Homework 3 solution

```
# hw3sol.r

d0 = read.csv("hw3.csv",header=T)
head(d0)
```

```
##           X X2012 X2013 X2014 X2015 X2016 X2012.1 X2013.1 X2014.1 X2015.1
## 1   January    NA  1.45  2.31  2.31  2.56      NA    46.8    46.8    43.8
## 2 February     NA  1.80  1.89  1.99  2.28      NA    48.0    48.6    45.6
## 3    March     NA  2.03  2.02  2.42  2.69      NA    60.0    59.4    57.6
## 4    April     NA  1.99  2.23  2.45  2.48      NA    57.6    58.2    53.4
## 5      May     NA  2.32  2.39  2.57  2.73      NA    61.8    60.6    56.4
## 6     June     NA  2.20  2.14  2.42  2.37      NA    58.2    55.2    52.8
##   X2016.1
## 1    48.0
## 2    51.6
## 3    57.6
## 4    58.2
## 5    60.0
## 6    57.0
```

```
d1 = d0[,1:6]
names(d1) = c("Month","2012","2013","2014","2015","2016")
d1
```

```
##        Month 2012 2013 2014 2015 2016
## 1    January   NA 1.45 2.31 2.31 2.56
## 2   February   NA 1.80 1.89 1.99 2.28
## 3      March   NA 2.03 2.02 2.42 2.69
## 4      April   NA 1.99 2.23 2.45 2.48
## 5        May   NA 2.32 2.39 2.57 2.73
## 6       June   NA 2.20 2.14 2.42 2.37
## 7       July   NA 2.13 2.27 2.40 2.31
## 8     August   NA 2.43 2.21 2.50 2.23
## 9  September 1.71 1.90 1.89 2.09   NA
## 10   October 1.90 2.13 2.29 2.54   NA
## 11  November 2.74 2.56 2.83 2.97   NA
## 12  December 4.20 4.16 4.04 4.35   NA
```

```
d2 = d0[,c(1,7:11)]
names(d2) = names(d1)
d2
```

```
##        Month  2012  2013  2014  2015  2016
## 1    January    NA  46.8  46.8  43.8  48.0
## 2   February    NA  48.0  48.6  45.6  51.6
## 3      March    NA  60.0  59.4  57.6  57.6
## 4      April    NA  57.6  58.2  53.4  58.2
## 5        May    NA  61.8  60.6  56.4  60.0
## 6       June    NA  58.2  55.2  52.8  57.0
## 7       July    NA  56.4  51.0  54.0  57.6
## 8     August    NA  63.0  58.8  60.6  61.8
## 9  September  55.8  57.6  49.8  47.4  69.0
## 10   October  56.4  53.4  54.6  54.6  75.0
## 11  November  71.4  71.4  65.4  67.8  85.2
## 12  December 117.6 114.0 102.0 100.2 121.8
```

```
#
# gather
library(tidyr)
#
# table d1
aux = names(d1)[-1]
aux
```

```
## [1] "2012" "2013" "2014" "2015" "2016"
```

```
d1 = gather(d1,aux,key='year',value='sales')
head(d1,12)
```

```
##        Month year sales
## 1    January 2012    NA
## 2   February 2012    NA
## 3      March 2012    NA
## 4      April 2012    NA
## 5        May 2012    NA
## 6       June 2012    NA
## 7       July 2012    NA
## 8     August 2012    NA
## 9  September 2012  1.71
## 10   October 2012  1.90
## 11  November 2012  2.74
## 12  December 2012  4.20
```

```
tail(d1,12)
```

```
##        Month year sales
## 49   January 2016  2.56
## 50  February 2016  2.28
## 51     March 2016  2.69
## 52     April 2016  2.48
## 53       May 2016  2.73
## 54      June 2016  2.37
## 55      July 2016  2.31
## 56    August 2016  2.23
## 57 September 2016    NA
## 58   October 2016    NA
## 59  November 2016    NA
## 60  December 2016    NA
```

```
str(d1)
```

```
## 'data.frame':    60 obs. of  3 variables:
##  $ Month: Factor w/ 12 levels "April","August",..: 5 4 8 1 9 7 6 2 12 11 ...
##  $ year : chr  "2012" "2012" "2012" "2012" ...
##  $ sales: num  NA NA NA NA NA NA NA NA 1.71 1.9 ...
```

```
#
# add column of month number to estimate the trend component
row = 1:60
d1 = data.frame(row,d1)
# select 2016 Q4
newval1 = d1[57:60,]
```

```r
newval1
```

```
##    row     Month year sales
## 57  57 September 2016    NA
## 58  58   October 2016    NA
## 59  59  November 2016    NA
## 60  60  December 2016    NA
```

```r
# gather table d2
d2 = gather(d2,aux,key='year',value='sales')
head(d2,12)
```

```
##        Month year sales
## 1    January 2012    NA
## 2   February 2012    NA
## 3      March 2012    NA
## 4      April 2012    NA
## 5        May 2012    NA
## 6       June 2012    NA
## 7       July 2012    NA
## 8     August 2012    NA
## 9  September 2012  55.8
## 10   October 2012  56.4
## 11  November 2012  71.4
## 12  December 2012 117.6
```

```r
tail(d2,12)
```

```
##        Month year sales
## 49   January 2016  48.0
## 50  February 2016  51.6
## 51     March 2016  57.6
## 52     April 2016  58.2
## 53       May 2016  60.0
## 54      June 2016  57.0
## 55      July 2016  57.6
## 56    August 2016  61.8
## 57 September 2016  69.0
## 58   October 2016  75.0
## 59  November 2016  85.2
## 60  December 2016 121.8
```

```r
str(d2)
```

```
## 'data.frame':    60 obs. of  3 variables:
##  $ Month: Factor w/ 12 levels "April","August",..: 5 4 8 1 9 7 6 2 12 11 ...
##  $ year : chr  "2012" "2012" "2012" "2012" ...
##  $ sales: num  NA NA NA NA NA NA NA NA 55.8 56.4 ...
```

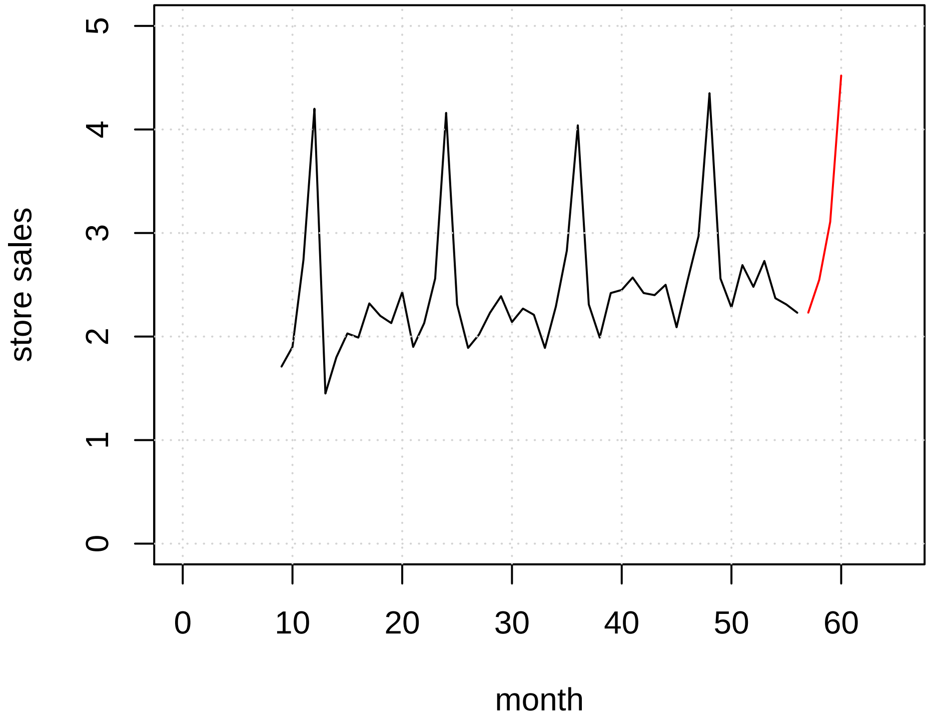
```r
#
# add column of month number to estimate the trend component
d2 = data.frame(row,d2)
# select 2016 Q4
newval2 = d2[57:60,]
newval2
```

```
##    row     Month year sales
## 57  57 September 2016  69.0
## 58  58   October 2016  75.0
## 59  59  November 2016  85.2
## 60  60  December 2016 121.8
```

```
#
#
#
#
# 1) Store sales had there been no hurricane
#
# plot store sales
row = 1:60
yb = c(0,5)
xb = c(0,65)
plot(d1$sales~row,type='l',ylim=yb,xlim=xb,ylab='store sales',xlab='month')
grid()
#
# predict store sales
train = 9:56
model1 = lm(sales~.-year,d1,subset = train)
pred = predict(model1,newval1)
newval1$sales = pred
newval1
```

```
##    row     Month year    sales
## 57  57 September 2016 2.230833
## 58  58   October 2016 2.548333
## 59  59  November 2016 3.108333
## 60  60  December 2016 4.520833
```

```
# plot prediction
newrow = 57:60
lines(pred~newrow,col='red')
#
# alternatively you may use ets()
library(ggplot2)  # ylab() with autoplot()
```



```r
library(forecast)
```

```
## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Registered S3 methods overwritten by 'forecast':
##   method             from
##   fitted.fracdiff    fracdiff
##   residuals.fracdiff fracdiff
```

```r
d11 = d1$sales
d11 = na.omit(d11)
head(d11)
```

```
## [1] 1.71 1.90 2.74 4.20 1.45 1.80
```

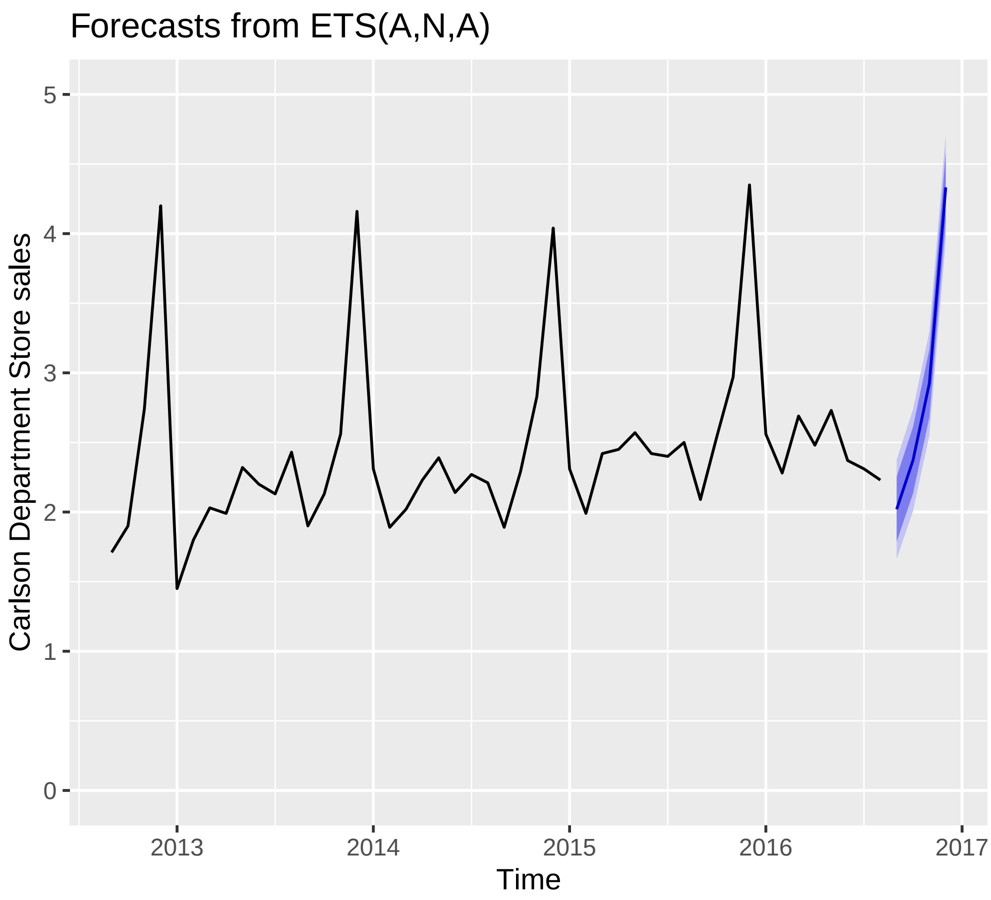
```r
ts1 = ts(d11,start = c(2012,9),end = c(2016,8),frequency = 12)
ts1
```

```
##       Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2012                                         1.71 1.90 2.74 4.20
## 2013 1.45 1.80 2.03 1.99 2.32 2.20 2.13 2.43 1.90 2.13 2.56 4.16
## 2014 2.31 1.89 2.02 2.23 2.39 2.14 2.27 2.21 1.89 2.29 2.83 4.04
## 2015 2.31 1.99 2.42 2.45 2.57 2.42 2.40 2.50 2.09 2.54 2.97 4.35
## 2016 2.56 2.28 2.69 2.48 2.73 2.37 2.31 2.23
```

```r
# exp smoothing model
es1 = ets(ts1)
es1pred = forecast(es1,4)
es1pred
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Sep 2016       2.020107 1.787240 2.252974 1.663967 2.376246
## Oct 2016       2.372952 2.134495 2.611409 2.008264 2.737640
## Nov 2016       2.925468 2.681550 3.169386 2.552427 3.298509
## Dec 2016       4.332404 4.083144 4.581665 3.951193 4.713615
```

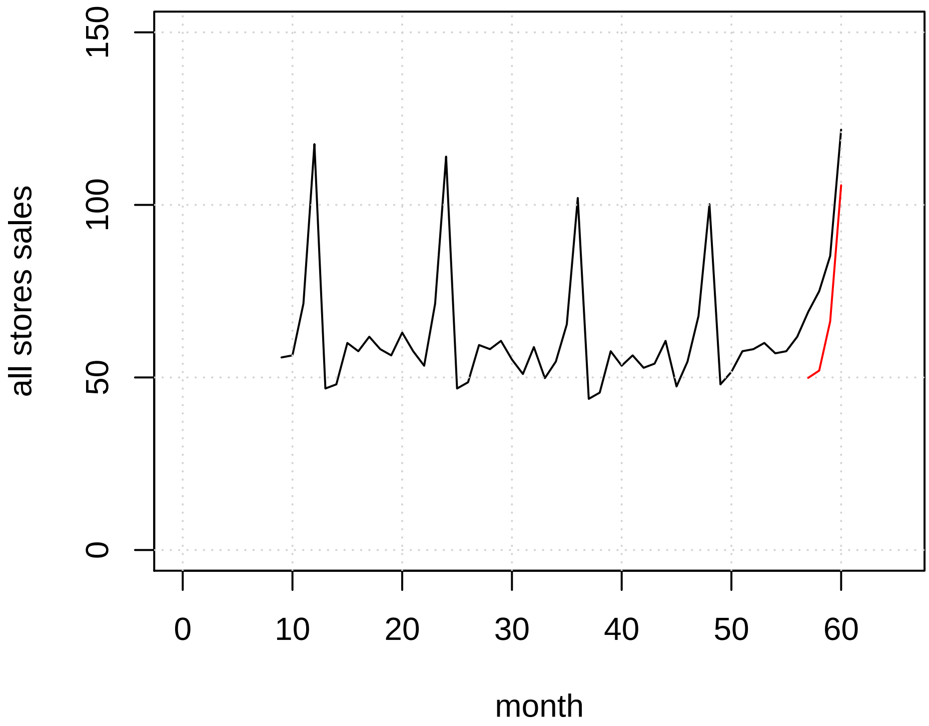
```r
autoplot(forecast(es1,4)) +
  ylab('Carlson Department Store sales') + ylim(0,5)
```



```
# 2) County-wide sales had there been no hurricane
#
# plot all stores sales
row = 1:60
yb = c(0,150)
xb = c(0,65)
plot(d2$sales~row,type='l',ylim=yb,xlim=xb,ylab='all stores sales',xlab='month')
grid()
#
# predict county sales
train = 9:56
model2 = lm(sales~.-year,d2,subset = train)
pred = predict(model2,newval2)
newval2$sales = pred
names(newval2)[4]='pred_sales'
newval2
```

```
##    row     Month year pred_sales
## 57  57 September 2016    49.8875
## 58  58   October 2016    51.9875
## 59  59  November 2016    66.2375
## 60  60  December 2016   105.6875
```

```
# plot prediction
newrow = 57:60
lines(pred~newrow,col='red')
```



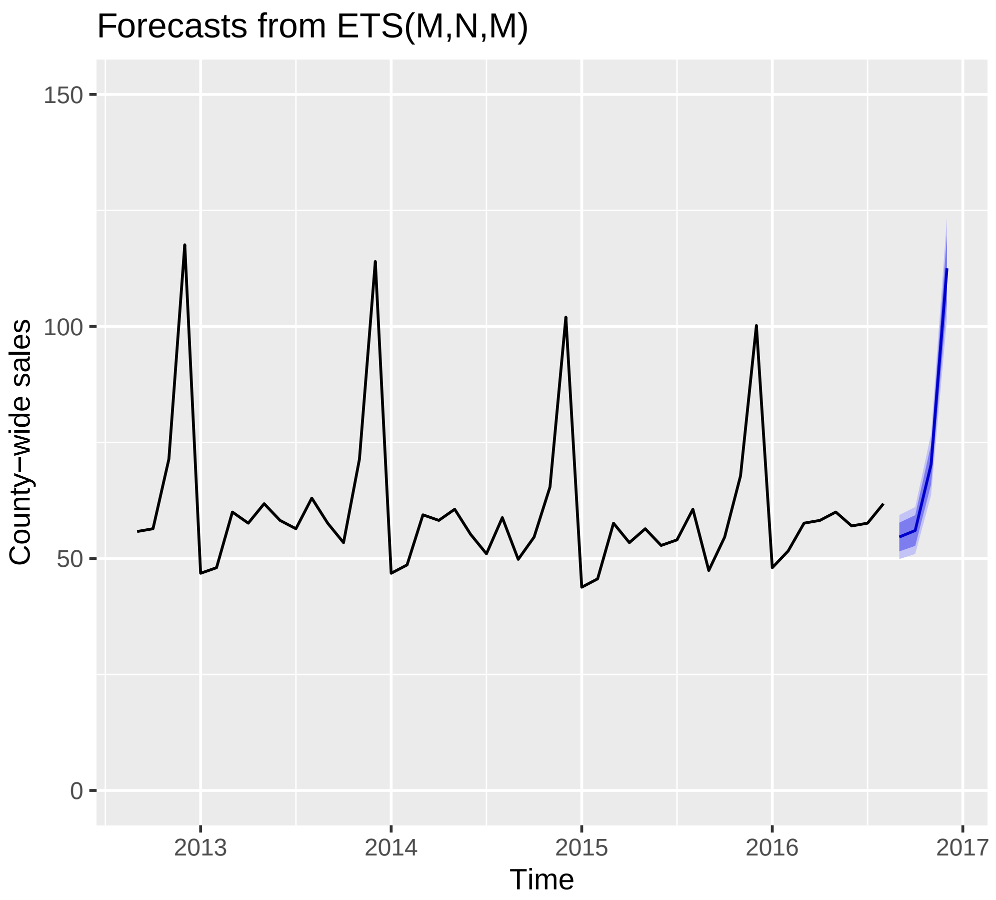
```
#
# prediction shows that there was excess sales related with the hurricane effects
#
#
#
# alternatively you may use ets()
d22 = d2$sales[9:56]
ts2 = ts(d22,start = c(2012,9),end = c(2016,8),frequency = 12)
ts2
```

```
##        Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
## 2012                                                  55.8  56.4  71.4 117.6
## 2013  46.8  48.0  60.0  57.6  61.8  58.2  56.4  63.0  57.6  53.4  71.4 114.0
## 2014  46.8  48.6  59.4  58.2  60.6  55.2  51.0  58.8  49.8  54.6  65.4 102.0
## 2015  43.8  45.6  57.6  53.4  56.4  52.8  54.0  60.6  47.4  54.6  67.8 100.2
## 2016  48.0  51.6  57.6  58.2  60.0  57.0  57.6  61.8
```

```
# exp smoothing model
es2 = ets(ts2)
es2pred = forecast(es2,4)
es2pred
```

```
##          Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## Sep 2016       54.61294  51.51457  57.71132  49.87438  59.35151
## Oct 2016       56.01602  52.70717  59.32488  50.95557  61.07648
## Nov 2016       70.23764  65.93084  74.54444  63.65096  76.82433
## Dec 2016      112.53414 105.38978 119.67850 101.60779 123.46050
```

```
autoplot(forecast(es2,4)) + ylab('County-wide sales') + ylim(0,150)
```



```
#
# for questions (3) and (4) I use the predictions from lm()
#
#
# 3) Store lost sales (regular and excess storm related)
#
table3 = d2[57:60,]
table4 = merge(table3,newval2)
table4
```

```
##   row     Month year sales pred_sales
## 1  57 September 2016  69.0    49.8875
## 2  58   October 2016  75.0    51.9875
## 3  59  November 2016  85.2    66.2375
## 4  60  December 2016 121.8   105.6875
```

```
#
# actual sales exceed predicted sales
# the difference is excess storm related sales
#
# excess ratio
table4$ratio = table4$sales/table4$pred_sales
table4
```

```
##   row     Month year sales pred_sales    ratio
## 1  57 September 2016  69.0    49.8875 1.383112
## 2  58   October 2016  75.0    51.9875 1.442654
## 3  59  November 2016  85.2    66.2375 1.286280
## 4  60  December 2016 121.8   105.6875 1.152454
```

```
#
# lost sales = store predicted sales * ratio
#
ratio = table4$ratio
newval1$ratio = ratio
newval1$lost_sales = newval1$sales * ratio
names(newval1)[4]="pred_sales"
#
# Store sales
# regular sales lost
sum(newval1$pred_sales)
```

```
## [1] 12.40833
```

```
#
# excess storm related lost
sum(newval1$lost_sales) - sum(newval1$pred_sales)
```

```
## [1] 3.561765
```

```
#
# total lost sales
sum(newval1$lost_sales)
```

```
## [1] 15.9701
```

```
#
# Store may claim to the insurance company a total loss of 15,970,100 dollars.
```