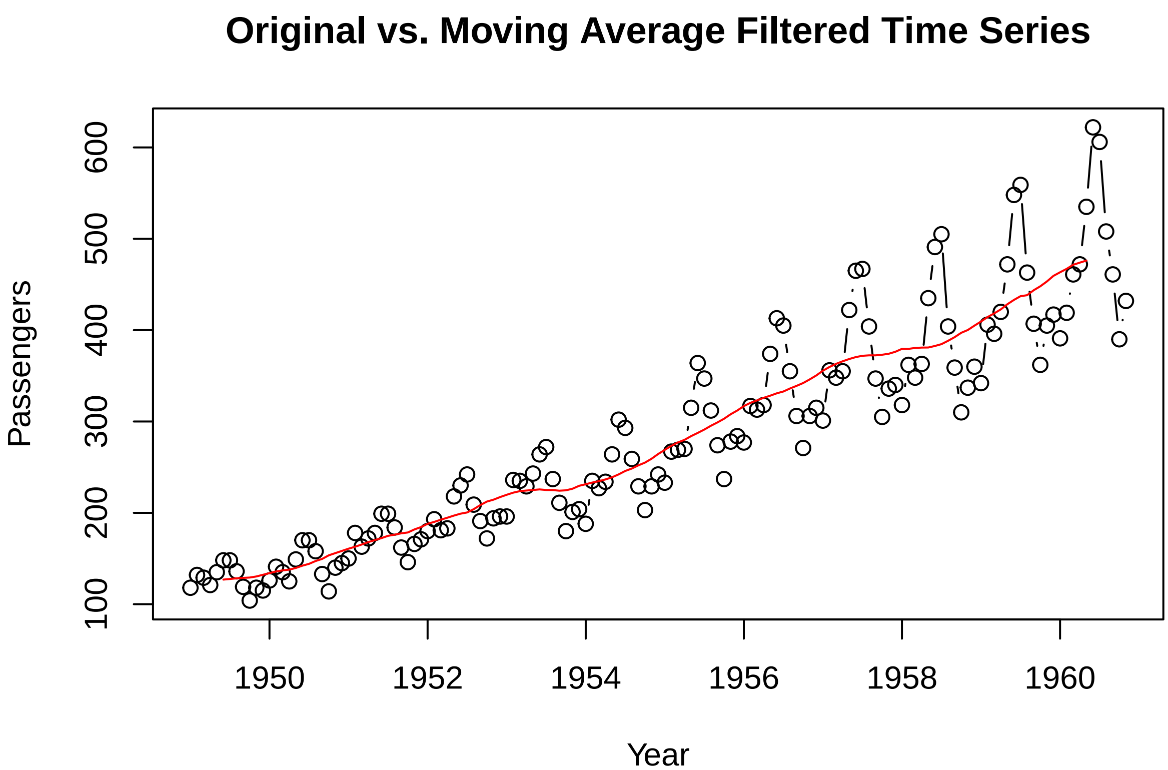
hw04

Samuel Swain

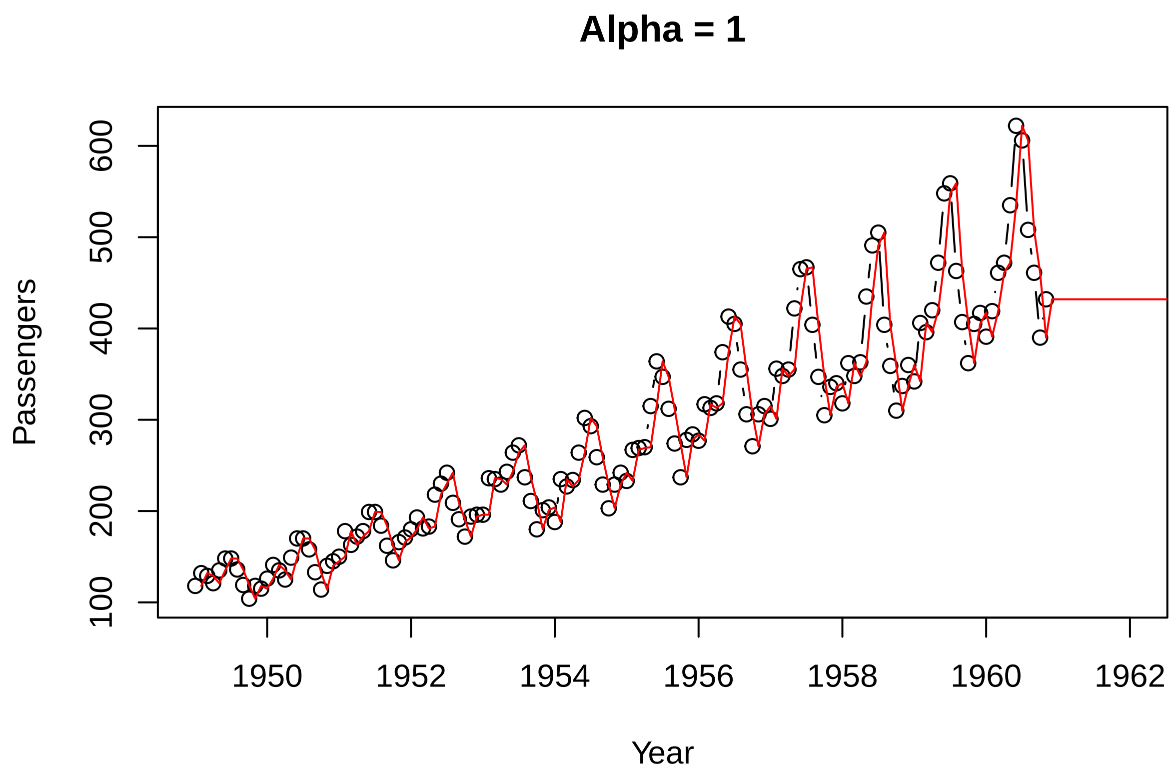
2023-03-10

Problem 1

1(a)



1(b)

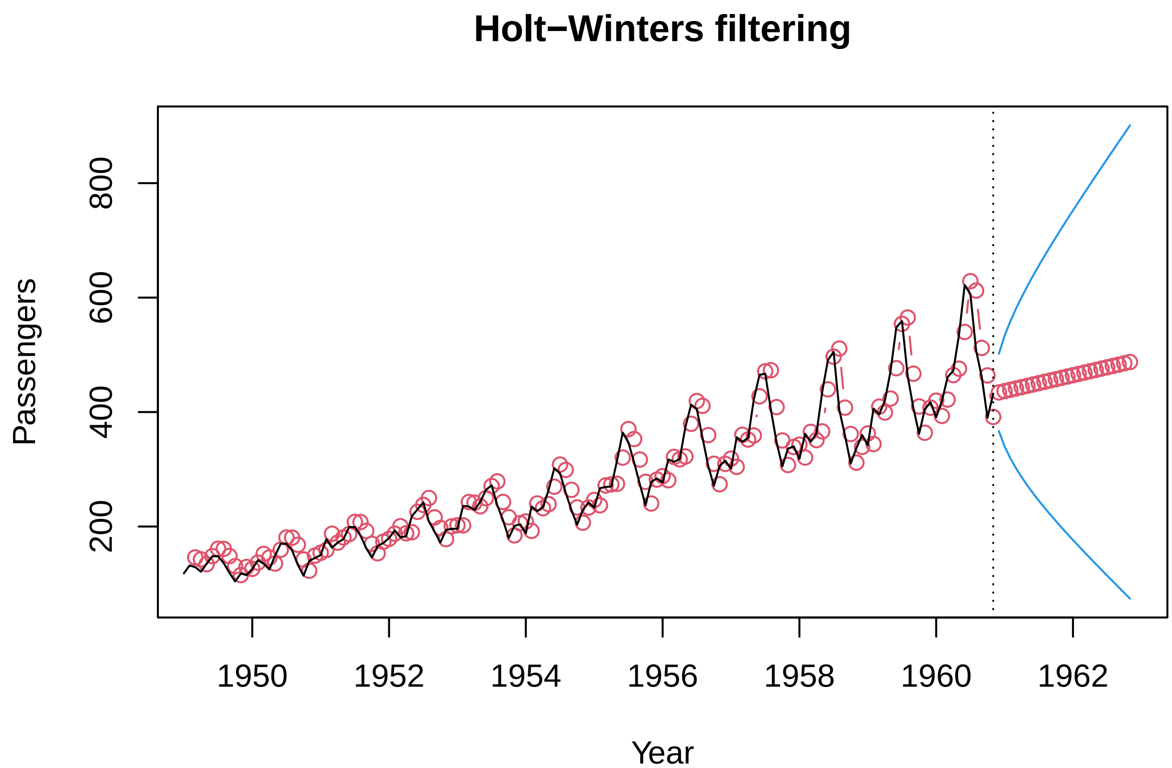


Holt Parameters:

- α: 1

As we can observe from the results, the optimal value for alpha is 1. This means that the forecast for the next two years is simply the value of the last observation in the data, as there is no seasonal component to this forecast. Given this, the model is performing the best it can in terms of accuracy.

