snewt
ISYE6501x - HW#3
Due June 06, 2018

```
> setwd("c://users/sophia/desktop/isye6501x/hw#3")
> list.files(getwd())
[1] "7.2tempsSummer2018.txt"   "8.2uscrimeSummer2018.txt" "week_3_hw-summer2018.pdf"
> temps<- read.delim("7.2tempsSummer2018.txt",header=TRUE)
> head(temps)
   DAY X1996 X1997 X1998 X1999 X2000 X2001 X2002 X2003 X2004 X2005 X2006 X2007
X2008
1 1-Jul    98    86    91    84    89    84    90    73    82    91    93    95    85
2 2-Jul    97    90    88    82    91    87    90    81    81    89    93    85    87
3 3-Jul    97    93    91    87    93    87    87    87    86    86    93    82    91
4 4-Jul    90    91    91    88    95    84    89    86    88    86    91    86    90
5 5-Jul    89    84    91    90    96    86    93    80    90    89    90    88    88
6 6-Jul    93    84    89    91    96    87    93    84    90    82    81    87    82
  X2009 X2010 X2011 X2012 X2013 X2014 X2015
1    95    87    92   105    82    90    85
2    90    84    94    93    85    93    87
3    89    83    95    99    76    87    79
4    91    85    92    98    77    84    85
5    80    88    90   100    83    86    84
6    87    89    90    98    83    87    84
> temps[[1]][1:92]
 [1] 1-Jul  2-Jul  3-Jul  4-Jul  5-Jul  6-Jul  7-Jul  8-Jul  9-Jul  10-Jul 11-Jul
[12] 12-Jul 13-Jul 14-Jul 15-Jul 16-Jul 17-Jul 18-Jul 19-Jul 20-Jul 21-Jul 22-Jul
[23] 23-Jul 24-Jul 25-Jul 26-Jul 27-Jul 28-Jul 29-Jul 30-Jul 31-Jul 1-Aug  2-Aug
[34] 3-Aug  4-Aug  5-Aug  6-Aug  7-Aug  8-Aug  9-Aug  10-Aug 11-Aug 12-Aug 13-Aug
[45] 14-Aug 15-Aug 16-Aug 17-Aug 18-Aug 19-Aug 20-Aug 21-Aug 22-Aug 23-Aug 24-Aug
[56] 25-Aug 26-Aug 27-Aug 28-Aug 29-Aug 30-Aug 31-Aug 1-Sep  2-Sep  3-Sep  4-Sep
[67] 5-Sep  6-Sep  7-Sep  8-Sep  9-Sep  10-Sep 11-Sep 12-Sep 13-Sep 14-Sep 15-Sep
[78] 16-Sep 17-Sep 18-Sep 19-Sep 20-Sep 21-Sep 22-Sep 23-Sep 24-Sep 25-Sep 26-Sep
[89] 27-Sep 28-Sep 29-Sep 30-Sep
123 Levels: 1-Aug 1-Jul 1-Oct 1-Sep 10-Aug 10-Jul 10-Oct 10-Sep 11-Aug ... 9-Sep
> hot_dates<-temps[[1]][1:92]
> temp <- cusum(temps[[1]][1:92],
+               center = avg_summer[1],
+               std.dev = sd_summer[1],
+               decision.interval = 2,
+               se.shift = 3,
+               plot = TRUE)
Error in cusum(temps[[1]][1:92], center = avg_summer[1], std.dev = sd_summer[1],  :
  could not find function "cusum"
> tail(temps)
      DAY X1996 X1997 X1998 X1999 X2000 X2001 X2002 X2003 X2004 X2005 X2006 X2007
118 26-Oct    75    71    79    69    75    64    68    68    79    61    62    68
```

