Data 624: Week 5 Homework

Angrand, Burke, Deboch, Groysman, Karr

October 12, 2019

Week 5 Assignment

Chapter 7 HA 7.5, 7.6 and 7.10

7.5 Forecast the next four days of paperback and hardcover books using the data set books which contains the same store daily sales data for paperback and hardcover books.

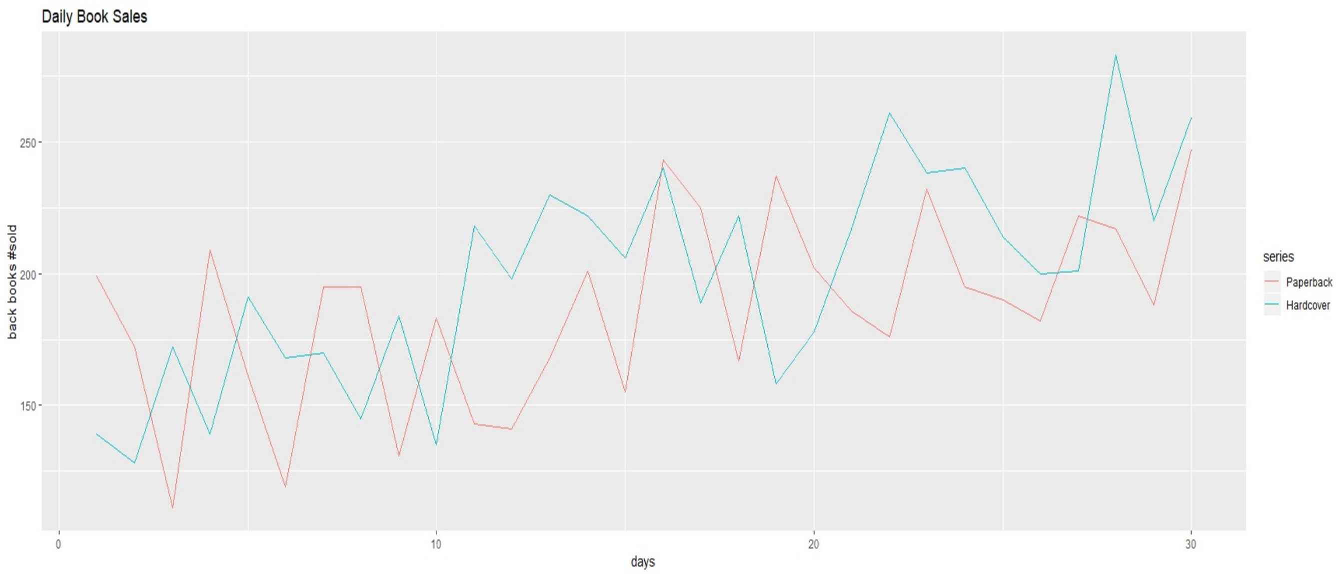
The problem set uses the books timeseries which contains 30 observations of same-store hardback and paperback sales.

```
#sampling and shape of dataset, preliminary EDA
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

dim(books)

## [1] 30  2
```

a. Plot the series and discuss its main features.

- This shows the 30 day timeseries for both types of books by number sold. Visually, the main features of the data are the positive trend and a seasonality pattern of approximately every three days (up/down).

```
data(books)
autoplot(books) +
  ylab("back books #sold") + xlab("days")+ ggtitle("Daily Book Sales")
```



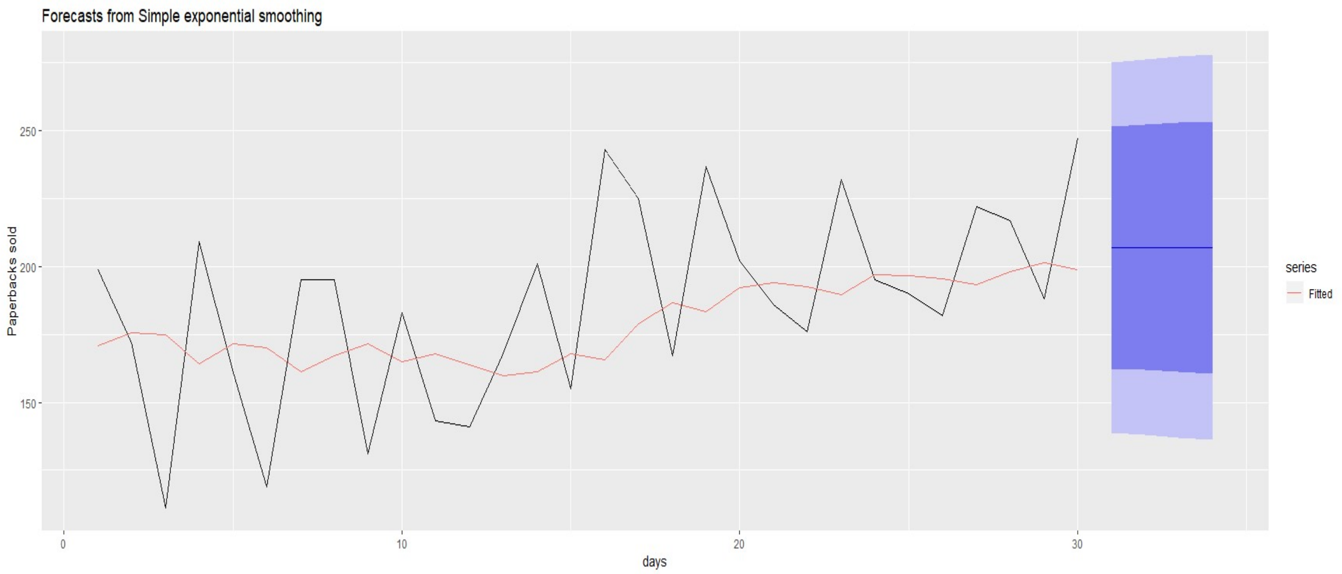
b. Uses the ses() function to forecast each series and plot the forecasts.