1(c)



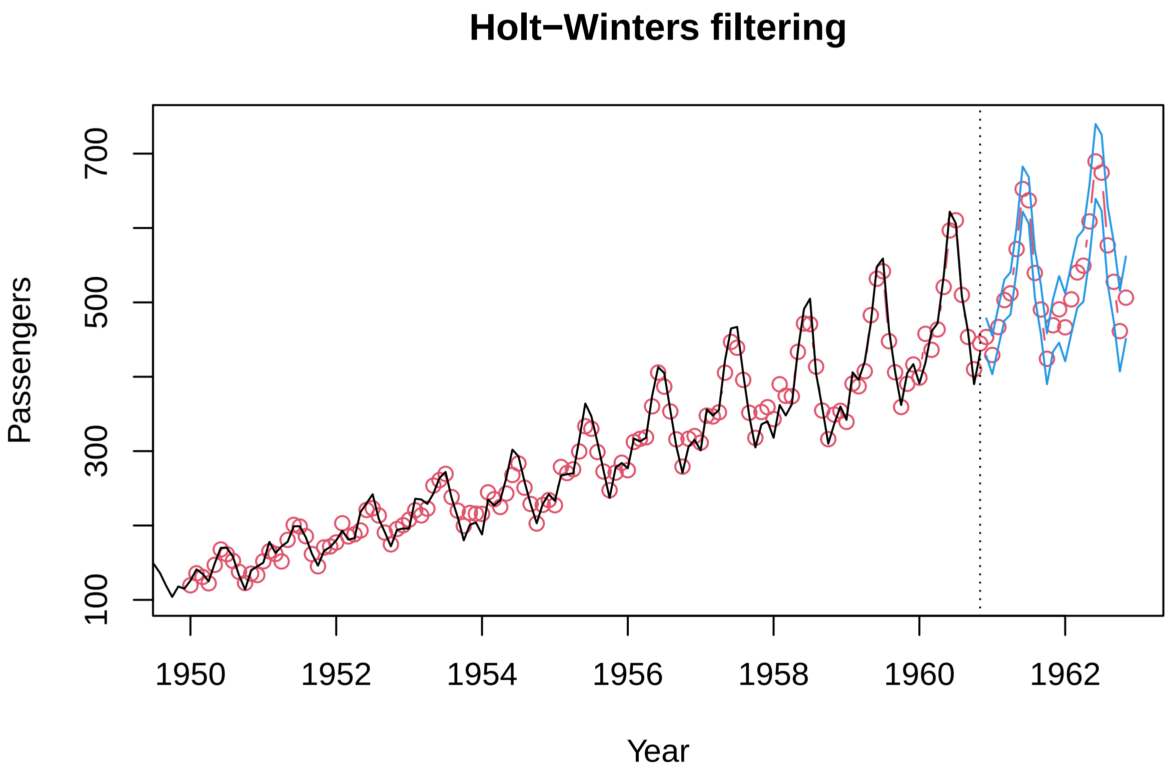

```
## Holt-Winters exponential smoothing with trend and without seasonal component.
##
## Call:
## HoltWinters(x = y_1a, gamma = FALSE, seasonal = "additive")
##
## Smoothing parameters:
##  alpha: 1
##  beta : 0.02095455
##  gamma: FALSE
##
## Coefficients:
##         [,1]
## a 432.000000
## b   2.312409
```

Holt Parameters:

- α: 1
- β: 0.0209545

Unlike the ewma prediction in 1b, the Holt Winters method above contains a trend component, β. In this problem, β is 0.0209545. This indicates that there is a slight positive trend that we can see illustrated in the outputted plot.

