, col=c("blue", "red"), c("Pass", "Additive"))

ts.plot(resid(fit6), ylab="residual",
        main="그림 3-10 가법모형의 예측 오차")
abline(h=0)

acf(resid(fit6), main="Residual ACF")
t.test(resid(fit6), mu=0)

# holt winters multiplication model
fit7 = hw(pass, seasonal="multiplicative", h=12)
fit7$model

plot(fit7$fitted, ylab="passenger", xlab="year", lty=1, col="blue",
     main="그림 3-9 승법모형")
lines(fit7$fitted, col="red", lty=2)
legend("topleft", lty=1:2, col=c("blue", "red"), c("Pass",
"Multiplicative"))

ts.plot(resid(fit7), ylab="residual",
        main="Figure 3-11 승법모형의 예측오차")
abline(h=0)

acf(resid(fit7), main="Residual ACF")
t.test(resid(fit7), mu=0)
```

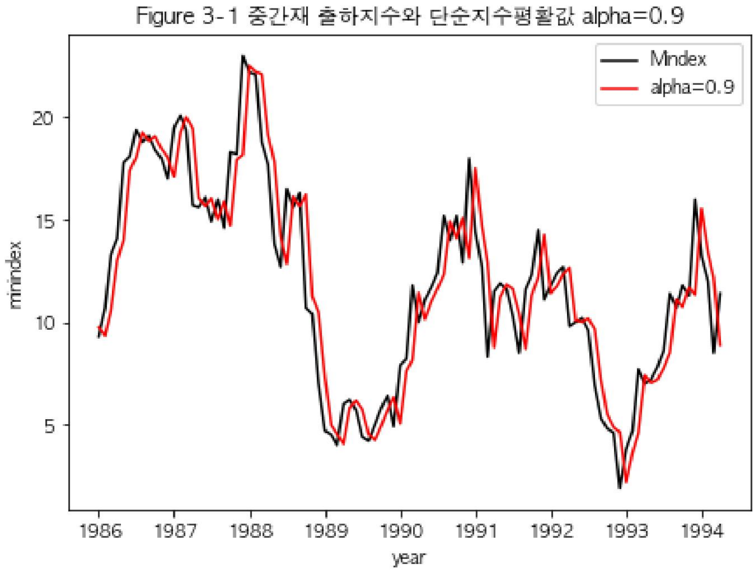
Appdendix: Python code

```
In [1]: import math
        import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        plt.rc('font', family='AppleGothic')
        plt.rcParams['axes.unicode_minus'] = False

        import matplotlib.dates as mdates

        from statsmodels.tsa.api import ExponentialSmoothing, SimpleExpSmoothing, Holt
```

```
In [2]: # Example 3.1
        z = []

        with open('../data/mindex.txt') as f:
            for line in f.readlines():
                for elem in line.rstrip().split(" "):
                    if len(elem):
                        z.append(float(elem))

        index = pd.date_range(start="1986", periods=len(z), freq="MS")
        data = pd.Series(z, index)

        w1 = np.arange(0.1, 0.8, 0.1)
        w2 = np.arange(0.81, 0.99, 0.01)
        w = np.hstack([w1, w2])

        residuals = []
        for w_ in w:
            model = SimpleExpSmoothing(data, initialization_method="heuristic")
            results = model.fit(smoothing_level=w_, optimized=False)
            residuals.append(np.sum(results.resid**2))