- Separate the paperback and hardback sales into distinct timeseries (books[,1], books[,2])
- Run the Simple Exponential Smoothing (SES) function against both hardbacks and paperbacks for h=4 or 4 days period of forecasting
  - The ses() function returns forecasts and metadata for exponential smoothing forecasts applied each timeseries
- Rounded the training errors and plotted the series

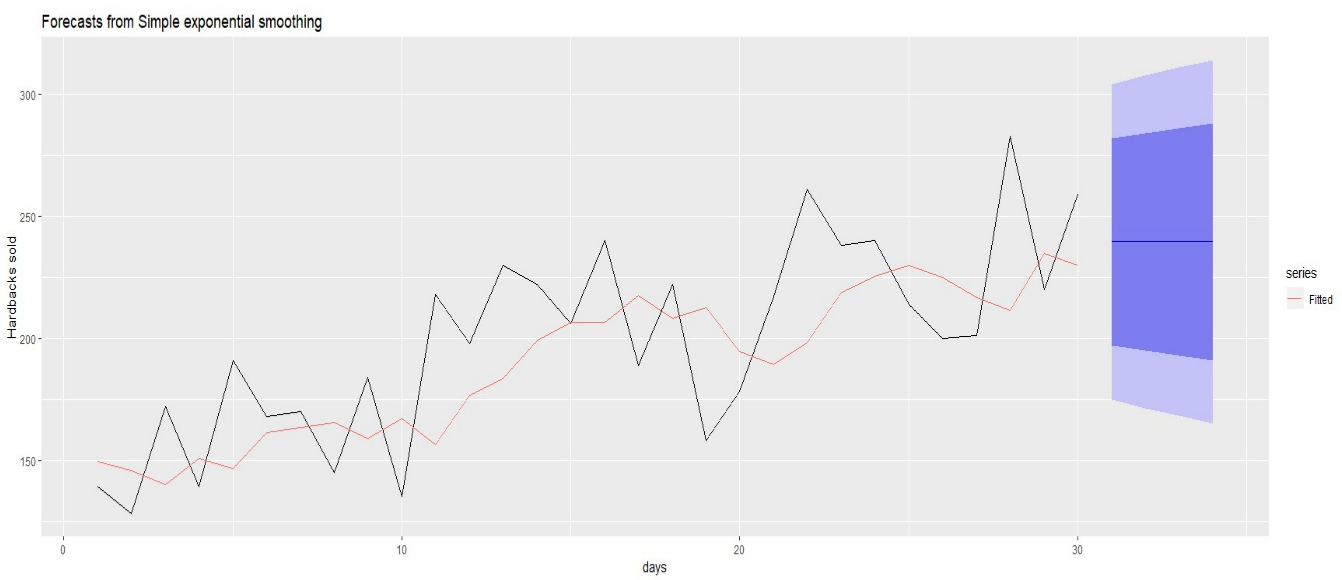
```
#1) create distinct timeseries
paperback_books_ts<-books[,1]
hardback_books_ts<-books[,2]

#2) Estimate parameters
fc_pb_ses<-ses(paperback_books_ts, h=4)
fc_hb_ses<-ses(hardback_books_ts, h=4)

data.frame(fc_pb_ses)

##    Point.Forecast    Lo.80    Hi.80    Lo.95    Hi.95
## 31       207.1097 162.4882 251.7311 138.8670 275.3523
## 32       207.1097 161.8589 252.3604 137.9046 276.3147
## 33       207.1097 161.2382 252.9811 136.9554 277.2639
## 34       207.1097 160.6259 253.5935 136.0188 278.2005

data.frame(fc_hb_ses)

##    Point.Forecast    Lo.80    Hi.80    Lo.95    Hi.95
## 31       239.5601 197.2026 281.9176 174.7799 304.3403
## 32       239.5601 194.9788 284.1414 171.3788 307.7414
```

```
## 33       239.5601 192.8607 286.2595 168.1396 310.9806
## 34       239.5601 190.8347 288.2855 165.0410 314.0792
```

```
autoplot(fc_pb_ses) +
  autolayer(fitted(fc_pb_ses),series="Fitted") +
  ylab("Paperbacks sold") + xlab("days")
```




```
autoplot(fc_hb_ses) +
  autolayer(fitted(fc_hb_ses),series="Fitted") +
  ylab("Hardbacks sold") + xlab("days")
```



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

- The accuracy function returns a range of summary measures of the forecast accuracy including Root Mean Square Error. for each training data timeseries including RMSE. For paperbacks books RMSE=33.64 and for hardback books RMSE=31.93.

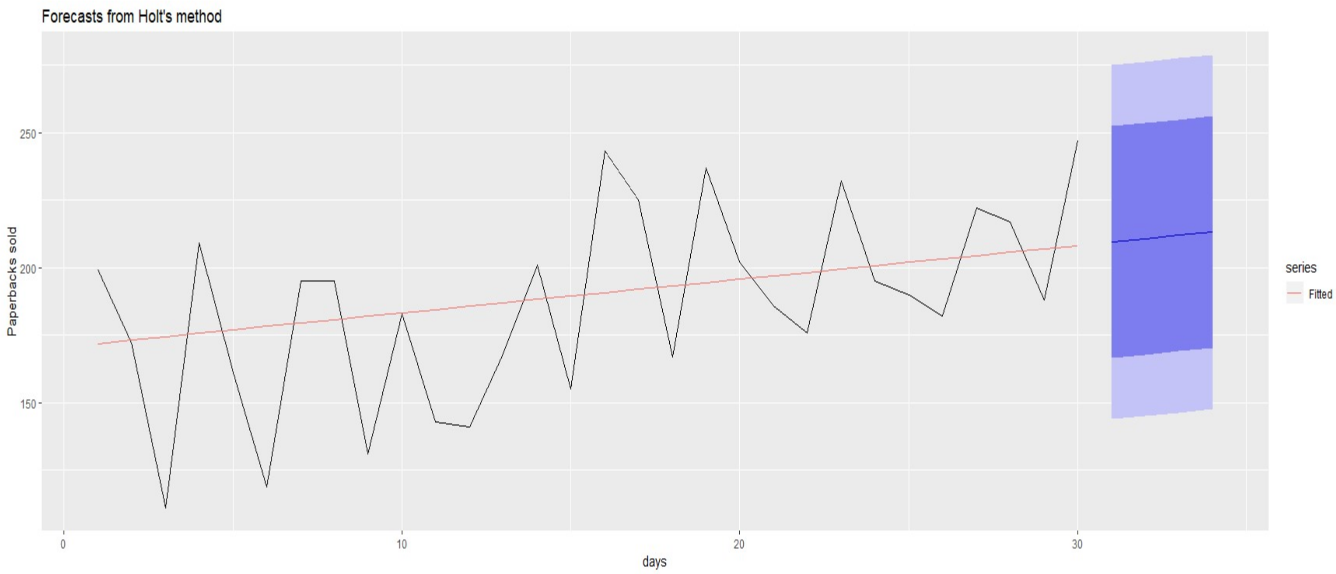
```
#3) Accuracy of one-step-ahead training errors paperback
round(accuracy(fc_pb_ses),2)

##                ME  RMSE   MAE  MPE  MAPE MASE  ACF1
## Training set 7.18 33.64 27.84 0.47 15.58  0.7 -0.21

#4) Accuracy of one-step-ahead training errors hardback
round(accuracy(fc_hb_ses),2)

##                ME  RMSE   MAE  MPE  MAPE MASE  ACF1
## Training set 9.17 31.93 26.77 2.64 13.39  0.8 -0.14
```

7.6 (a continuation of problem 7.5)

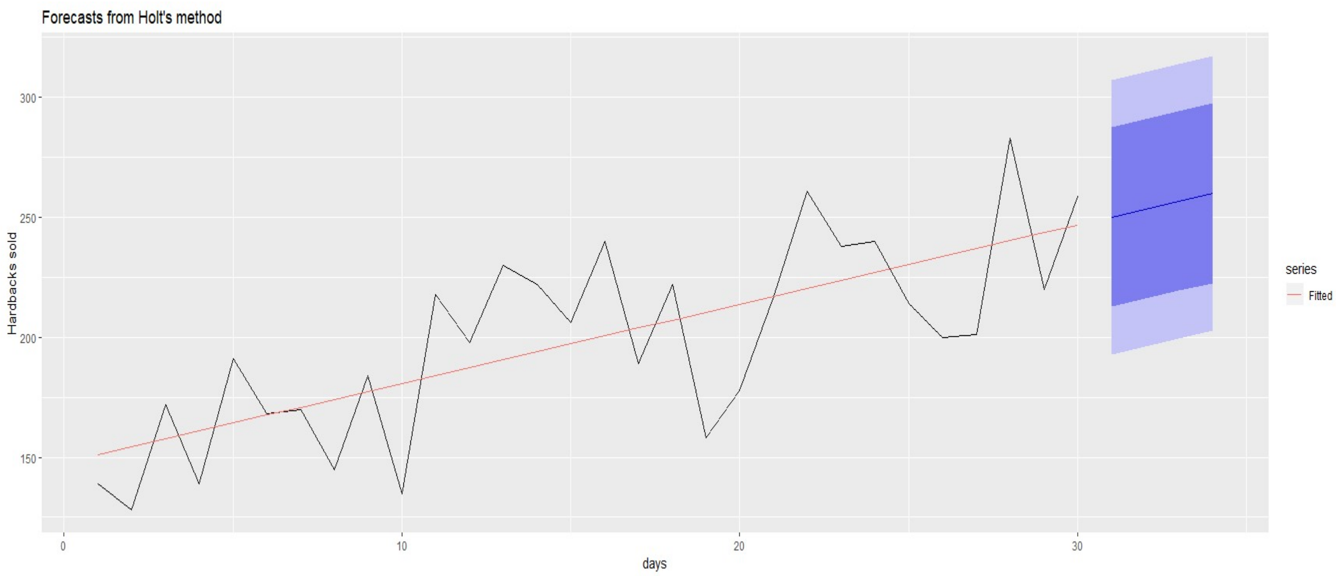
a. Now apply Holt's linear method to the paperback and hardback series and compute four-day forecasts in each case.

- The Holt method uses h=4 (4 days) as input parameters projecting a 4 day forecast.

```
# repeat with holt() functiobn, same params as ses()
fc_pb_holt<-holt(paperback_books_ts, h=4)
fc_hb_holt<-holt(hardback_books_ts, h=4)

data.frame(fc_pb_holt)

##    Point.Forecast    Lo.80    Hi.80    Lo.95    Hi.95
## 31       209.4668 166.6035 252.3301 143.9130 275.0205
## 32       210.7177 167.8544 253.5811 145.1640 276.2715
## 33       211.9687 169.1054 254.8320 146.4149 277.5225
## 34       213.2197 170.3564 256.0830 147.6659 278.7735

data.frame(fc_hb_holt)

##    Point.Forecast    Lo.80    Hi.80    Lo.95    Hi.95
## 31       250.1739 212.7390 287.6087 192.9222 307.4256
## 32       253.4765 216.0416 290.9113 196.2248 310.7282
## 33       256.7791 219.3442 294.2140 199.5274 314.0308
## 34       260.0817 222.6468 297.5166 202.8300 317.3334

autoplot(fc_pb_holt) +
  autolayer(fitted(fc_pb_holt),series="Fitted") +
  ylab("Paperbacks sold") + xlab("days")
```




```
autoplot(fc_hb_holt) +
  autolayer(fitted(fc_hb_holt),series="Fitted") +
  ylab("Hardbacks sold") + xlab("days")
```



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.

```
#HOLT Accuracy of one-step-ahead training errors paperback
round(accuracy(fc_pb_holt),2)

##                  ME  RMSE   MAE   MPE  MAPE MASE  ACF1
## Training set -3.72 31.14 26.18 -5.51 15.58 0.66 -0.18
```

```r
#SES Accuracy of one-step-ahead training errors paperback
round(accuracy(fc_pb_ses),2)

##                ME  RMSE   MAE  MPE  MAPE MASE  ACF1
## Training set 7.18 33.64 27.84 0.47 15.58  0.7 -0.21

#HOLT Accuracy of one-step-ahead training errors hardback
round(accuracy(fc_hb_holt),2)

##                 ME  RMSE   MAE   MPE  MAPE MASE  ACF1
## Training set -0.14 27.19 23.16 -2.11 12.16 0.69 -0.03

#SES Accuracy of one-step-ahead training errors hardback
round(accuracy(fc_hb_ses),2)

##                ME  RMSE   MAE  MPE  MAPE MASE  ACF1
## Training set 9.17 31.93 26.77 2.64 13.39  0.8 -0.14
```

- The holt series yields an RMSE of 31.14 for paperback books and 27.19 for hardback books. The ses series yields an RMSE of 33.64 for paperback books and 31.93 for hardback books. The smaller value of RMSE for hardback books indicates a better fit for both series using the Holt method. Using both methods hardbacks yield a lower RMSE which indicates a better fit then paperbacks. It appears that by extending the SES method with a trend equation for forecasting, the overall fit of the Holt method is an improvement over the SES method, presumably where the data being examined has a clear trend.

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

- The RMSE is smaller for the paperback book data and therefore better fitted using the holt model. The trend line from the training data to the predictions does seem to extrapolate more accurately as well.

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

- Note the 95% confidence intervals for the ses method have a greater range then that of the holt method for these predictions.The Holt 95% prediction interval for the 1st forecast of the paperback timeseries is between 149.69 and 265.45 and the Holt 95% prediction interval for the 1st forecast of the hardback timeseries is between 197.78 and 293.05. This compares with the ses 95% prediction interval for the 1st forecast of the paperback timeseries is between 135.96 and 277.16 and the ses 95% prediction interval for the 1st forecast of the hardback timeseries is between 197.78 and 293.05.

```r
head(data.frame(holt(paperback_books_ts,bootstrap=TRUE)),1)

##    Point.Forecast    Lo.80   Hi.80    Lo.95    Hi.95
## 31       209.4668 162.1512 251.972 149.6907 265.4503

head(data.frame(ses(paperback_books_ts,bootstrap=TRUE)),1)
```

```
##    Point.Forecast    Lo.80    Hi.80   Lo.95    Hi.95
## 31       207.1097 159.1981 246.1439 135.961 277.1593
```

```
head(data.frame(holt(hardback_books_ts,bootstrap=TRUE)),1)
```

```
##    Point.Forecast    Lo.80    Hi.80    Lo.95    Hi.95
## 31       250.1739 214.3904 289.6273 197.7726 293.0467
```

```
head(data.frame(ses(hardback_books_ts,bootstrap=TRUE)),1)
```

```
##    Point.Forecast    Lo.80    Hi.80    Lo.95    Hi.95
## 31       239.5601 202.0118 291.8263 175.7541 301.8855
```

- The tsCV function computes the forecast errors obtained by applying forecast function to subsets of the time series paperback_books_ts and hardback_books_ts using a rolling forecast origin.

```
#get the tsCV for each model
e1<-tsCV(paperback_books_ts,ses,h=4)
e2<-tsCV(paperback_books_ts,holt,h=4)
e3<-tsCV(paperback_books_ts,holt,damped=TRUE,h=4)
e4<-tsCV(hardback_books_ts,ses,h=4)
e5<-tsCV(hardback_books_ts,holt,h=4)
e6<-tsCV(hardback_books_ts,holt, damped=TRUE,h=4)

#Compare MSE for paperbacks:
mse.pb <-
data.frame(mean(e1^2,na.rm=TRUE),mean(e2^2,na.rm=TRUE),mean(e3^2,na.rm=TRUE))
names(mse.pb)<- c("mse.pb.ses", "mse.pb.holt", "mse.pb.holt.damped")
mse.pb
```

```
##   mse.pb.ses mse.pb.holt mse.pb.holt.damped
## 1   1475.265    2129.625           2152.224
```

```
#Compare MSE for hardbacks
mse.hb <-
data.frame(mean(e4^2,na.rm=TRUE),mean(e5^2,na.rm=TRUE),mean(e6^2,na.rm=TRUE))
names(mse.hb)<- c("mse.hb.ses", "mse.hb.holt", "mse.hb.holt.damped")
mse.hb
```

```
##   mse.hb.ses mse.hb.holt mse.hb.holt.damped
## 1   1500.838    1554.578           1589.293
```

- display Holt model summary

```
fc_pb_holt<-holt(paperback_books_ts,h=4)
fc_hb_holt<-holt(hardback_books_ts,h=4)

fc_pb_holt[["model"]]
```

```
## Holt's method
##
## Call:
```

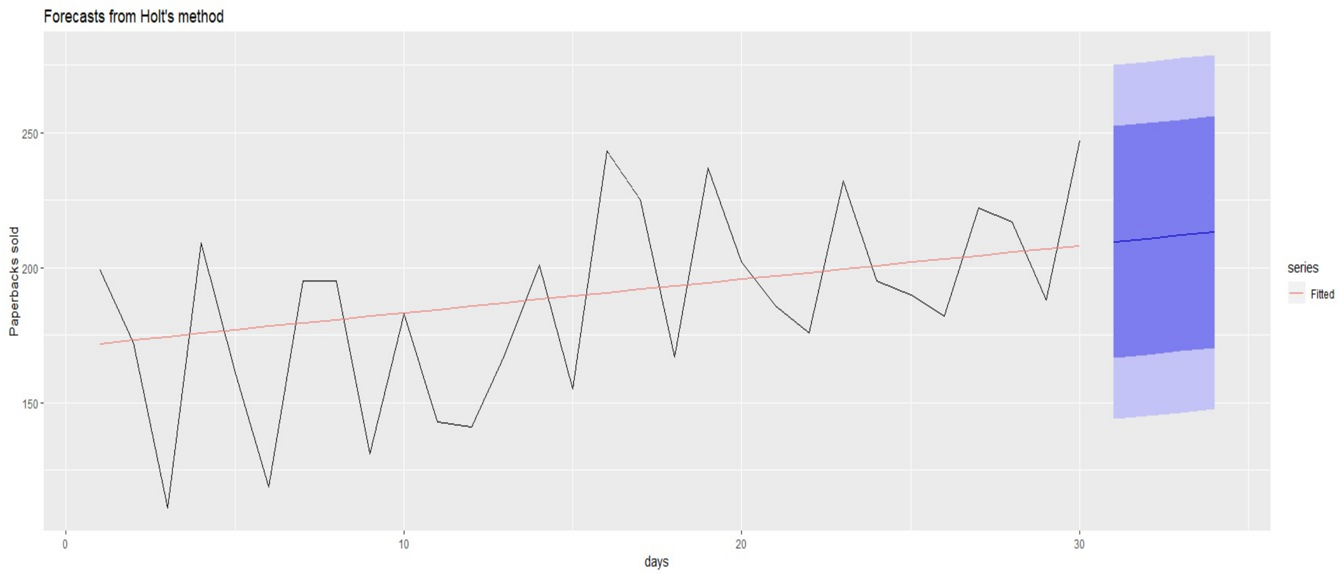
```
##  holt(y = paperback_books_ts, h = 4)
##
##   Smoothing parameters:
##     alpha = 1e-04
##     beta  = 1e-04
##
##   Initial states:
##     l = 170.699
##     b = 1.2621
##
##   sigma:  33.4464
##
##      AIC     AICc      BIC
## 318.3396 320.8396 325.3456
```

```
fc_hb_holt[["model"]]
```

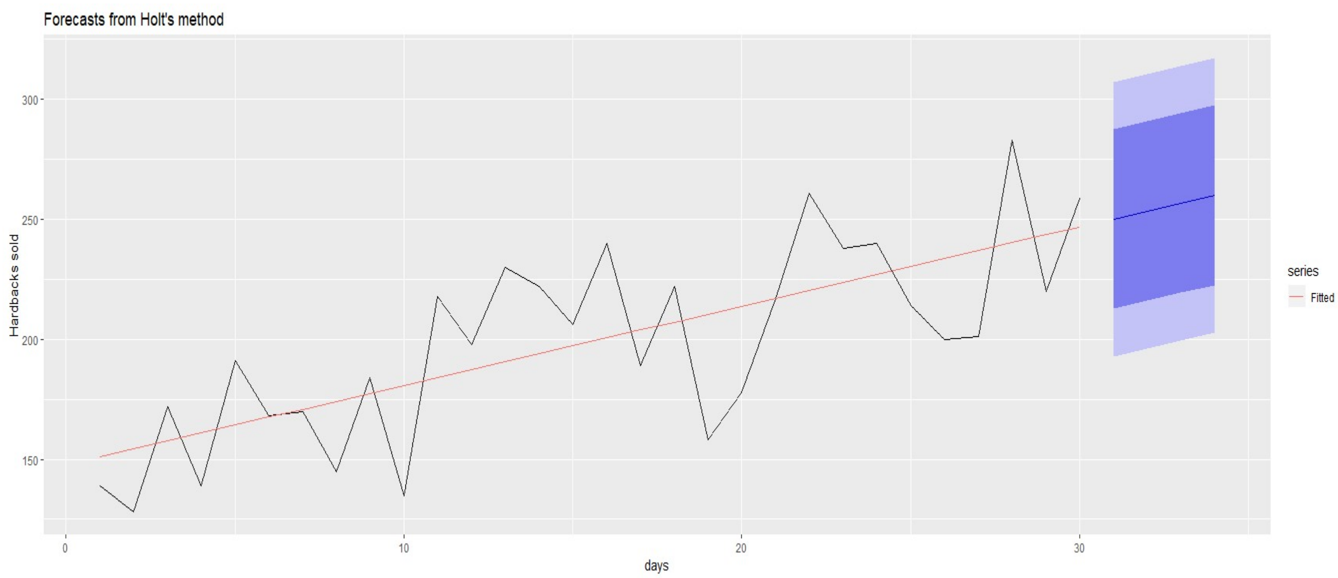
```
## Holt's method
##
## Call:
##  holt(y = hardback_books_ts, h = 4)
##
##   Smoothing parameters:
##     alpha = 1e-04
##     beta  = 1e-04
##
##   Initial states:
##     l = 147.7935
##     b = 3.303
##
##   sigma:  29.2106
##
##      AIC     AICc      BIC
## 310.2148 312.7148 317.2208
```

- Plot Holt model summary forecasts

```
autoplot(fc_pb_holt) +
  autolayer(fitted(fc_pb_holt),series="Fitted") +
  ylab("Paperbacks sold") + xlab("days")
```




```r
autoplot(fc_hb_holt) +
  autolayer(fitted(fc_hb_holt),series="Fitted") +
  ylab("Hardbacks sold") + xlab("days")
```




```r
round(accuracy(fc_pb_holt),2)

##                 ME  RMSE   MAE   MPE  MAPE MASE  ACF1
## Training set -3.72 31.14 26.18 -5.51 15.58 0.66 -0.18

round(accuracy(fc_hb_holt),2)

##                 ME  RMSE   MAE   MPE  MAPE MASE  ACF1
## Training set -0.14 27.19 23.16 -2.11 12.16 0.69 -0.03

round(accuracy(fc_pb_holt,damped=TRUE),2)
```

```
##                    ME  RMSE   MAE   MPE  MAPE MASE  ACF1
## Training set -3.72 31.14 26.18 -5.51 15.58 0.66 -0.18
```

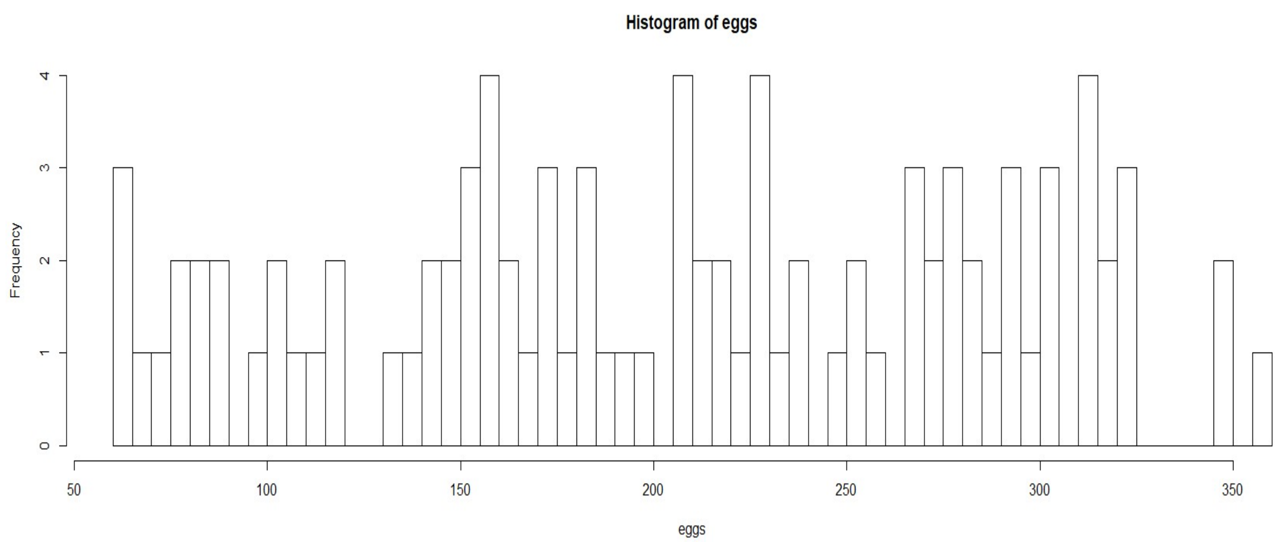
```r
round(accuracy(fc_hb_holt,damped=TRUE),2)
```

```
##                    ME  RMSE   MAE   MPE  MAPE MASE  ACF1
## Training set -0.14 27.19 23.16 -2.11 12.16 0.69 -0.03
```

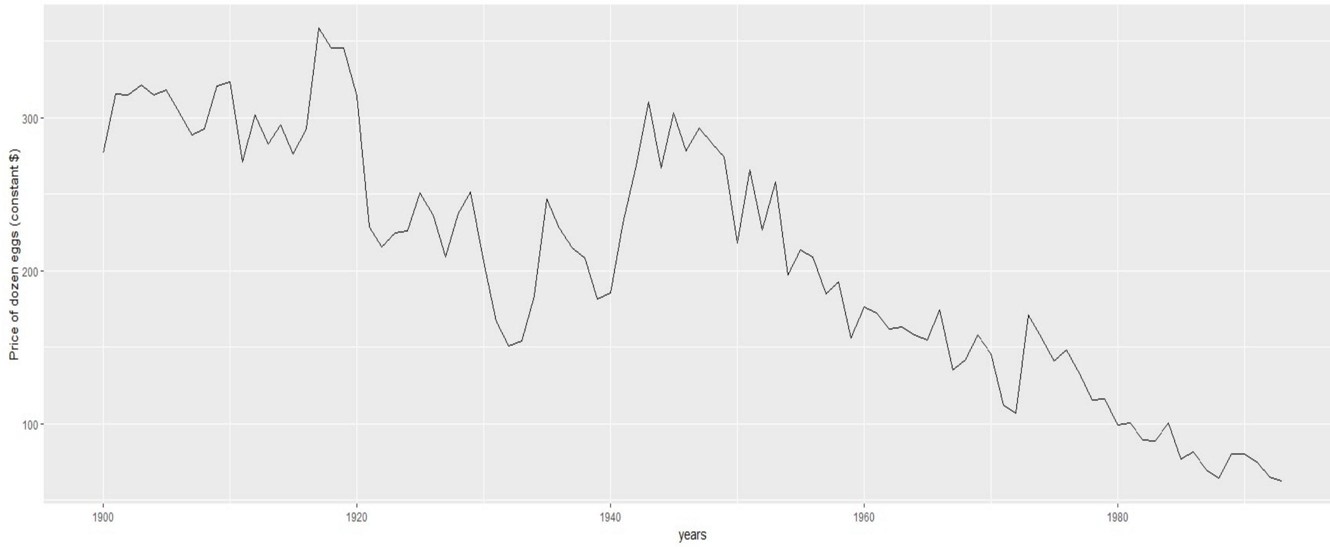
7.7 For this exercise use data set eggs, the price of a dozen eggs in the United States from 1900-1993.

a. Experiment with the various options in the holt() function to see how much the forecasts change with damped trend or Box-Cox transformation.

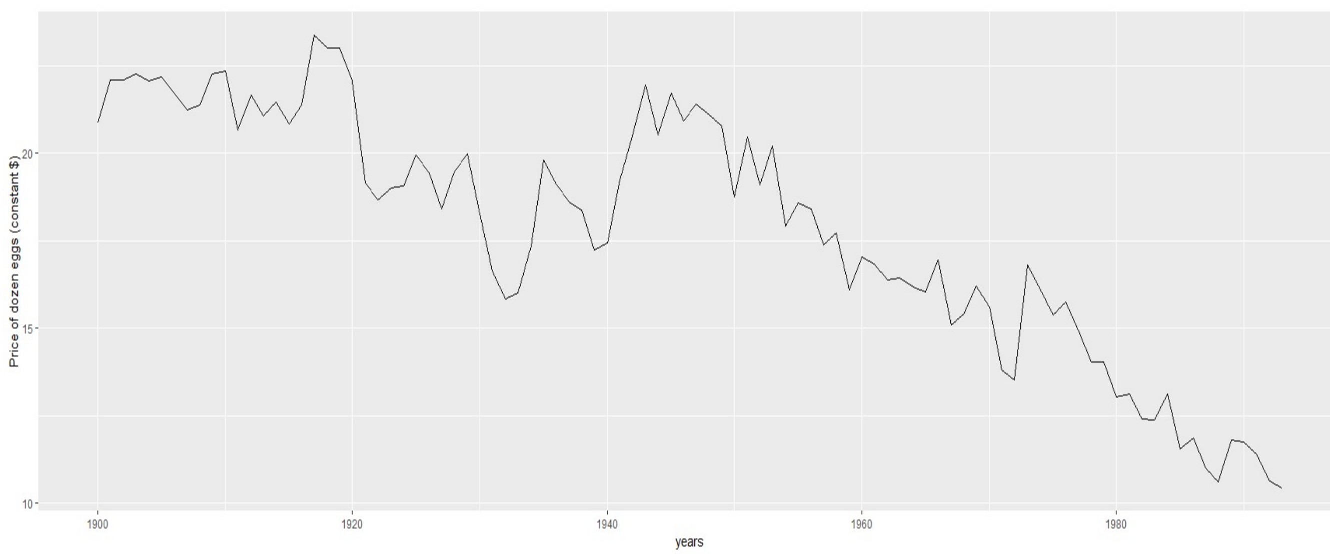
```r
data(eggs)
hist(eggs,breaks=100)
```




```r
#Price of dozen eggs in US, 1900 to 1993, in constant dollars.
autoplot(eggs) +
  ylab("Price of dozen eggs (constant $)") + xlab("years")
```




```
lambda<-BoxCox.lambda(eggs)
autoplot(BoxCox(eggs,lambda)) +
  ylab("Price of dozen eggs (constant $)") + xlab("years")
```



b. 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.]

- MSE Evaluations

```
e1<-tsCV(eggs,ses,h=100)
e2<-tsCV(eggs,holt,h=100)
e3<-tsCV(eggs,holt,damped=TRUE,h=100)

#Compare MSE for eggs:
mse.eggs <- data.frame(mean(e1^2,na.rm=TRUE),
```

```r
mean(e2^2,na.rm=TRUE),mean(e3^2,na.rm=TRUE))
names(mse.eggs)<- c("mse.ses", "mse.holt", "mse.holt.damped")
mse.eggs

##    mse.ses mse.holt mse.holt.damped
## 1 13200.06 77028.22        77608.15
```

- Model Components Summary

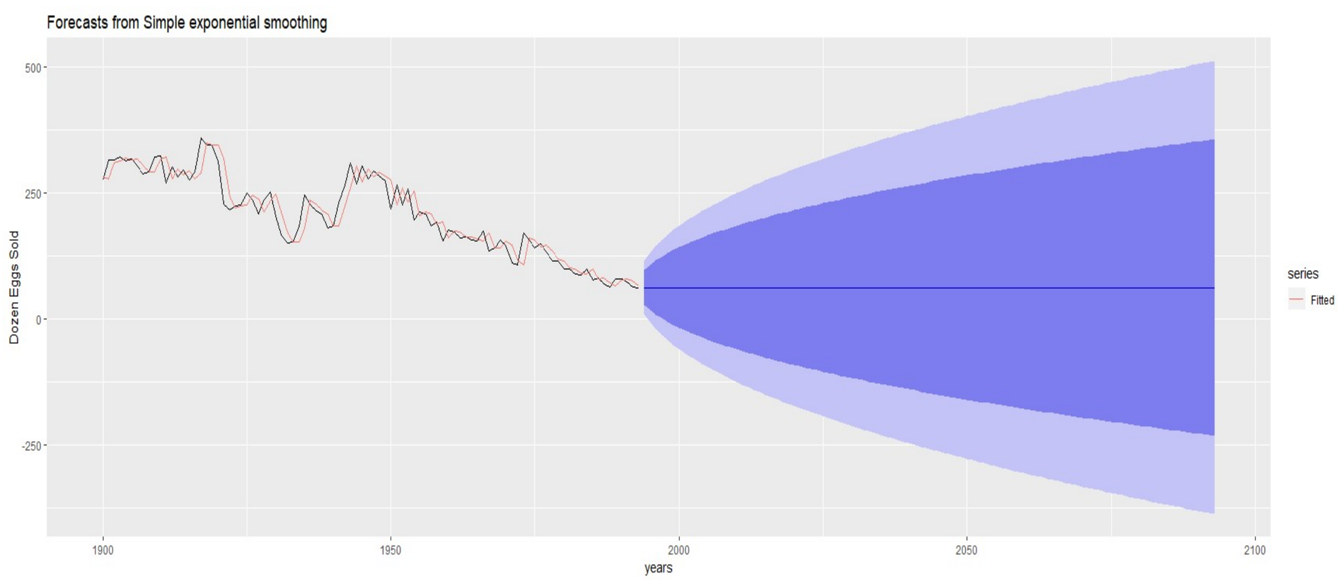
```r
fc_eggs_ses<-ses(eggs,h=100)
fc_eggs_holt<-holt(eggs,h=100)
fc_eggs_holt_damped<-holt(eggs,damped=TRUE,h=100)

# Build Models
fc_eggs_ses[["model"]]

## Simple exponential smoothing
##
## Call:
##  ses(y = eggs, h = 100)
##
##   Smoothing parameters:
##     alpha = 0.8525
##
##   Initial states:
##     l = 282.4981
##
##   sigma:  26.8511
##
##      AIC     AICc      BIC
## 1049.626 1049.893 1057.256

fc_eggs_holt[["model"]]

## Holt's method
##
## Call:
##  holt(y = eggs, h = 100)
##
##   Smoothing parameters:
##     alpha = 0.8124
##     beta  = 1e-04
##
##   Initial states:
##     l = 314.7232
##     b = -2.7222
##
##   sigma:  27.1665
##
##      AIC     AICc      BIC
## 1053.755 1054.437 1066.472
```

```
fc_eggs_holt_damped[["model"]]
```

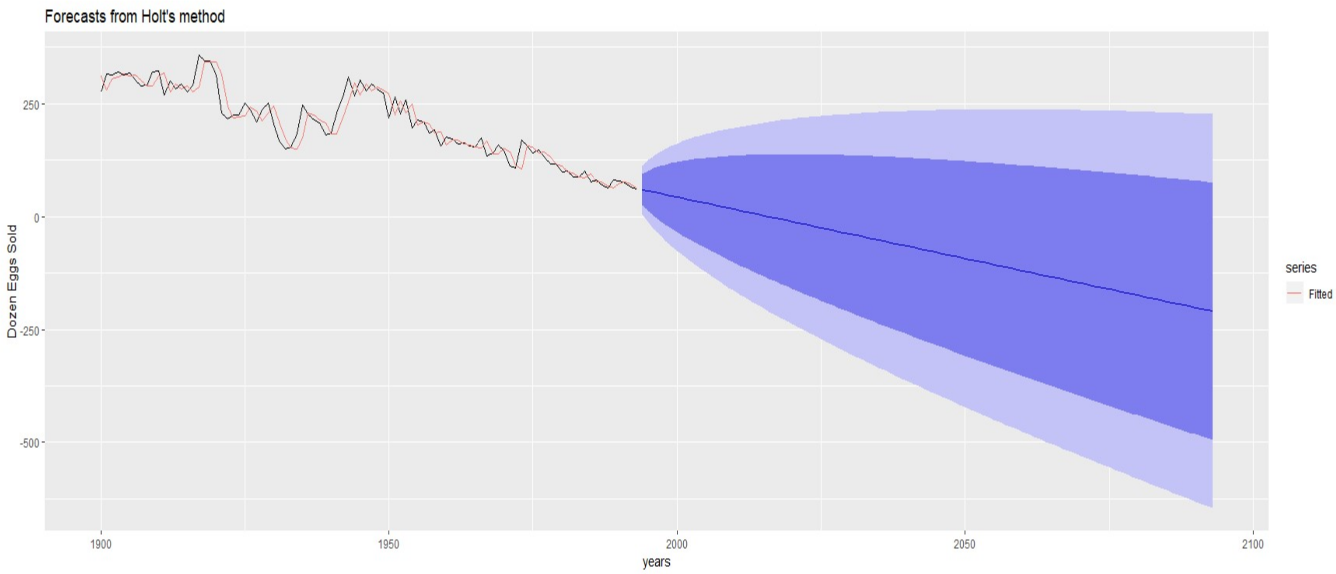
```
## Damped Holt's method
##
## Call:
##  holt(y = eggs, h = 100, damped = TRUE)
##
##   Smoothing parameters:
##     alpha = 0.8462
##     beta  = 0.004
##     phi   = 0.8
##
##   Initial states:
##     l = 276.9842
##     b = 4.9966
##
##   sigma:  27.2755
##
##      AIC     AICc      BIC
## 1055.458 1056.423 1070.718
```

- Plot Forecasts

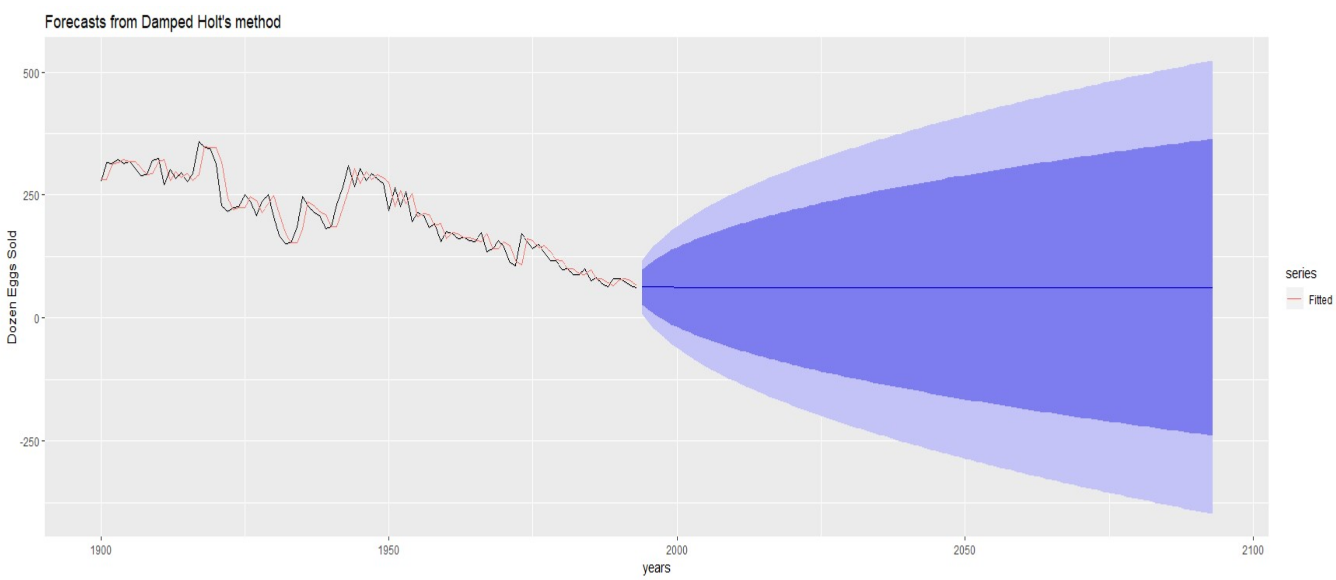
```
#plot forecasts predictions
autoplot(fc_eggs_ses) +
  autolayer(fitted(fc_eggs_ses),series="Fitted") +
  ylab("Dozen Eggs Sold") + xlab("years")
```



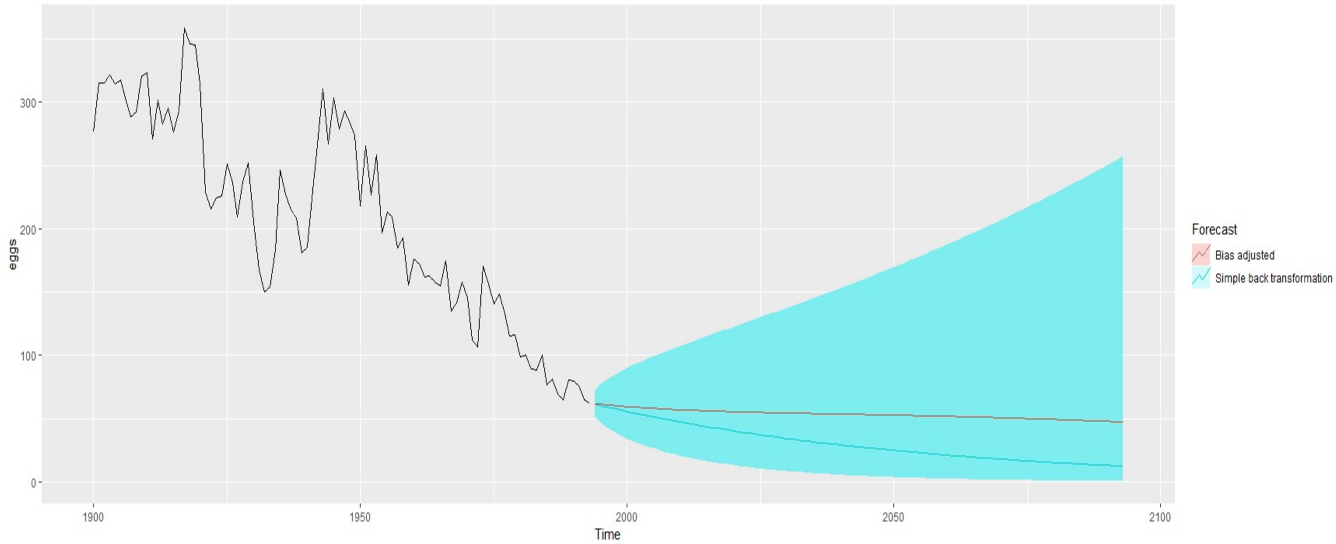

```
autoplot(fc_eggs_holt) +
  autolayer(fitted(fc_eggs_holt),series="Fitted") +
  ylab("Dozen Eggs Sold") + xlab("years")
```




```r
autoplot(fc_eggs_holt_damped) +
  autolayer(fitted(fc_eggs_holt_damped),series="Fitted") +
  ylab("Dozen Eggs Sold") + xlab("years")
```




```r
fc_eggs_BC<-rwf(eggs,drift=TRUE,lambda=0,h=100,level=80)
fc2_eggs_BC<-rwf(eggs,drift=TRUE,lambda=0,h=100,level=80,biasadj=TRUE)

autoplot(eggs) +
  autolayer(fc_eggs_BC,series="Simple back transformation") +
  autolayer(fc2_eggs_BC,series="Bias adjusted",PI=FALSE) +
  guides(colour=guide_legend(title="Forecast"))
```



c. Which model gives the best RMSE?

RMSE–the square root of a variance–can be interpreted as the standard deviation of the unexplained variance. The Holt damped RMSE is the lowest and best fit model in this case. Lower values indicate better fit. Based upon the diagrams, the Holt model and the simple back transformation of BoxCox appear to show the most accurate trendlines, but based on being the lowest RMSE values, the damped Holt model has the best fit.