snewt
ISYE6501x - HW#3
Due June 06, 2018

```
119 27-Oct   75   57   79   75   78   51   69   64   81   63   66   67
120 28-Oct   81   55   79   73   80   55   75   57   78   62   63   70
121 29-Oct   82   64   78   72   75   63   75   70   75   64   72   62
122 30-Oct   82   66   82   75   77   72   68   77   78   69   73   67
123 31-Oct   81   60   79   75   78   71   60   75   82   70   68   71
    X2008 X2009 X2010 X2011 X2012 X2013 X2014 X2015
118    70    65    85    77    80    61    84    67
119    59    60    76    79    70    69    84    56
120    50    71    74    74    56    64    77    78
121    59    75    68    59    56    75    73    70
122    65    66    71    61    56    78    68    70
123    67    69    75    65    65    74    63    62
> dates<-temps[[1]][1:123]
> temps[[2]][1:5]
[1] 98 97 97 90 89
> mean(temps[[2]][1:5])
[1] 94.2
> mean(98,97,97,90,89)
[1] 98
> avg_summer<-c()
> avg_summer<-for(i in 2:21){
+     t<-temps[[i]][1:92] #check out the i'th column, first 92 days (q3)
+     avg_summer<- c(avg_summer,mean(t))
+ }
> avg_summer
NULL
> t<-temps[[2]][1:92]
> mean(t)
[1] 87.1087
> avg_summer<-c()
> avg_summer<-for(i in 2:21){
+     t<-temps[[i]][1:92] #check out the i'th column, first 92 days (q3)
+     avg_summer<- c(avg_summer,mean(t))
+ }
> avg_summer<-c()
> avg_summer<-for(i in 2:21){
+     t<-temps[[i]][1:92] #check out the i'th column, first 92 days (q3)
+     avg_summer<- c(avg_summer,mean(t)); i<-i+1
+ }
> avg_summer
NULL
> avg_summer<-c()
> for(i in 2:21){
```

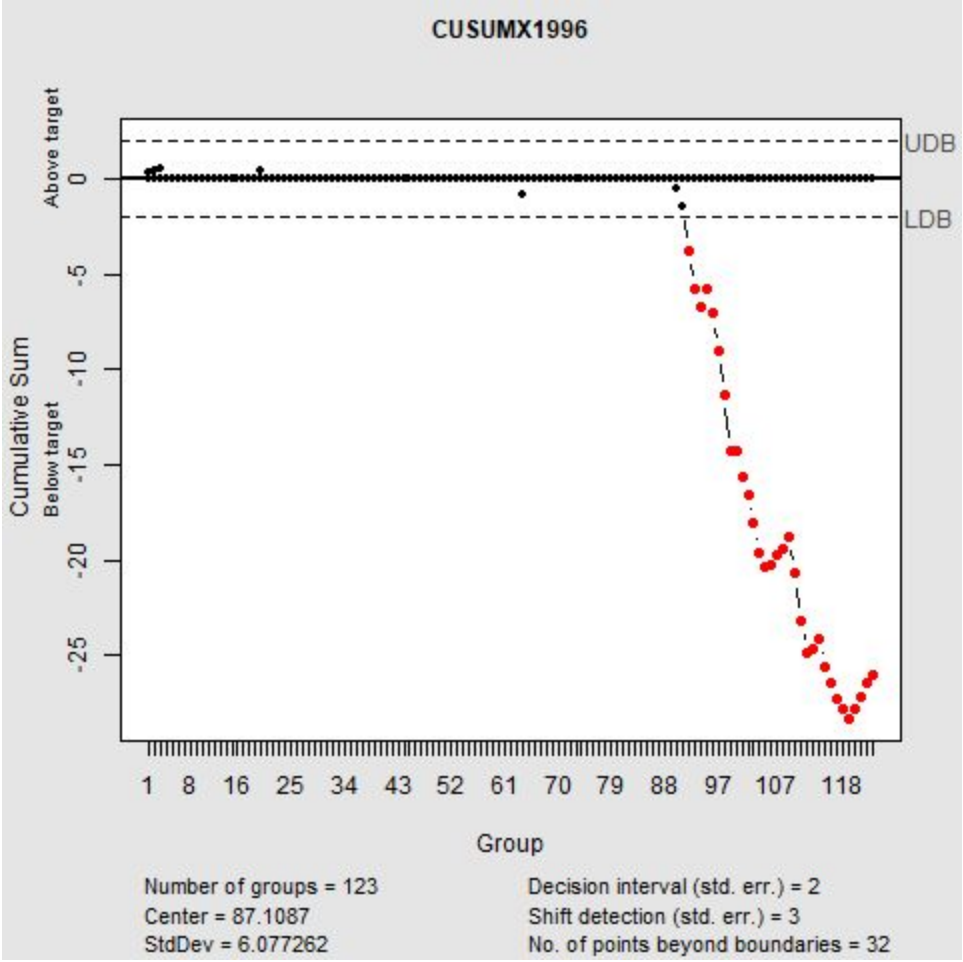
snewt
ISYE6501x - HW#3
Due June 06, 2018

```
+     t<-temps[[i]][1:92] #check out the i'th column, first 92 days (q3)
+     avg_summer<- c(avg_summer,mean(t)); i<-i+1
+ }
> names(temps)
 [1] "DAY"   "X1996" "X1997" "X1998" "X1999" "X2000" "X2001" "X2002" "X2003" "X2004"
[11] "X2005" "X2006" "X2007" "X2008" "X2009" "X2010" "X2011" "X2012" "X2013" "X2014"
[21] "X2015"
> year<-names(temps[2:21])
> sd(temps[[2]][1:92])
[1] 6.077262
> sd_summer<-c()
> for(i in 2:21){
+     t<-temps[[i]][1:92]
+     sd_summer<-c(sd_summer,sd(t))
+ }
> library(qcc)

  __ _  ___ ___
 / _  |/ __/ __|  Quality Control Charts and
| (_| | (_| (__   Statistical Process Control
 \__  |\___\___|
    |_|           version 2.7
Type 'citation("qcc")' for citing this R package in publications.
> cusum(temps["X1996"],center = avg_summer[1],std.dev = sd_summer[1],decision.interval =
2,se.shift = 3,plot=TRUE)
List of 14
 $ call            : language cusum(data = temps["X1996"], center = avg_summer[1], std.dev =
sd_summer[1], decision.interval = 2,      se.shift| __truncated__
 $ type            : chr "cusum"
 $ data.name       : chr "temps[\"X1996\"]"
 $ data            : int [1:123, 1] 98 97 97 90 89 93 93 91 93 93 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics      : Named int [1:123] 98 97 97 90 89 93 93 91 93 93 ...
  ..- attr(*, "names")= chr [1:123] "1" "2" "3" "4" ...
 $ sizes           : int [1:123] 1 1 1 1 1 1 1 1 1 1 ...
 $ center          : num 87.1
 $ std.dev         : num 6.08
 $ pos             : num [1:123] 0.292 0.42 0.547 0 0 ...
 $ neg             : num [1:123] 0 0 0 0 0 0 0 0 0 0 ...
 $ head.start      : num 0
 $ decision.interval: num 2
 $ se.shift        : num 3
 $ violations      :List of 2
 - attr(*, "class")= chr "cusum.qcc"
```

