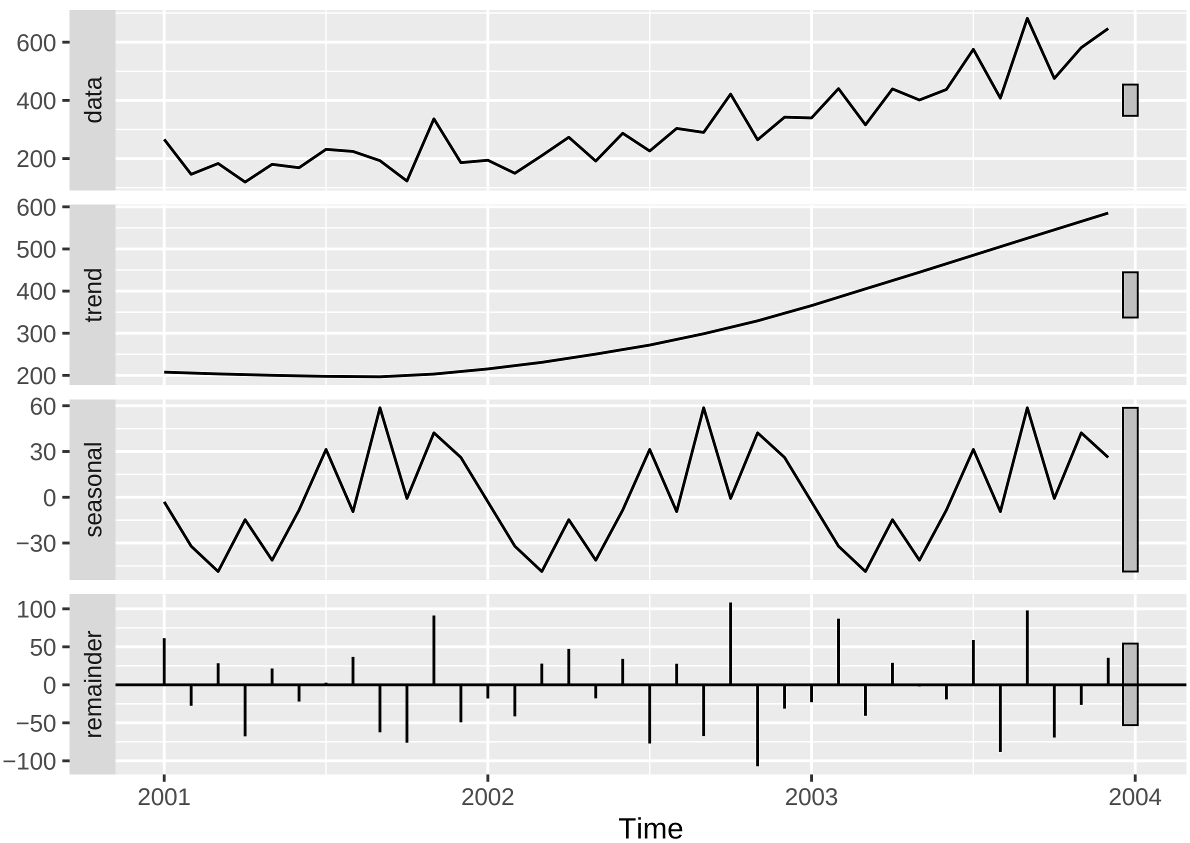
An Application of Holt-Winters Model: Shampoo

Logistics and Supply Chain Analytics

Jiahua Wu

In this example, we are going to use the same shampoo data set to illustrate how to fit Holt-Winters models in R.

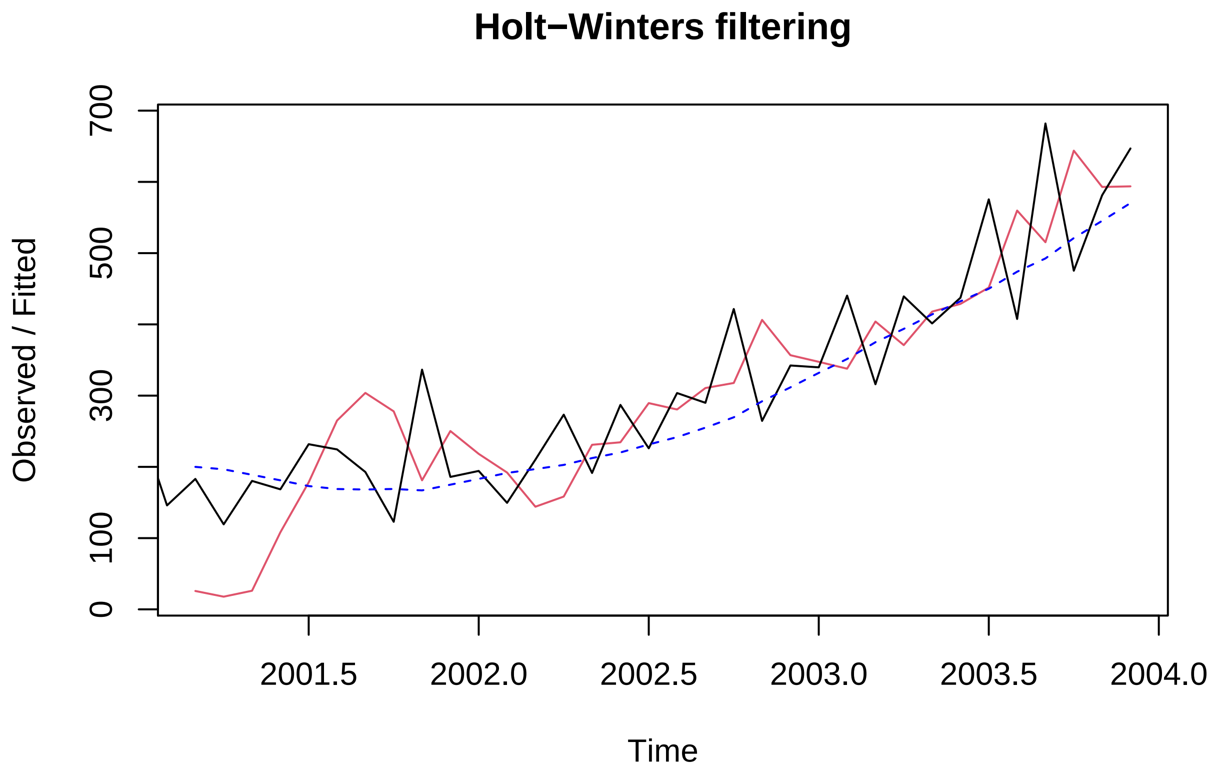
```
data <- read.csv(file = "shampoo.csv", header = TRUE)
shampoo <- ts(data[, 2], frequency = 12, start = c(2001, 1))
# autoplot(shampoo)
shampoo %>% stl(s.window = "period") %>% autoplot
```



For time series analysis, the first step is always to visually inspect the time series. In this regard, the *stl*() function is quite useful. It decomposes the original time series into trend, seasonal factors, and random error terms. The relative importance of different components are indicated by the grey bars in the plots. For this data set, the grey bar of the trend panel is only slightly larger than that on the orginal time series panel, which indicates that trend component contributes to a great proportion of variations in the time series. On the other hand, the grey bar of the seasonal panel is very large, even larger than the grey bar of random error term, which indicates that the contribution of seasonal factors to the variation in the original time series is marginal. In other words, it indicates that there is no seasonality in the data.

```r
### Model estimation
# using HoltWinters
shampoo.HW <- HoltWinters(shampoo, gamma = FALSE)

# HW is sensitive to starting values
shampoo.HW2 <- HoltWinters(shampoo, gamma = FALSE, optim.start = c(alpha = 0, beta = 0),
                           l.start = 200, b.start = -2.6)
plot(shampoo.HW)
lines(fitted(shampoo.HW2)[,2], col = "blue", lty = 2)
```




```r
# in-sample one-step forecast
sqrt(shampoo.HW$SSE/(length(shampoo)-2))
```

```
## [1] 95.95459
```

```r
sqrt(shampoo.HW2$SSE/(length(shampoo)-2))
```

```
## [1] 67.11598
```

We first fit the data with $HoltWinters()$ function. As we observe no seasonality in the data, we specify $gamma = F$ in the function. Fitting the model with $HoltWinters()$ is straightforward, but one thing we need to keep in mind is that, as any optimization problem, results from $HoltWinters()$ are sensitive to starting values. We estimate two models - one with the default starting values and one with manually supplied values - and we notice that the models differ significantly. The root of mean squared errors reduces by around one-third after changing the initial values. In practice, it is generally hard to guess the "correct" starting points, but luckily this problem is solved by $ets()$ to some extent.

With $ets()$, initial states and smoothing parameters are jointly estimated by maximizing the likelihood function. We need to specify the model in $ets()$ using three letters. The way to approach this is: (1) check out time series plot, and see if there is any trend and seasonality, and whether they are additive (linear trend, constant variations in seasonal factors) or multiplicative; (2) run $ets()$ with model = "ZZZ", and see whether the best model is consistent with your expectation; (3) if they are consistent, it gives us confidence that our model specification is correct; Otherwise try to figure out the reason for the discrepancy.

```
# using ets
shampoo.ets <- ets(shampoo, model = "AAN")
shampoo.ets2 <- ets(shampoo, model = "ZZZ")
shampoo.ets2
```

```
## ETS(A,A,N)
##
## Call:
##  ets(y = shampoo, model = "ZZZ")
##
##   Smoothing parameters:
##     alpha = 0.0563
##     beta  = 0.0563
##
##   Initial states:
##     l = 176.6845
##     b = -1.237
##
##   sigma:  70.3984
##
##      AIC     AICc      BIC
## 441.0668 443.0668 448.9844
```

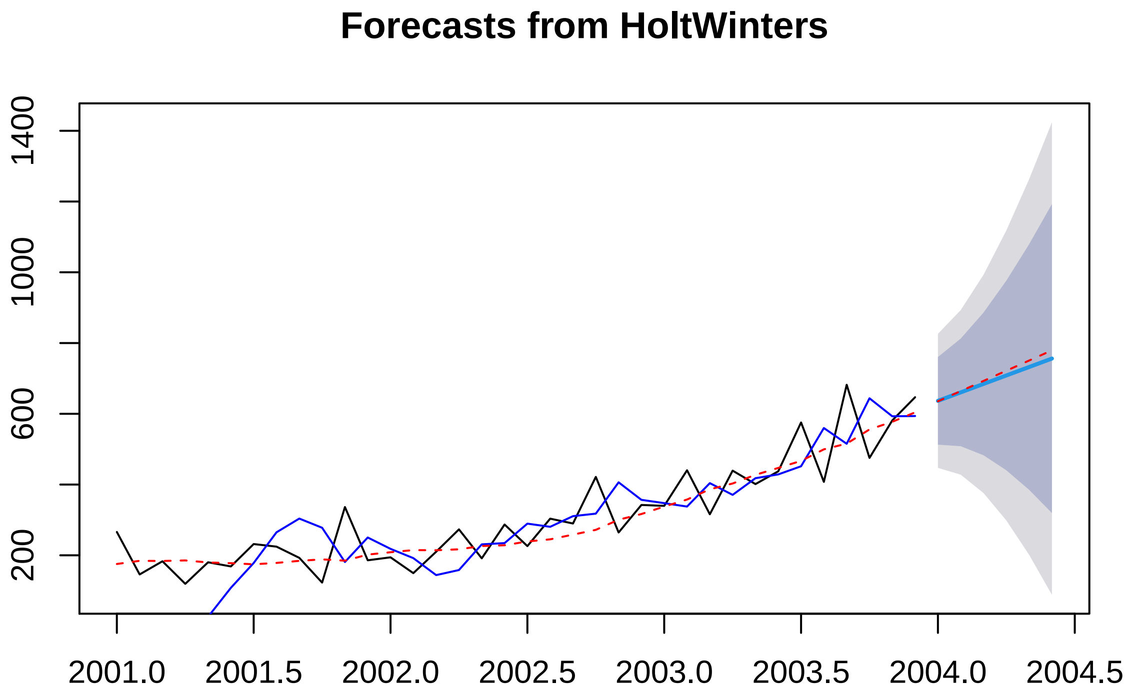
After estimation, we can use $accuracy()$ function to determine in-sample fit and $forecast()$ function to generate forecast. We plot forecasts from $HoltWinters()$ and $ets()$ on the same plot, and notice that, even though the in-sample fit from the two functions differ significantly, the forecasting results are more or less comparable.

```
# in-sample one-step forecast
accuracy(shampoo.ets)
```

```
##                    ME     RMSE     MAE       MPE    MAPE      MASE       ACF1
## Training set 14.84085 66.37224 50.5648 0.2848111 17.5656 0.3294034 -0.4839511
```

```
# out-of-sample forecast
shampoo.HW.f <- forecast(shampoo.HW, h = 6)
shampoo.ets.f <- forecast(shampoo.ets, h = 6)

plot(shampoo.HW.f)
lines(fitted(shampoo.HW.f), col = "blue")
lines(fitted(shampoo.ets), col = "red", lty = 2)
lines(shampoo.ets.f$mean, col = "red", lty = 2)
```



The above is an illustration of $HoltWinters()$ and $ets()$ functions. In practice, we need to first split samples into training and test sets, and then evaluate models by comparing out-of-sample performance. Suppose we use the first 2.5-year of data for training, and then evaluate the performance of models from $HoltWinters()$ and $ets()$. The results show that the model from $ets()$ is consistently better across all different measures.

```
### model evaluation
# Fit model with first 2.5 years of data
m1 <- HoltWinters(window(shampoo, end = c(2003, 6)), gamma = FALSE)
m2 <- ets(window(shampoo, end = c(2003, 6)), model = "AAN")

# out-of-sample multi-step forecasts
accuracy(forecast(m1, h = 6), window(shampoo, start = c(2003, 7)))
```

```
##                    ME      RMSE      MAE      MPE     MAPE      MASE       ACF1
## Training set 12.56592  87.02346 72.12966 2.647523 32.75038 0.6001358 -0.1434291
## Test set     62.15626 108.21178 94.39078 8.636494 16.15520 0.7853536 -0.7637161
##              Theil's U
## Training set        NA
## Test set     0.6349905
```

```
accuracy(forecast(m2, h = 6), window(shampoo, start = c(2003, 7)))
```

```
##                    ME      RMSE      MAE       MPE     MAPE      MASE
## Training set 12.07206  58.35153 43.79584 0.5196929 17.89491 0.3643918
## Test set     54.47860 104.04027 92.37836 7.1909678 16.01970 0.7686098
##                    ACF1 Theil's U
## Training set -0.3082067        NA
## Test set     -0.7591104 0.6137215
```