        fig, ax = plt.subplots(figsize=(7, 5))
        ax.plot(w, residuals, 'o-')
        ax.set_xlabel("weight")
        ax.set_ylabel("sse")
        ax.set_title("Figure3-2 1 시차 후 예측오차의 제곱합")
        plt.show()
```

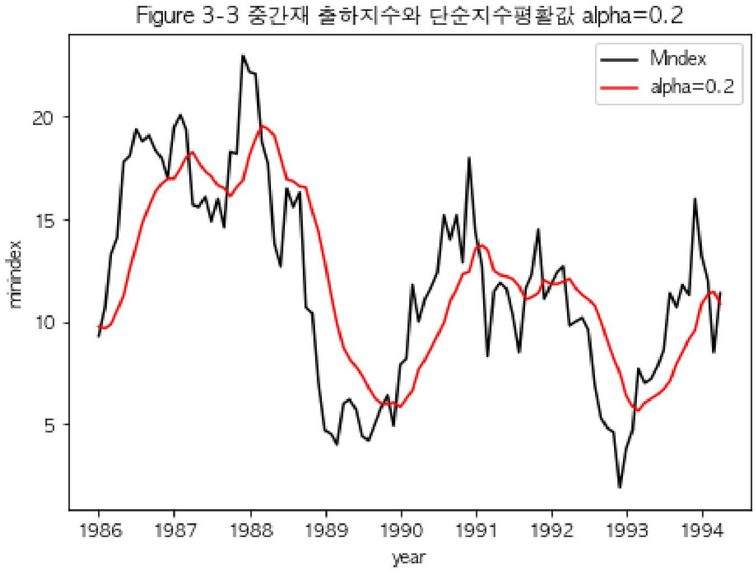


In [3]:
```python
fit1 = SimpleExpSmoothing(data, initialization_method="heuristic")
results = model.fit(smoothing_level=0.9, optimized=False)

fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(data, 'black', label="Mindex")
ax.plot(results.fittedvalues, 'red', label="alpha=0.9")
ax.set_xlabel("year")
ax.set_ylabel("minindex")
ax.set_title("Figure 3-1 중간재 출하지수와 단순지수평활값 alpha=0.9")
plt.legend()
plt.show()
```

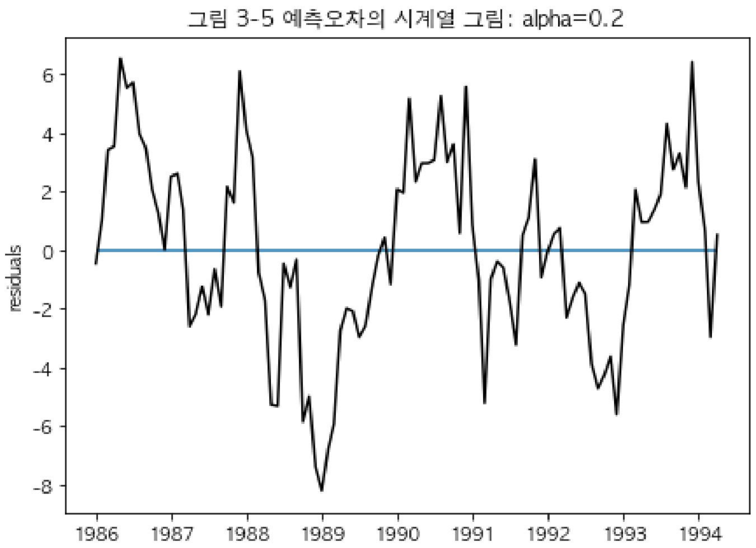


In [4]:
```python
fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(results.resid, 'black')
ax.set_ylabel("residuals")
ax.set_title("그림 3-4 예측오차의 시계열 그림: alpha=0.9")
ax.hlines(0, min(data.index), max(data.index))
plt.show()
```

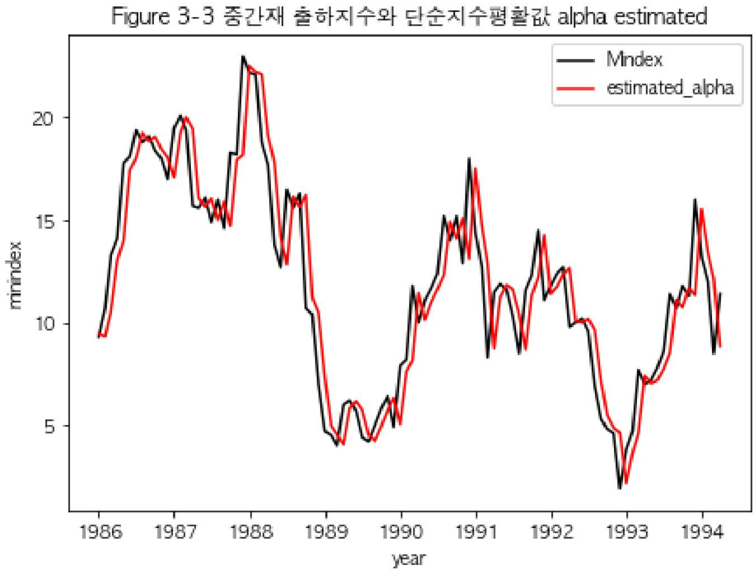
```
In [5]: fit1 = SimpleExpSmoothing(data, initialization_method="heuristic")
        results = model.fit(smoothing_level=0.2, optimized=False)