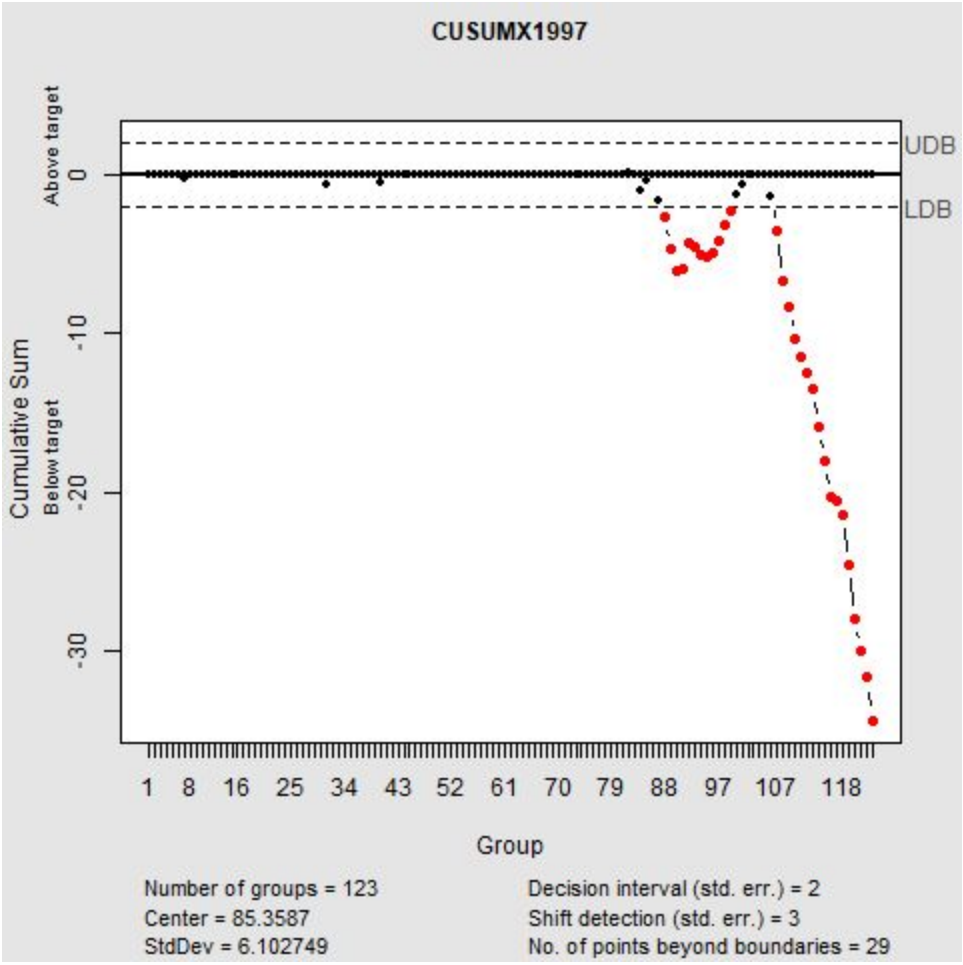
snewt
ISYE6501x - HW#3
Due June 06, 2018

Create cusum plots for all 20 years of data

```
for(i in 1:20){

  newpath<- file.path("c:","users","sophia","desktop","ISYE6501x","HW#3",
                      paste("CUSUM_",names(temps[i+1]),".png", sep=""))
  png(filename = newpath)
  cusum(temps[[i+1]],
        center = avg_summer[i],
        std.dev = sd_summer[i],
        decision.interval = 2,
        se.shift = 3,
        plot = TRUE,
        add.stats = TRUE,
        title = paste("CUSUM", names(temps[i+1]), sep = ""))
  dev.off()
}
```



snewt
ISYE6501x - HW#3
Due June 06, 2018



See: 2001, 2010, 2011, 2015 for early ends to summer (35+ points beyond boundaries)

Okay so to actually answer the question:

Question 7.2

Using Holt Winters formula:

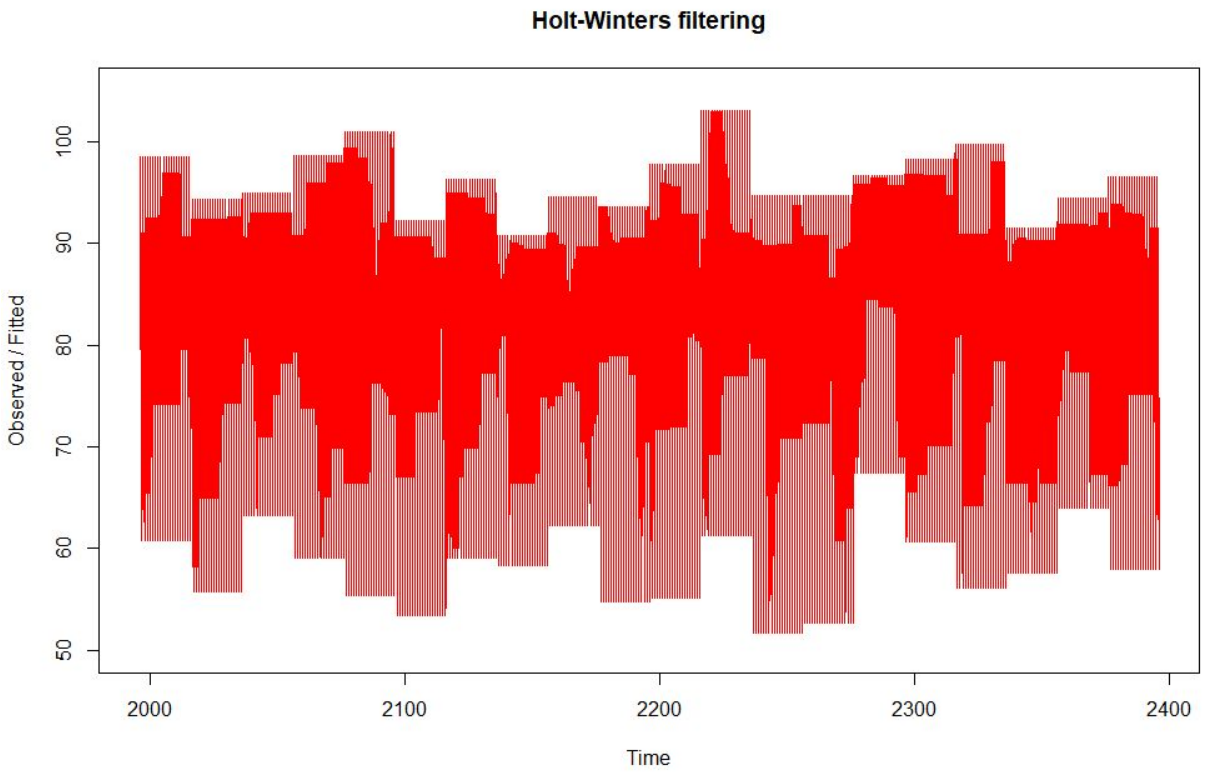
First we need to make it a time series data because it isn't yet.
We need frequency = 1 (1 temp/day),

```
> otemps <- temps[,2:21]
> View(otemps)
> my_ts<-ts(otemps,start=c(1996,1),end=c(2015,123),deltat = 1/123)
> my_ts
Time Series:
Start = c(1996, 1)
End = c(2015, 123)
Frequency = 123
```