1(d)




```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## Call:
## HoltWinters(x = y_1a, seasonal = "additive")
##
## Smoothing parameters:
##  alpha: 0.247001
##  beta : 0.0323183
##  gamma: 1
##
## Coefficients:
##           [,1]
## a   477.924593
## b     3.100107
## s1  -27.599184
## s2  -54.835067
## s3  -20.352131
## s4   12.756244
## s5   18.715044
## s6   75.142139
## s7  152.758731
## s8  134.489015
## s9   33.664374
## s10 -18.472758
```

```
## s11 -87.869995
## s12 -45.924593
```

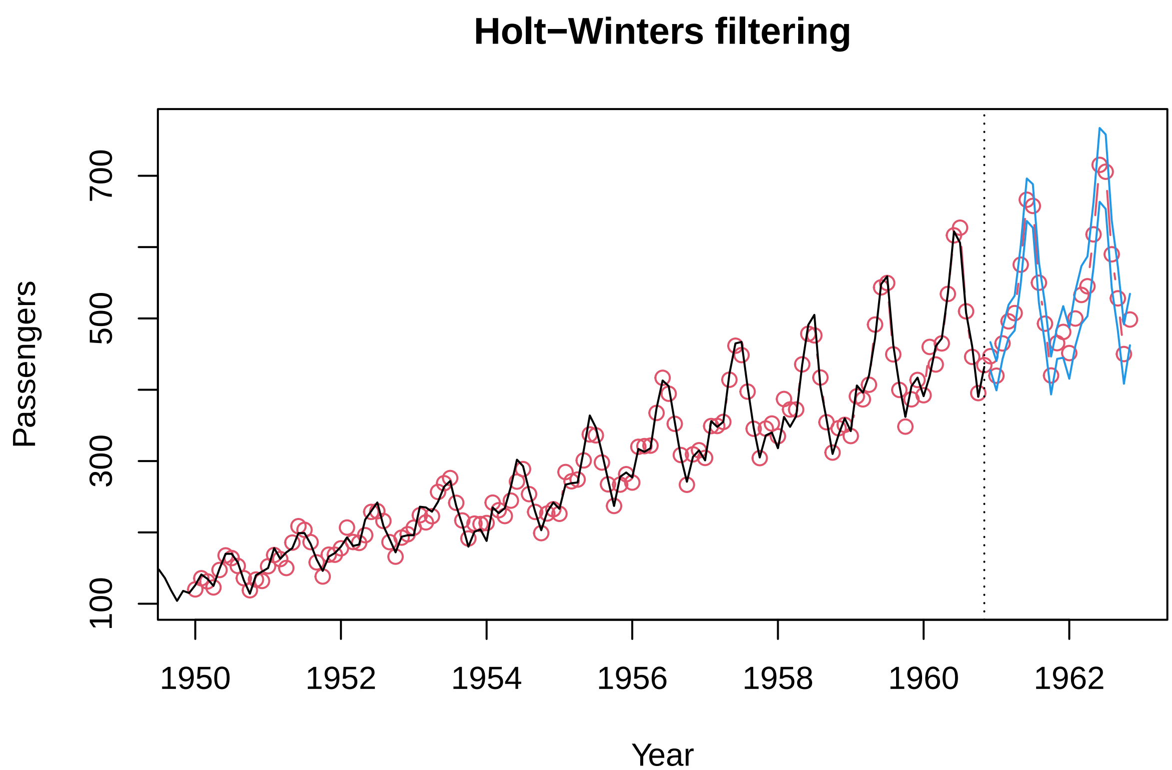
Holt Parameters:

- α: 0.247001
- β: 0.0323183
- γ: 1

This additive model build on the model from 1c by adding a seasonal component to it, γ. We can see above that γ is 1.Above is the s coefficients. These indicate the seasonality of each season. In this case, the Holt-Winters model estimated negative seasonal factors for the first three months (January to March), indicating that the observations tend to be lower than the overall trend and level during these months. Conversely, it estimated positive seasonal factors for the next three months (April to June), indicating that the observations tend to be higher than the overall trend and level during these months.

The model then estimated larger positive seasonal factors for the next four months (July to October), indicating that the observations have an even stronger tendency to be higher than the overall trend and level during this period. The model then estimated negative seasonal factors for November and December, indicating that the observations tend to be lower than the overall trend and level during these months.

1(e)




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

```
##
## Call:
## HoltWinters(x = y_1a, seasonal = "multiplicative")
##
## Smoothing parameters:
##  alpha: 0.2736169
##  beta : 0.03054619
##  gamma: 0.8646593
##
## Coefficients:
##            [,1]
## a   470.2666025
## b     3.0017724
## s1    0.9445527
## s2    0.8812072
## s3    0.9701226
## s4    1.0282494
## s5    1.0456516
## s6    1.1783932
## s7    1.3565012
## s8    1.3307315
## s9    1.1061253
## s10   0.9847913
## s11   0.8344900
## s12   0.9191454
```

Holt Parameters:

- α: 0.2736169
- β: 0.0305462
- γ: 0.8646593

The interpretation of this model is similar to 1d, the only difference being in the s coefficients are multiplicative and not additive. In an additive seasonal model, the seasonal component is assumed to be constant over time, and is added to the trend and level components of the model. This means that the magnitude of the seasonal fluctuations remains constant as the level of the time series changes. For example, if the average sales for a product are increasing over time, the additive seasonal model would assume that the seasonal fluctuations around the average also remain constant, regardless of the increasing trend.

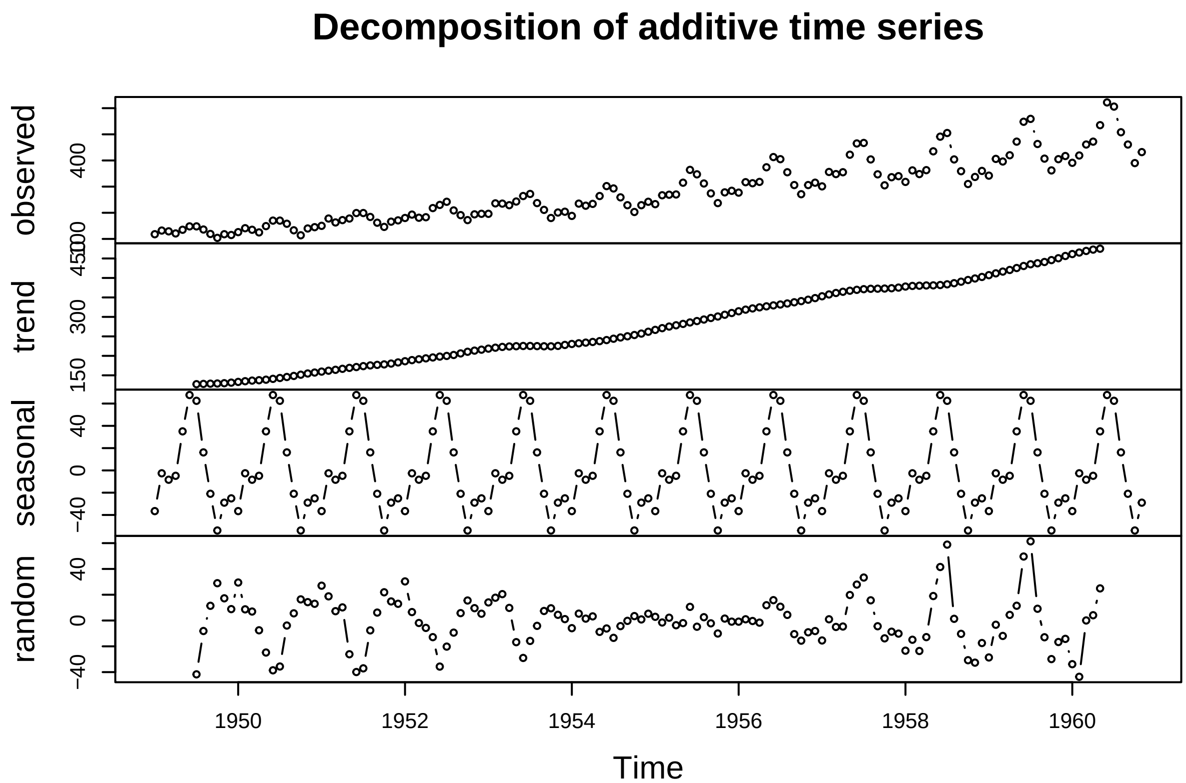
On the other hand, in a multiplicative seasonal model, the seasonal component is assumed to be proportional to the trend and level components of the model. This means that the magnitude of the seasonal fluctuations increases or decreases as the level of the time series changes. For example, if the average sales for a product are increasing over time, the multiplicative seasonal model would assume that the seasonal fluctuations around the average also increase proportionally with the increasing trend.

1(f)

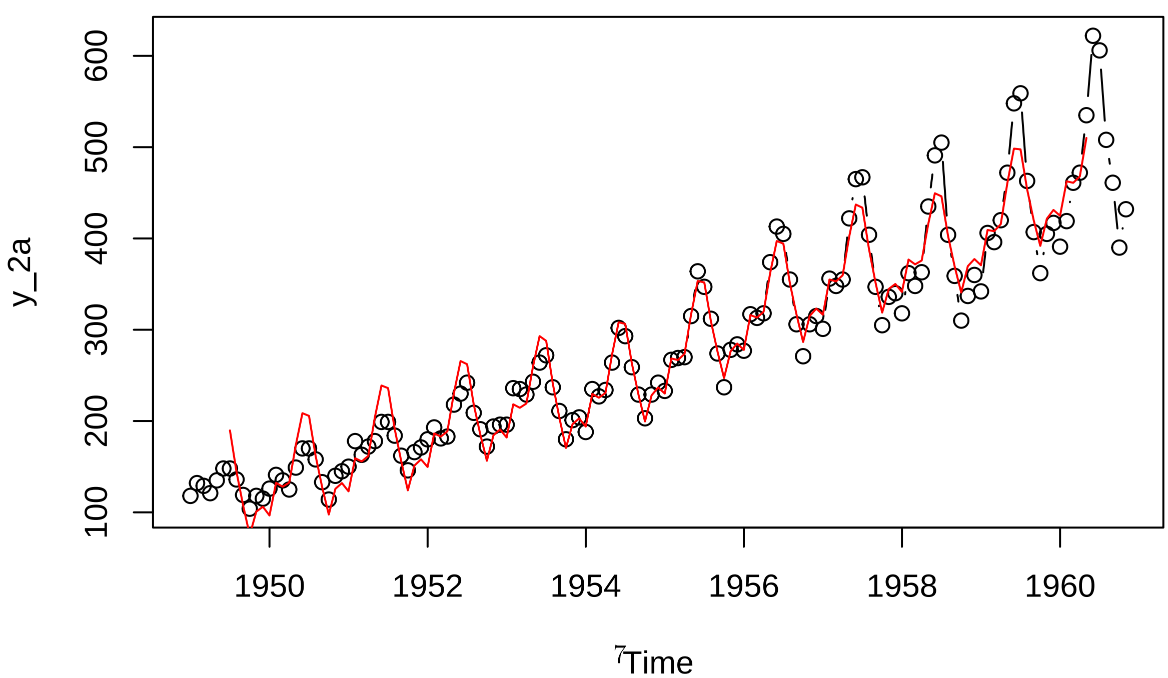
Above, we tried many different forecasting methods. These included an EWMA model along with different Holt Winters models. These included additive models with α, α and β, and α, β, and γ. After this we tried a multiplicative model with α, β, and γ. The multiplicative model seemed to perform the best. As time goes on, the data seems to vary more. This would indicate that a multiplicative model is going to be the best fit for the data.

Problem 2

2(a)



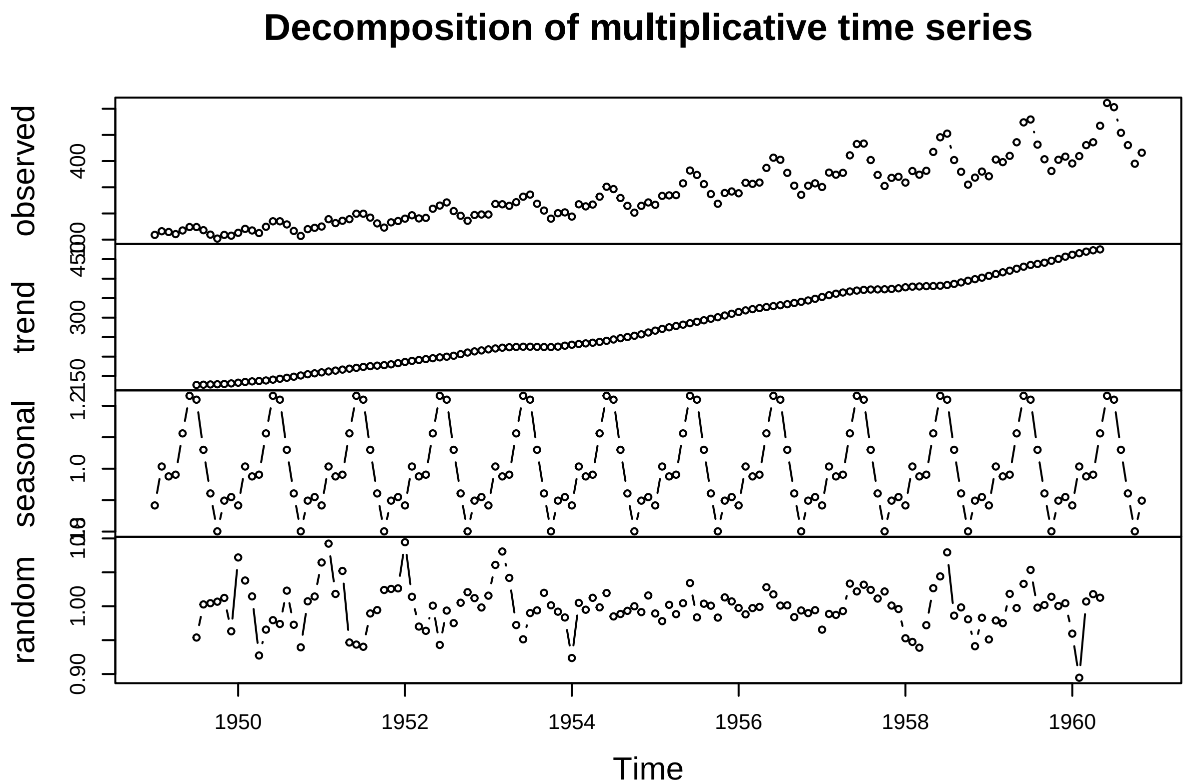
Decomposition of additive time series
observed
trend
seasonal
random
100
400
150
300
450
-40
0
40
-40
0
40
1950
1952
1954
1956
1958
1960
Time



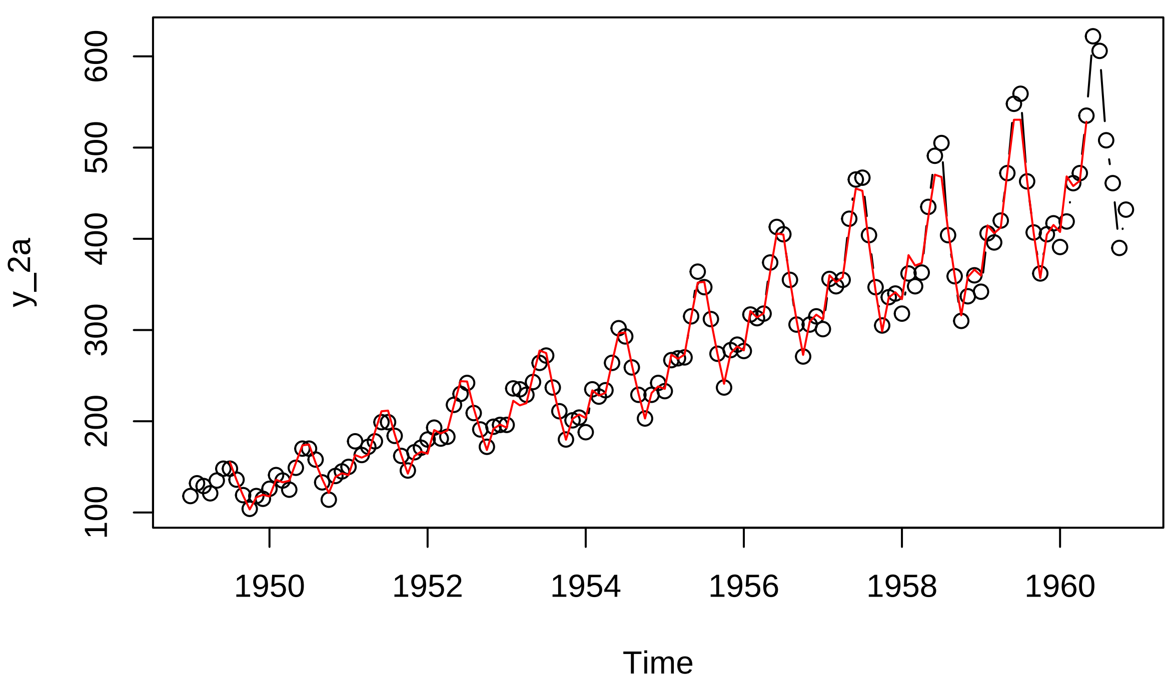
y_2a
100
200
300
400
500
600
1950
1952
1954
1956
1958
1960
Time

It is evident that the model is unable to fully explain certain aspects of the trend, as it significantly overestimates the values at the start and underestimates them towards the end. Incorporating a multiplicative feature is the missing piece of the puzzle, and the multiplicative Holt Winters model provides a solution for this.

2(b)



Decomposition of multiplicative time series
observed
trend
seasonal
random
Time



y_2a
Time

As shown above, the multiplicative model provides a much better fit for the trend. While it shares similarities with the additive model, the crucial difference lies in its approach to the trend. Unlike the additive model, the multiplicative model does not make the assumption of a linear continuation of the trend, which is why it is better equipped to fit the model.

Appendix

```
# Libraries
library(readxl)
```

Problem 1

```
# read in the data
df_1 <- read_excel("HW4_data.xls")
```

1(a)

```
# Convert the first column of `df_1` into a time series with frequency of
# 12 and start date of Jan 1949
y_1a <- ts(df_1[[1]], frequency = 12, start = c(1949, 1))

# Define variables for the moving average filter
m <- 12
w <- rep(1/m, m)

# Apply the moving average filter using the convolution method
filter_1a <- filter(y_1a, filter = w, method = "convolution", sides = 2)

# Plot the original time series and the filtered time series
plot(y_1a, type = "b", main = "Original vs. Moving Average Filtered Time Series",
     ylab="Passengers", xlab="Year")
lines(filter_1a, col = "red")
```

1(b)

```
# Set values for ewma analysis
n <- length(y_1a)
k <- 24

# Create empty vectors to store results
alpha_vec <- seq(0.01, 1, 0.01)
avg_error <- numeric(length(alpha_vec))

for (i in 1:length(alpha_vec)) {
  # Generate forecast for current alpha value
```

```r
  alpha <- alpha_vec[i]
  ewma <- filter(alpha * y_1a, filter = 1 - alpha, method = "recursive",
                 sides = 2, init = y_1a[1])
  y_pred <- c(NA, ewma, rep(ewma[n], k - 1))

  # Calculate error between forecast and actual values
  y_actual <- y_1a[(n-k+2):n]
  error <- y_actual - y_pred[(n-k+2):n]

  # Calculate average error and store in results vector
  avg_error[i] <- mean(abs(error))
}

# Print results
results <- data.frame(alpha = alpha_vec, avg_error = avg_error)
best_alpha <- results[which.min(results$avg_error),1]

ewma <- filter(best_alpha * y_1a, filter = 1 - best_alpha, method = "recursive",
               sides = 2, init = y_1a[1])
y_pred <- c(NA, ewma, rep(ewma[n], k - 1))
y_pred <- ts(y_pred, frequency = 12, start = c(1949, 1))
plot(y_1a, type = "b", xlim = c(1949, 1962), xlab="Year", ylab="Passengers")
lines(y_pred, col = "red")
title(paste("Alpha =", best_alpha))
```

1(c)

```r
holt_method_1c <-
  HoltWinters(y_1a, seasonal = "additive", gamma = FALSE)
holt_method_pred_1c <-
  predict(
    holt_method_1c,
    n.ahead = k,
    prediction.interval = T,
    level = 0.95
  )
plot(
  holt_method_1c,
  holt_method_pred_1c,
  type = "b",
  ylab = "Passengers",
  xlab = "Year"
)
holt_method_1c
```

1(d)

```r
holt_method_1d <- HoltWinters(y_1a, seasonal = "additive")
holt_method_pred_1d <-
```

```r
  predict(
    holt_method_1d,
    n.ahead = k,
    prediction.interval = T,
    level = 0.95
  )
plot(
  holt_method_1d,
  holt_method_pred_1d,
  type = "b",
  ylab = "Passengers",
  xlab = "Year"
)
holt_method_1d
```

1(e)

```r
holt_method_1e <- HoltWinters(y_1a, seasonal = "multiplicative")
holt_method_pred_1e <-
  predict(
    holt_method_1e,
    n.ahead = k,
    prediction.interval = T,
    level = 0.95
  )
plot(
  holt_method_1e,
  holt_method_pred_1e,
  type = "b",
  ylab = "Passengers",
  xlab = "Year"
)
holt_method_1e
```

Problem 2

2(a)

```r
y_2a <- ts(df_1[[1]], frequency = 12, start = c(1949, 1))
decomposition <- decompose(y_2a, type = "additive")

plot(decomposition, type = "b")

y_hat_2a <- decomposition$trend + decomposition$seasonal
plot(y_2a, type="b")
lines(y_hat_2a, col="red")
```

2(b)

```
y_2a <- ts(df_1[[1]], frequency = 12, start = c(1949, 1))
decomposition <- decompose(y_2a, type = "multiplicative")
plot(decomposition, type = "b")

y_hat_2b <- decomposition$trend * decomposition$seasonal
plot(y_2a, type="b")
lines(y_hat_2b, col="red")
```