        fig, ax = plt.subplots(figsize=(7, 5))
        ax.plot(data, 'black', label="Mindex")
        ax.plot(results.fittedvalues, 'red', label="alpha=0.2")
        ax.set_xlabel("year")
        ax.set_ylabel("minindex")
        ax.set_title("Figure 3-3 중간재 출하지수와 단순지수평활값 alpha=0.2")
        plt.legend()
        plt.show()
```



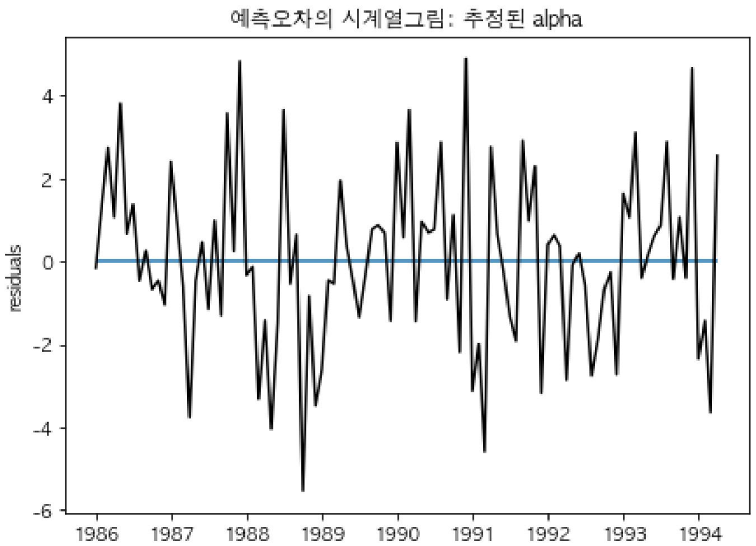

```
In [6]: fig, ax = plt.subplots(figsize=(7, 5))
        ax.plot(results.resid, 'black')
        ax.set_ylabel("residuals")
        ax.set_title("그림 3-5 예측오차의 시계열 그림: alpha=0.2")
        ax.hlines(0, min(data.index), max(data.index))
        plt.show()
```



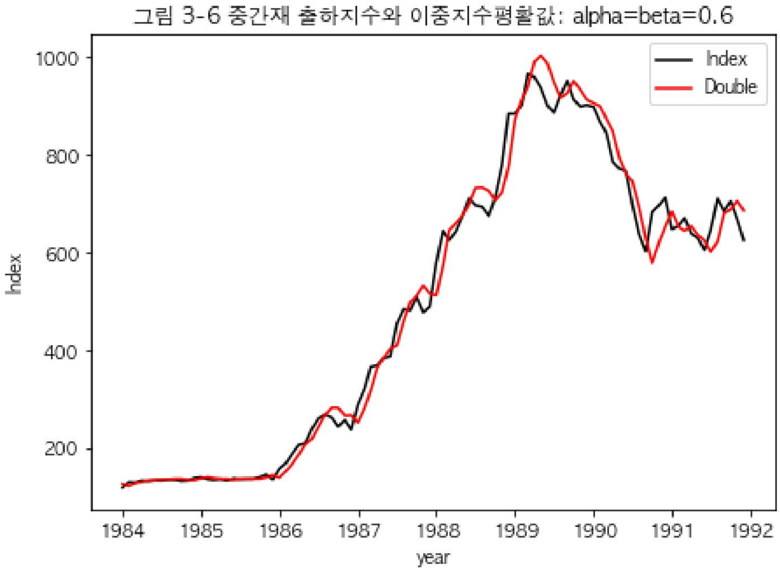
In [7]:
```
fit3 = SimpleExpSmoothing(data, initialization_method="heuristic")
results = model.fit()

fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(data, 'black', label="Mindex")
ax.plot(results.fittedvalues, 'red', label="estimated_alpha")
ax.set_xlabel("year")
ax.set_ylabel("minindex")
ax.set_title("Figure 3-3 중간재 출하지수와 단순지수평활값 alpha estimated")
plt.legend()
plt.show()
```



In [8]:
```
fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(results.resid, 'black')
ax.set_ylabel("residuals")
ax.set_title("예측오차의 시계열그림: 추정된 alpha")
ax.hlines(0, min(data.index), max(data.index))
plt.show()
```

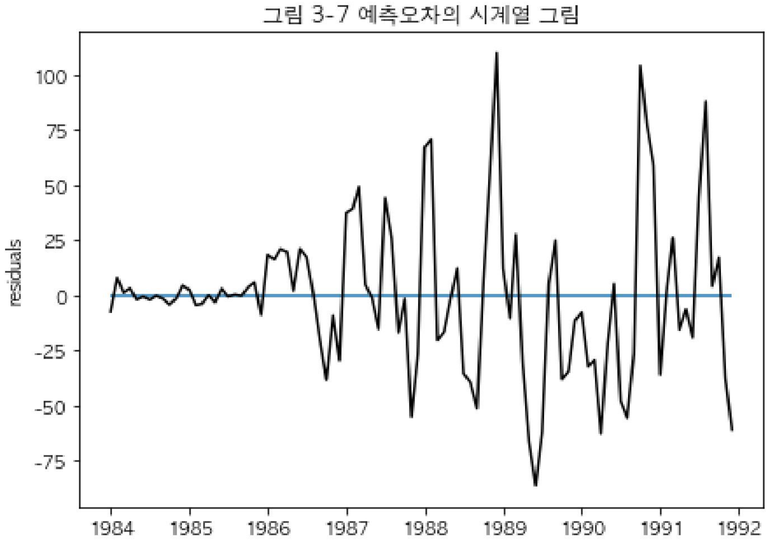


In [9]:
```
# Example 3.2
z = []

with open('../data/stock.txt') as f:
    for line in f.readlines():
        for elem in line.rstrip().split(" "):
            if len(elem):
                z.append(float(elem))

index = pd.date_range(start="1984", periods=len(z), freq="MS")
data = pd.Series(z, index)

fit4 = Holt(data, initialization_method="heuristic")
results = fit4.fit(smoothing_level=0.6, smoothing_trend=0.2, optimized=False)

fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(data, 'black', label="Index")
ax.plot(results.fittedvalues, 'red', label="Double")
ax.set_xlabel("year")
ax.set_ylabel("Index")
ax.set_title("그림 3-6 중간재 출하지수와 이중지수평활값: alpha=beta=0.6")
plt.legend()
plt.show()
```

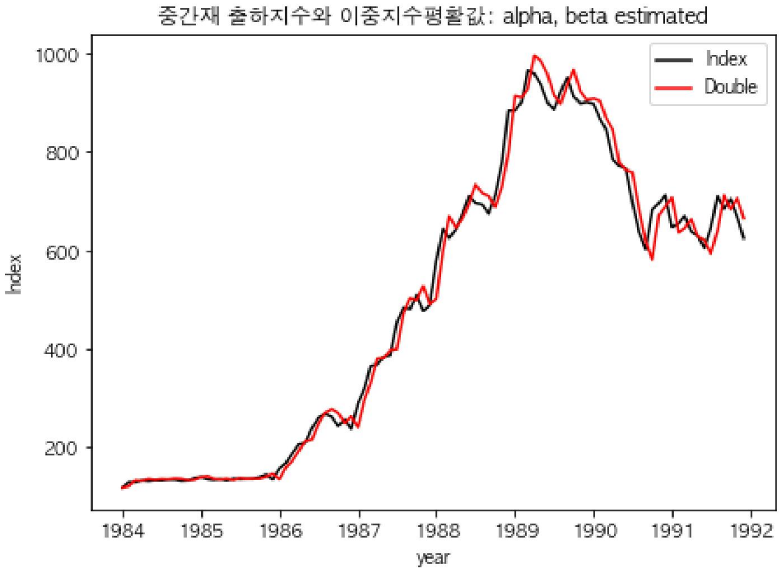


In [10]:

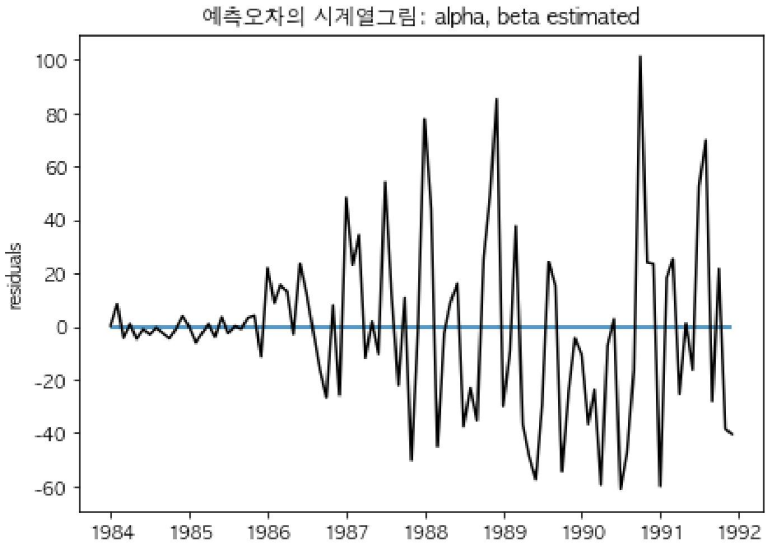
```
fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(results.resid, 'black')
ax.set_ylabel("residuals")
ax.set_title("그림 3-7 예측오차의 시계열 그림")
ax.hlines(0, min(data.index), max(data.index))
plt.show()
```



In [11]:
```python
fit5 = Holt(data, initialization_method="heuristic")
results = fit5.fit()

fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(data, 'black', label="Index")
ax.plot(results.fittedvalues, 'red', label="Double")
ax.set_xlabel("year")
ax.set_ylabel("Index")
ax.set_title("중간재 출하지수와 이중지수평활값: alpha, beta estimated")
plt.legend()
plt.show()
```

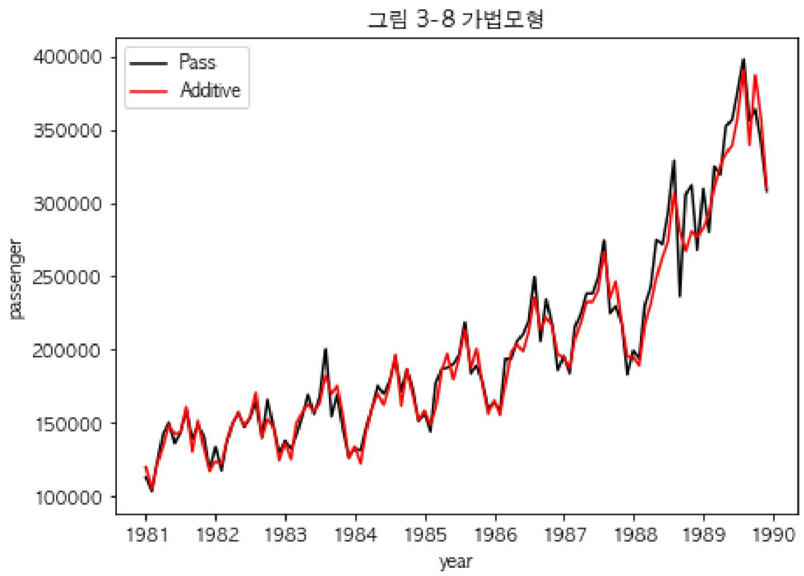


In [12]:
```python
fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(results.resid, 'black')
ax.set_ylabel("residuals")
ax.set_title("예측오차의 시계열그림: alpha, beta estimated")
ax.hlines(0, min(data.index), max(data.index))
plt.show()
```

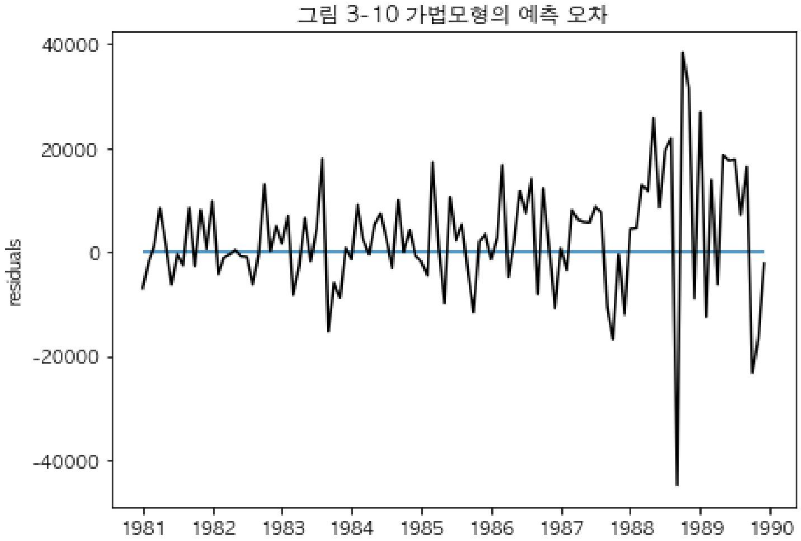


In [13]:
```python
z = []

with open('../data/koreapass.txt') as f:
    for line in f.readlines():
        for elem in line.rstrip().split(" "):
            if len(elem):
                z.append(float(elem))

index = pd.date_range(start="1981", periods=len(z), freq="MS")
data = pd.Series(z, index)

fit6 = ExponentialSmoothing(data,
                            initialization_method="heuristic",
                            seasonal="add")
results = fit6.fit()

fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(data, 'black', label="Pass")
ax.plot(results.fittedvalues, 'red', label="Additive")
ax.set_xlabel("year")
ax.set_ylabel("passenger")
ax.set_title("그림 3-8 가법모형")
plt.legend()
plt.show()
```