snewt
ISYE6501x - HW#3
Due June 06, 2018

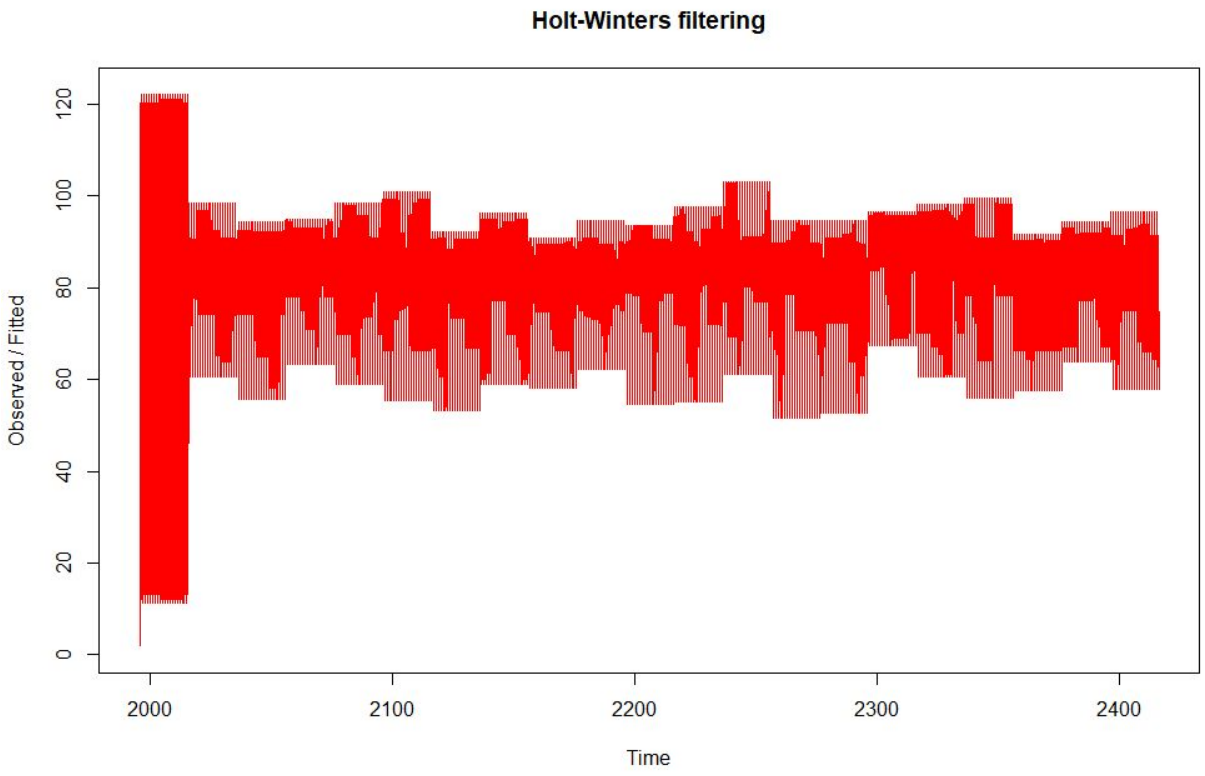
```
> hw <- HoltWinters(my_ts, beta = FALSE, gamma = FALSE)
> lines(fitted(hw)[,1], col=3)
Error in plot.xy(xy.coords(x, y), type = type, ...) :
  plot.new has not been called yet
> summary(hw)
             Length Class  Mode
fitted       98398  mts    numeric
x            49200  mts    numeric
alpha            1  -none- numeric
beta             1  -none- logical
gamma            1  -none- logical
coefficients     1  -none- numeric
seasonal         1  -none- character
SSE              1  -none- numeric
call             4  -none- call

> hw
Holt-Winters exponential smoothing without trend and without seasonal component.

Call:
HoltWinters(x = my_ts, beta = FALSE, gamma = FALSE)

Smoothing parameters:
 alpha: 0.8372888
 beta : FALSE
 gamma: FALSE

Coefficients:
      [,1]
a 63.32208
```

snewt
ISYE6501x - HW#3
Due June 06, 2018




```
> my_ts2<-ts(temps,start=c(1996,1),end=c(2015,123),deltat = 1/123)
> hw2 <- HoltWinters(my_ts2, beta = FALSE, gamma = FALSE)
> hw2
Holt-Winters exponential smoothing without trend and without seasonal component.

Call:
HoltWinters(x = my_ts2, beta = FALSE, gamma = FALSE)

Smoothing parameters:
 alpha: 0.8430084
 beta : FALSE
 gamma: FALSE

Coefficients:
      [,1]
a 63.27465
```

snewt
ISYE6501x - HW#3
Due June 06, 2018



So using the first one, hw, because the last one included the dates as a weird date somehow that doesn't match the rest.

That model includes beta = FALSE, meaning that we got an exponential smoothing model, gamma = FALSE so we also got a nil seasonal parameter to start with. The alpha value we started with was 0.8372888.

The additive Holt-Winters prediction function (for time series with period length p) is

Yhat[t+h] = a[t] + h * b[t] + s[t - p + 1 + (h - 1) mod p],

where a[t], b[t] and s[t] are given by

a[t] = α (Y[t] - s[t-p]) + (1-α) (a[t-1] + b[t-1])

b[t] = β (a[t] - a[t-1]) + (1-β) b[t-1]

s[t] = γ (Y[t] - a[t]) + (1-γ) s[t-p]

The multiplicative Holt-Winters prediction function (for time series with period length p) is

snewt
ISYE6501x - HW#3
Due June 06, 2018

Yhat[t+h] = (a[t] + h * b[t]) * s[t - p + 1 + (h - 1) mod p],

where a[t], b[t] and s[t] are given by

a[t] = α (Y[t] / s[t-p]) + (1-α) (a[t-1] + b[t-1])

b[t] = β (a[t] - a[t-1]) + (1-β) b[t-1]

s[t] = γ (Y[t] / a[t]) + (1-γ) s[t-p]

The data in x are required to be non-zero for a multiplicative model, but it makes most sense if they are all positive.

The function tries to find the optimal values of α and/or β and/or γ by minimizing the squared one-step prediction error if they are NULL (the default). optimize will be used for the single-parameter case, and optim otherwise.

For seasonal models, start values for a, b and s are inferred by performing a simple decomposition in trend and seasonal component using moving averages (see function decompose) on the start.periods first periods (a simple linear regression on the trend component is used for starting level and trend). For level/trend-models (no seasonal component), start values for a and b are x[2] and x[2] - x[1], respectively. For level-only models (ordinary exponential smoothing), the start value for a is x[1].

Question 8.2

```
 > lmcrime <- lm(crime$Crime ~.,data = crime)
> summary(lmcrime)

Call:
lm(formula = crime$Crime ~ ., data = crime)

Residuals:
    Min      1Q  Median      3Q     Max
-395.74  -98.09   -6.69  112.99  512.67

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
M            8.783e+01  4.171e+01   2.106 0.043443 *
So          -3.803e+00  1.488e+02  -0.026 0.979765
```

snewt
ISYE6501x - HW#3
Due June 06, 2018

```
Ed          1.883e+02  6.209e+01   3.033 0.004861 **
Po1         1.928e+02  1.061e+02   1.817 0.078892 .
Po2        -1.094e+02  1.175e+02  -0.931 0.358830
LF         -6.638e+02  1.470e+03  -0.452 0.654654
M.F         1.741e+01  2.035e+01   0.855 0.398995
Pop        -7.330e-01  1.290e+00  -0.568 0.573845
NW          4.204e+00  6.481e+00   0.649 0.521279
U1         -5.827e+03  4.210e+03  -1.384 0.176238
U2          1.678e+02  8.234e+01   2.038 0.050161 .
Wealth      9.617e-02  1.037e-01   0.928 0.360754
Ineq        7.067e+01  2.272e+01   3.111 0.003983 **
Prob       -4.855e+03  2.272e+03  -2.137 0.040627 *
Time       -3.479e+00  7.165e+00  -0.486 0.630708
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 209.1 on 31 degrees of freedom
Multiple R-squared:  0.8031,  Adjusted R-squared:  0.7078
F-statistic: 8.429 on 15 and 31 DF,  p-value: 3.539e-07
```