

# Data624 - Homework4

Amit Kapoor

2/28/2021

## Contents

<b>Exercise 7.1</b>	<b>1</b>
a) . . . . .	1
b) . . . . .	3
<b>Exercise 7.5</b>	<b>3</b>
a) . . . . .	4
b) . . . . .	4
c) . . . . .	5
<b>Exercise 7.6</b>	<b>6</b>
a) . . . . .	6
b) . . . . .	6
c) . . . . .	7
d) . . . . .	8
<b>Exercise 7.7</b>	<b>9</b>
<b>Exercise 7.8</b>	<b>9</b>
a) . . . . .	9
b) . . . . .	9
c) . . . . .	9
d) . . . . .	9
e) . . . . .	10
<b>Exercise 7.9</b>	<b>10</b>

## Exercise 7.1

Consider the `pigs` series — the number of pigs slaughtered in Victoria each month.

```
str(pigs)
```

```
## Time-Series [1:188] from 1980 to 1996: 76378 71947 33873 96428 105084 ...
```

a)

Use the `ses()` function in R to find the optimal values of  $\alpha$  and  $\ell_0$ , and generate forecasts for the next four months.

```
# Using ses for pigs
pigs_ses <- ses(pigs, h=4)
```

```

#summary
summary(pigs_ses)

##
## Forecast method: Simple exponential smoothing
##
## Model Information:
## Simple exponential smoothing
##
## Call:
## ses(y = pigs, h = 4)
##
## Smoothing parameters:
##   alpha = 0.2971
##
## Initial states:
##   l = 77260.0561
##
## sigma: 10308.58
##
##      AIC      AICc      BIC
## 4462.955 4463.086 4472.665
##
## Error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
## Training set 385.8721 10253.6 7961.383 -0.922652 9.274016 0.7966249 0.01282239
##
## Forecasts:
##      Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## Sep 1995      98816.41 85605.43 112027.4 78611.97 119020.8
## Oct 1995      98816.41 85034.52 112598.3 77738.83 119894.0
## Nov 1995      98816.41 84486.34 113146.5 76900.46 120732.4
## Dec 1995      98816.41 83958.37 113674.4 76092.99 121539.8

```

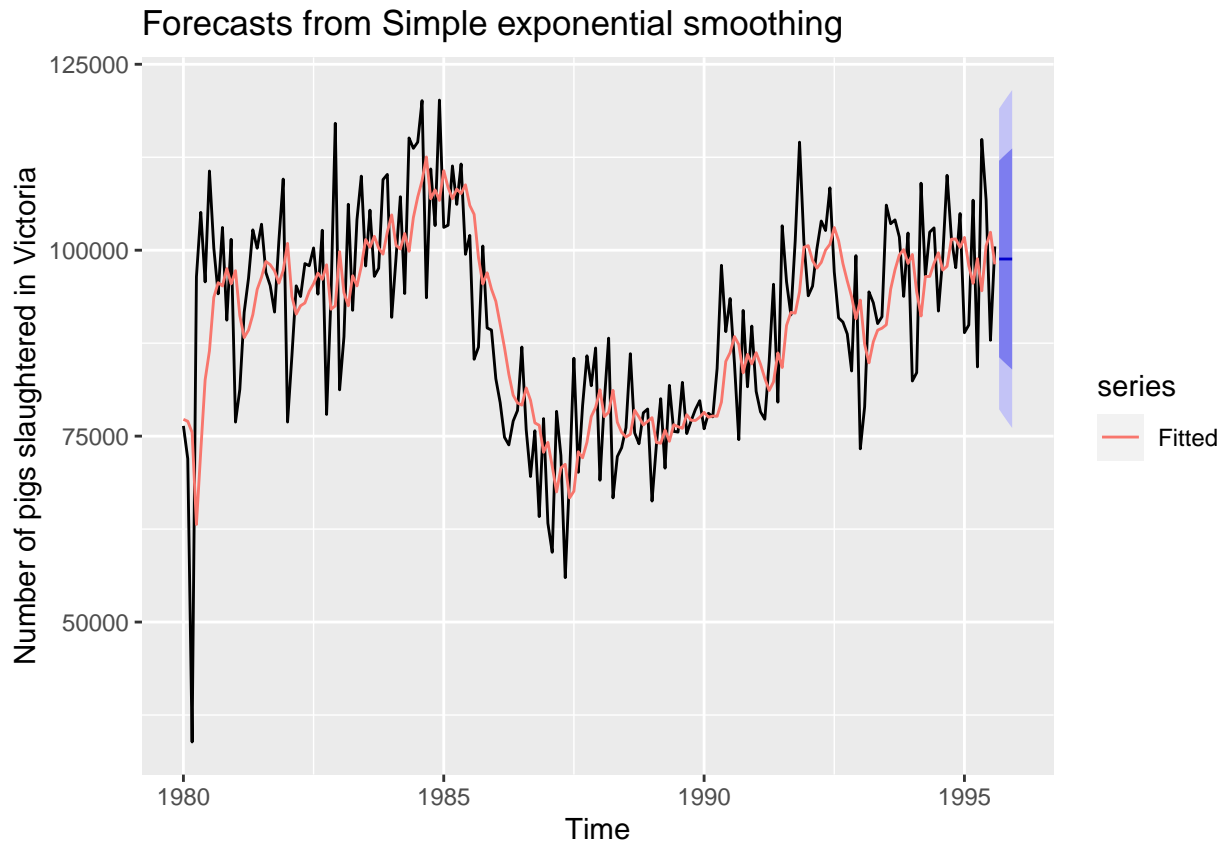
Above summary shows the the optimal values of  $\alpha$  and  $\ell_0$  are 0.2971 and 77260.0561 respectively. Using these values forecast is generated for next 4 months.

Next plot shows the forecast from simple exponential smoothing. Also one-step-ahead fitted values are plotted with the data over the period.

```

autoplot(pigs_ses) +
  autolayer(fitted(pigs_ses), series="Fitted") +
  ylab("Number of pigs slaughtered in Victoria")

```



b)

Compute a 95% prediction interval for the first forecast using  $\hat{y} \pm 1.96\sigma$  where  $\sigma$  is the standard deviation of the residuals. Compare your interval with the interval produced by R.

```
# 95% prediction interval for the first forecast
sd <- sd(residuals(pigs_ses))
ci95 <- c(lower = pigs_ses$mean[1] - 1.96*sd, upper = pigs_ses$mean[1] + 1.96*sd)
ci95
```

```
##      lower      upper
## 78679.97 118952.84
```

```
# By R
ci95_R <- c(pigs_ses$lower[1, "95%"], pigs_ses$upper[1, "95%"])
names(ci95_R) <- c("lower", "upper")
ci95_R
```

```
##      lower      upper
## 78611.97 119020.84
```

It appears the 95% prediction interval calculated by R is a little wider than the one given by the formula.

## Exercise 7.5

Data set `books` contains the daily sales of paperback and hardcover books at the same store. The task is to forecast the next four days' sales for paperback and hardcover books.

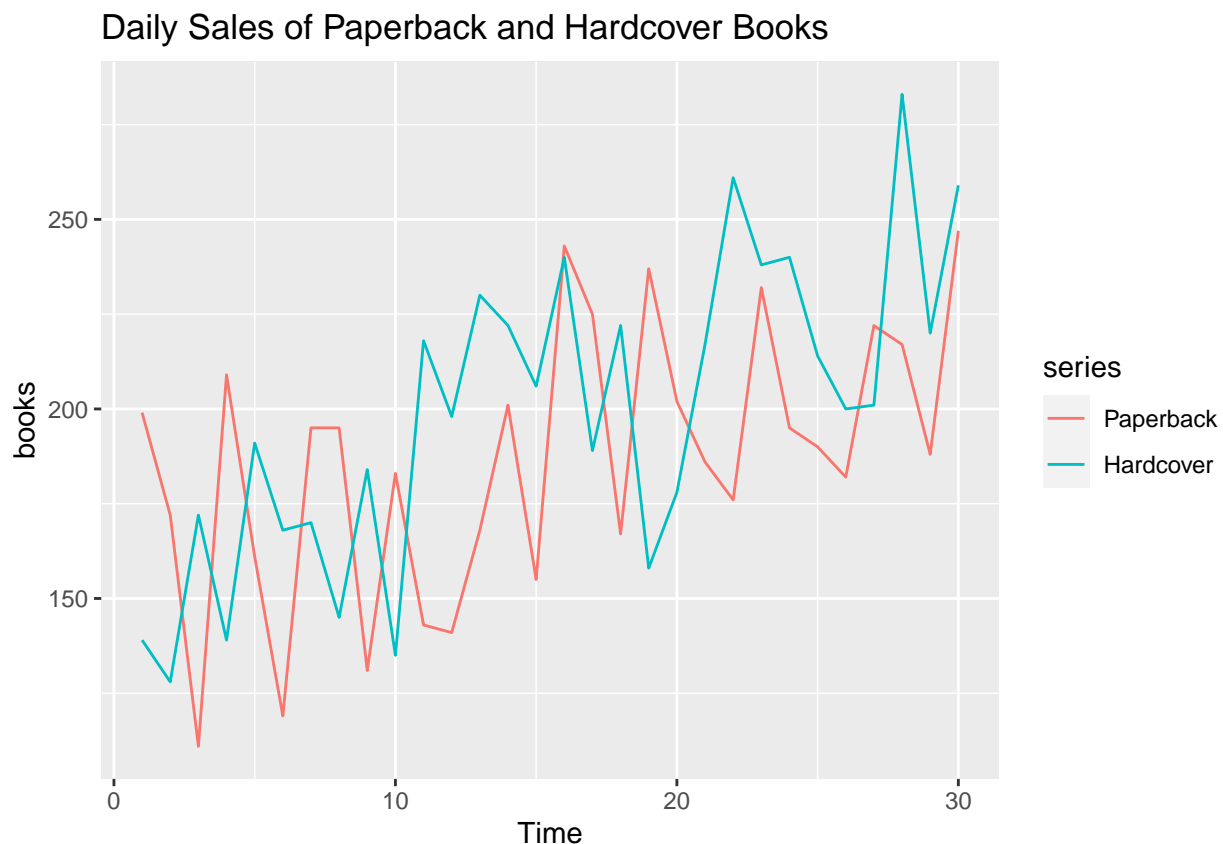
```
head(books)
```

```
## Time Series:  
## Start = 1  
## End = 6  
## Frequency = 1  
##   Paperback Hardcover  
## 1      199      139  
## 2      172      128  
## 3      111      172  
## 4      209      139  
## 5      161      191  
## 6      119      168
```

a)

Plot the series and discuss the main features of the data.

```
# plot series  
autoplot(books) +  
  labs(title = "Daily Sales of Paperback and Hardcover Books")
```



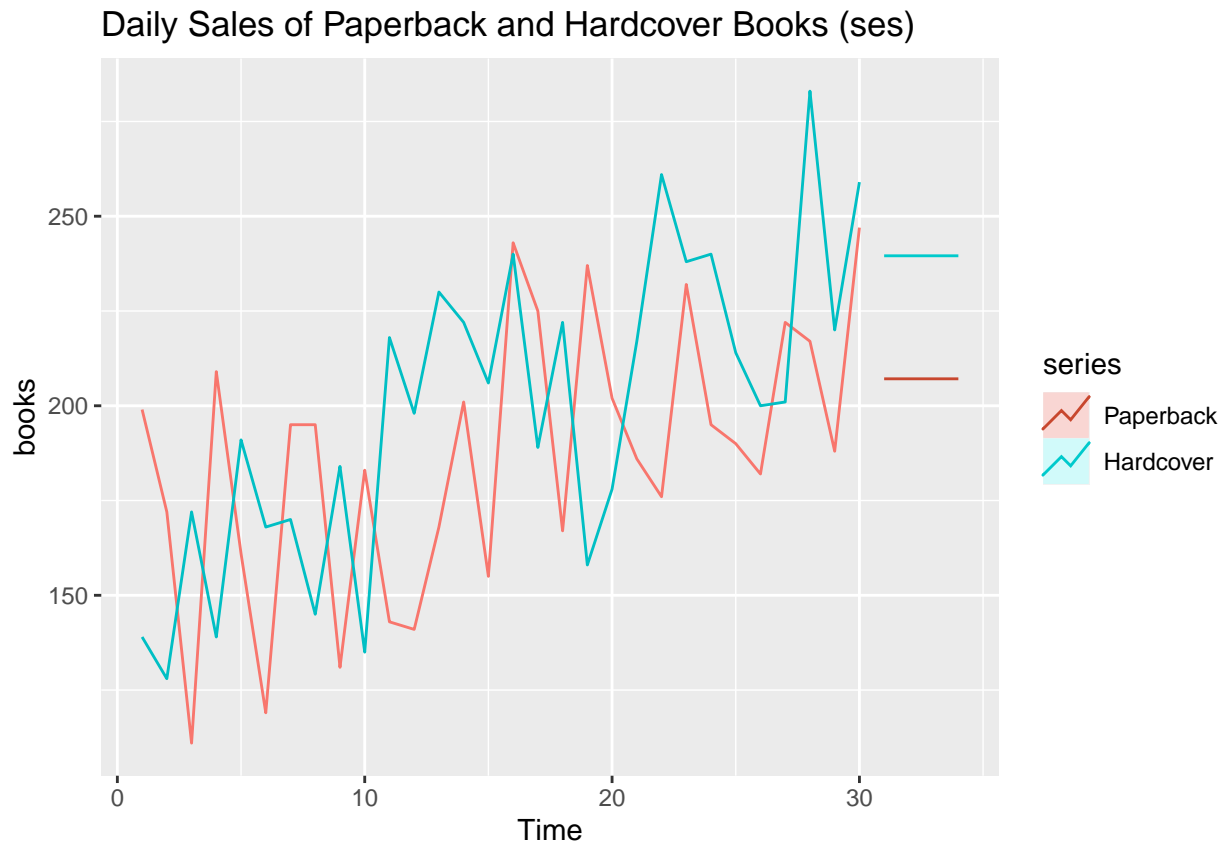
The series has an upward trend but don't see any seasonality or cyclicity in the plot. Also its only a 30 days of data so difficult to speak about seasonality. Another observation is hardcover sales in better than paperback.

b)

Use the `ses()` function to forecast each series, and plot the forecasts.

```
# Using ses for books
pb_ses <- ses(books[, 'Paperback'], h=4)
hc_ses <- ses(books[, 'Hardcover'], h=4)

autoplot(books) +
  autolayer(pb_ses, series="Paperback", PI=FALSE) +
  autolayer(hc_ses, series="Hardcover", PI=FALSE) +
  labs(title = "Daily Sales of Paperback and Hardcover Books (ses)")
```



The simple exponential smoothing plot above shows flat forecast and doesn't appear to capture upward trend.

c)

Compute the RMSE values for the training data in each case.

```
# RMSE for paperback
round(accuracy(pb_ses)[2], 2)
```

```
## [1] 33.64
```

```
# RMSE for hardcover
round(accuracy(hc_ses)[2], 2)
```

```
## [1] 31.93
```

RMSE of hardcover for the training data is slightly better than of paperback.

## Exercise 7.6

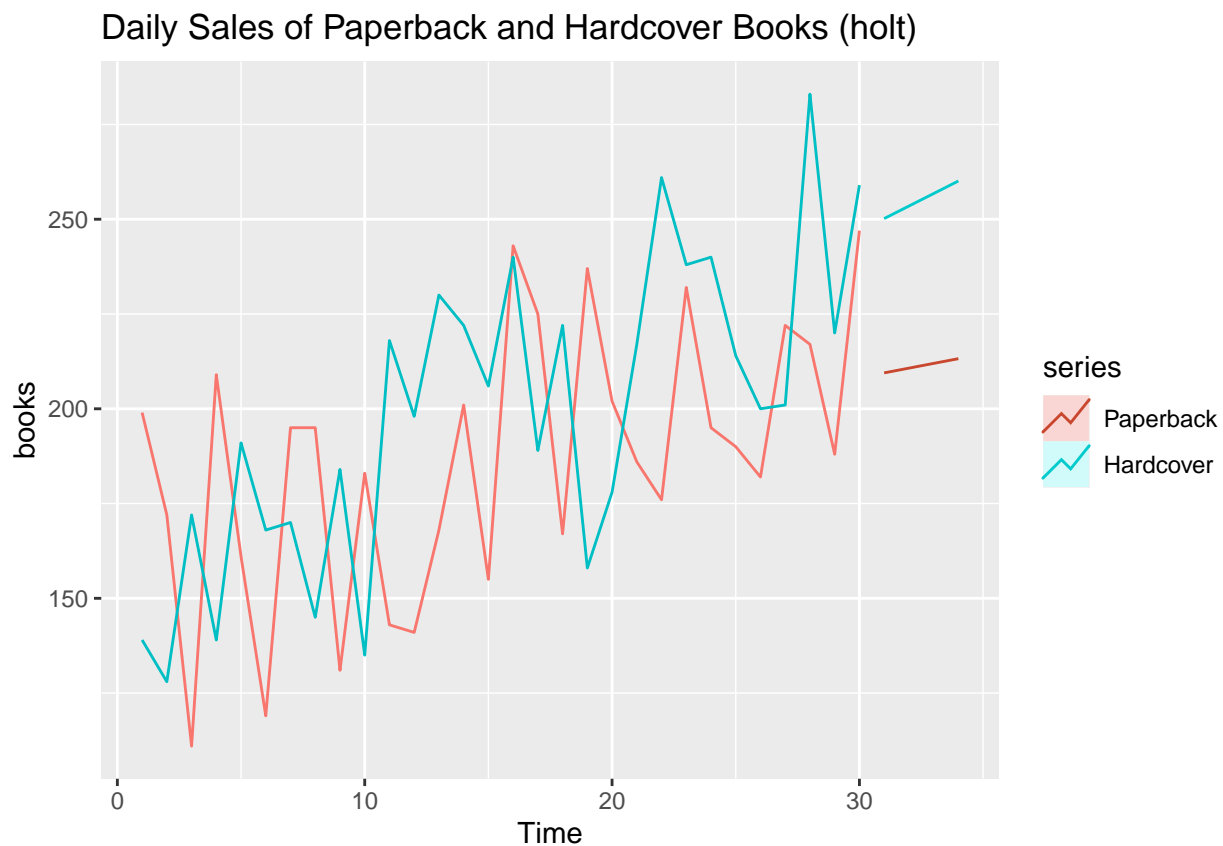
We will continue with the daily sales of paperback and hardcover books in data set books.

a)

Apply Holt's linear method to the `paperback` and `hardback` series and compute four-day forecasts in each case.

```
pb_holt <- holt(books[, 'Paperback'], h=4)
hc_holt <- holt(books[, 'Hardcover'], h=4)

autoplot(books) +
  autolayer(pb_holt, series="Paperback", PI=FALSE) +
  autolayer(hc_holt, series="Hardcover", PI=FALSE) +
  labs(title = "Daily Sales of Paperback and Hardcover Books (holt)")
```



Holt's linear forecast seems better as it is able to capture the upward trend of time series.

b)

Compare the RMSE measures of Holt's method for the two series to those of simple exponential smoothing in the previous question. (Remember that Holt's method is using one more parameter than SES.) Discuss the merits of the two forecasting methods for these data sets.

```
# RMSE for paperback - holt
round(accuracy(pb_holt)[2], 2)
```

```
## [1] 31.14
```

```
# RMSE for hardcover - holt
round(accuracy(hc_holt)[2], 2)
```

```
## [1] 27.19
```

The RMSEs for paperback and hardcover books sale are improved using holt's linear method as compared from simple exponential smoothing method. It happens since holt extended simple exponential smoothing to allow the forecasting of data with a trend and we see in above plot of holt's, capturing upward trend.

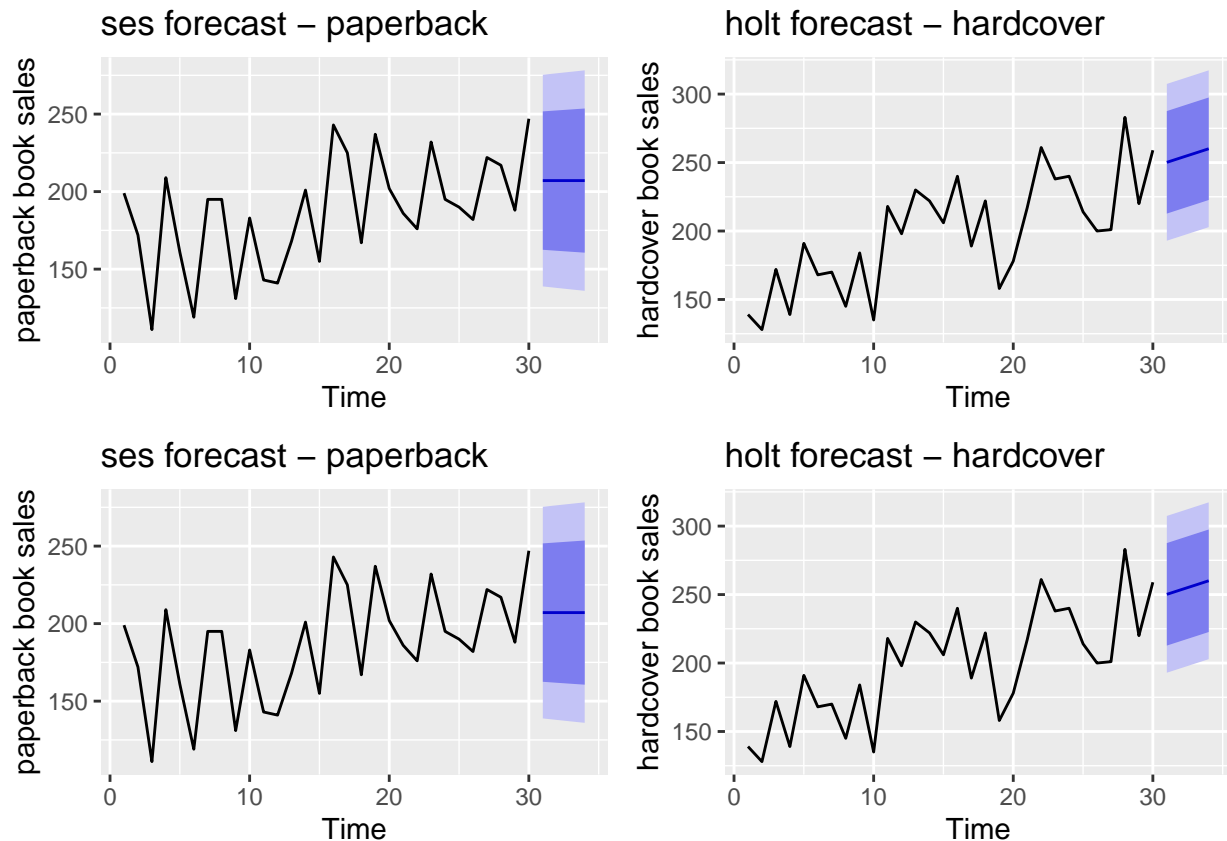
c)

Compare the forecasts for the two series using both methods. Which do you think is best?

```
# ses and holt comparison for paperback
s1 <- autoplot(pb_ses) +
  ylab("paperback book sales") +
  labs(title = "ses forecast - paperback")
h1 <- autoplot(pb_holt) +
  ylab("paperback book sales") +
  labs(title = "holt forecast - paperback")

# ses and holt comparison for hardcover
s2 <- autoplot(hc_ses) +
  ylab("hardcover book sales") +
  labs(title = "ses forecast - hardcover")
h2 <- autoplot(hc_holt) +
  ylab("hardcover book sales") +
  labs(title = "holt forecast - hardcover")

gridExtra::grid.arrange(s1, h2, s1,h2, nrow=2, ncol=2)
```



Compare the forecasts for the two series using the two, Holt's method is better than simple exponential smoothing method sith holt extended simple exponential smoothing to allow the forecasting of data with a trend. The RMSE is smaller for holt method.

d)

Calculate a 95% prediction interval for the first forecast for each series, using the RMSE values and assuming normal errors. Compare your intervals with those produced using `ses` and `holt`.

```
# 95% prediction interval for the first forecast
df <- data.frame(
  Pred_Int = c("Paperback-SES", "Paperback-Holt", "Hardcover-SES", "Hardcover-Holt"),
  lower = c(pb_ses$mean[1] - 1.96*accuracy(pb_ses)[2],
    pb_holt$mean[1] - 1.96*accuracy(pb_holt)[2],
    hc_ses$mean[1] - 1.96*accuracy(hc_ses)[2],
    hc_holt$mean[1] - 1.96*accuracy(hc_holt)[2]),
  upper = c(pb_ses$mean[1] + 1.96*accuracy(pb_ses)[2],
    pb_holt$mean[1] + 1.96*accuracy(pb_holt)[2],
    hc_ses$mean[1] + 1.96*accuracy(hc_ses)[2],
    hc_holt$mean[1] + 1.96*accuracy(hc_holt)[2])
)
df
```

```
##      Pred_Int  lower  upper
## 1 Paperback-SES 141.1798 273.0395
## 2 Paperback-Holt 148.4384 270.4951
## 3 Hardcover-SES 176.9753 302.1449
```



```
## 4 Hardcover-Holt 196.8745 303.4733
df2 <- data.frame(
  Pred_Int = c("Paperback-SES", "Paperback-Holt", "Hardcover-SES", "Hardcover-Holt"),
  lower = c(pb_ses$lower[1, "95%"],
            pb_holt$lower[1, "95%"],
            hc_ses$lower[1, "95%"],
            hc_holt$lower[1, "95%"]),
  upper = c(pb_ses$upper[1, "95%"],
            pb_holt$upper[1, "95%"],
            hc_ses$upper[1, "95%"],
            hc_holt$upper[1, "95%"])
)
df2
```

```
##      Pred_Int    lower    upper
## 1 Paperback-SES 138.8670 275.3523
## 2 Paperback-Holt 143.9130 275.0205
## 3 Hardcover-SES 174.7799 304.3403
## 4 Hardcover-Holt 192.9222 307.4256
```

From the interval range above, it is apparent that the interval calculated using RMSE is slightly narrower than from R using holt and ses methods.

## Exercise 7.7

For this exercise use data set `eggs`, the price of a dozen eggs in the United States from 1900–1993. Experiment with the various options in the `holt()` function to see how much the forecasts change with the damped trend, or with a Box-Cox transformation. Try to develop an intuition of what each argument is doing to the forecasts.

[Hint: use `h = 100` when calling `holt()` so you can clearly see the differences between the various options when plotting the forecasts.]

Which model gives the best RMSE?

## Exercise 7.8

Recall your retail time series data (from Exercise 3 in Section 2.10).

a)

Why is multiplicative seasonality necessary for this series?

b)

Apply Holt-Winters' multiplicative method to the data. Experiment with making the trend damped.

c)

Compare the RMSE of the one-step forecasts from the two methods. Which do you prefer?

d)

Check that the residuals from the best method look like white noise.

e)

Now find the test set RMSE while training the model to the end of 2010. Can you beat the seasonal naive approach from Exercise 8 in Section 3.7?

## Exercise 7.9

For the same retail data, try an STL decomposition applied to the Box-Cox transformed series, followed by ETS on the seasonally adjusted data. How does that compare with your best previous forecasts on the test set?