```
/Users/jonghyun/miniforge3/lib/python3.9/site-packages/statsmodels/tsa/holtwint
ers/model.py:920: ConvergenceWarning: Optimization failed to converge. Check ml
e_retvals.
  warnings.warn(
```

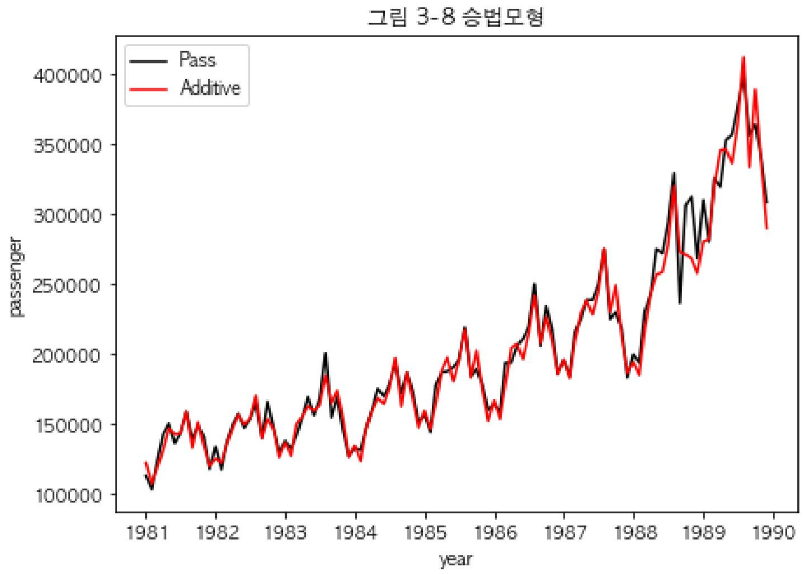


In [14]:
```
fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(results.resid, 'black')
ax.set_ylabel("residuals")
ax.set_title("그림 3-10 가법모형의 예측 오차")
ax.hlines(0, min(data.index), max(data.index))
plt.show()
```

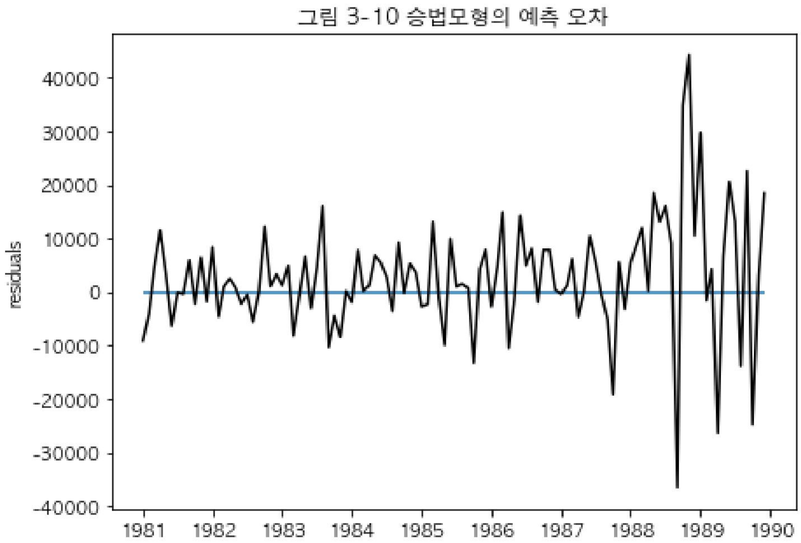


In [15]:
```
fit6 = ExponentialSmoothing(data,
                            initialization_method="heuristic",
                            seasonal="mul")
results = fit6.fit()

fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(data, 'black', label="Pass")
ax.plot(results.fittedvalues, 'red', label="Additive")
ax.set_xlabel("year")
ax.set_ylabel("passenger")
ax.set_title("그림 3-8 승법모형")
plt.legend()
plt.show()
```

```
/Users/jonghyun/miniforge3/lib/python3.9/site-packages/statsmodels/tsa/holtwint
ers/model.py:920: ConvergenceWarning: Optimization failed to converge. Check ml
e_retvals.
  warnings.warn(
```



In [16]:
```
fig, ax = plt.subplots(figsize=(7, 5))
ax.plot(results.resid, 'black')
ax.set_ylabel("residuals")
ax.set_title("그림 3-10 승법모형의 예측 오차")
ax.hlines(0, min(data.index), max(data.index))
plt.show()
```



In [ ]: