Homework_1-Q1.R

zeshi yang

Thu Feb 07 22:02:49 2019

```
library(MASS)
library(forecast)

## Warning: package 'forecast' was built under R version 3.5.2

#install.packages("forecast")

data(Boston)
summary(Boston)

##       crim                zn             indus            chas        
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000  
##  1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000  
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000  
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917  
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000  
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000  
##       nox               rm             age              dis        
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130  
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100  
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207  
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795  
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188  
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127  
##       rad              tax           ptratio          black       
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32  
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38  
##  Median : 5.000   Median :330.0   Median :19.05   Median :391.44  
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67  
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23  
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90  
##      lstat            medv      
##  Min.   : 1.73   Min.   : 5.00  
##  1st Qu.: 6.95   1st Qu.:17.02  
##  Median :11.36   Median :21.20  
##  Mean   :12.65   Mean   :22.53  
##  3rd Qu.:16.95   3rd Qu.:25.00  
##  Max.   :37.97   Max.   :50.00

Boston[1:2,]

##      crim zn indus chas   nox    rm  age    dis rad tax ptratio black
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.9
```

```
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.9
##   lstat medv
## 1  4.98 24.0
## 2  9.14 21.6
```

```r
#install.packages("GGally")
library("GGally")
```

```
## Warning: package 'GGally' was built under R version 3.5.2
```

```
## Loading required package: ggplot2
```

```r
library("ggplot2")
lmbos<-lm(medv~.,data=Boston)
summary(lmbos)
```

```
##
## Call:
## lm(formula = medv ~ ., data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.595  -2.730  -0.518   1.777  26.199
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
## zn           4.642e-02  1.373e-02   3.382 0.000778 ***
## indus        2.056e-02  6.150e-02   0.334 0.738288
## chas         2.687e+00  8.616e-01   3.118 0.001925 **
## nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## age          6.922e-04  1.321e-02   0.052 0.958229
## dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
## ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
## black        9.312e-03  2.686e-03   3.467 0.000573 ***
## lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

```r
step(lmbos,direction = "backward")
```

```
## Start:  AIC=1589.64
## medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad +
```

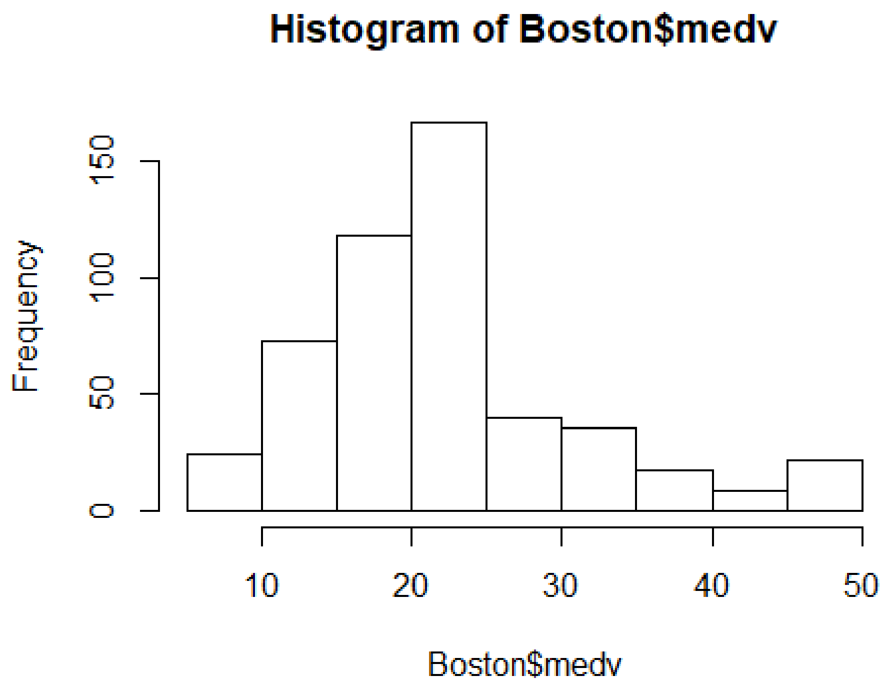
```
##     tax + ptratio + black + lstat
##
##           Df Sum of Sq   RSS    AIC
## - age      1      0.06 11079 1587.7
## - indus    1      2.52 11081 1587.8
## <none>                 11079 1589.6
## - chas     1    218.97 11298 1597.5
## - tax      1    242.26 11321 1598.6
## - crim     1    243.22 11322 1598.6
## - zn       1    257.49 11336 1599.3
## - black    1    270.63 11349 1599.8
## - rad      1    479.15 11558 1609.1
## - nox      1    487.16 11566 1609.4
## - ptratio  1   1194.23 12273 1639.4
## - dis      1   1232.41 12311 1641.0
## - rm       1   1871.32 12950 1666.6
## - lstat    1   2410.84 13490 1687.3
##
## Step:  AIC=1587.65
## medv ~ crim + zn + indus + chas + nox + rm + dis + rad + tax +
##     ptratio + black + lstat
##
##           Df Sum of Sq   RSS    AIC
## - indus    1      2.52 11081 1585.8
## <none>                 11079 1587.7
## - chas     1    219.91 11299 1595.6
## - tax      1    242.24 11321 1596.6
## - crim     1    243.20 11322 1596.6
## - zn       1    260.32 11339 1597.4
## - black    1    272.26 11351 1597.9
## - rad      1    481.09 11560 1607.2
## - nox      1    520.87 11600 1608.9
## - ptratio  1   1200.23 12279 1637.7
## - dis      1   1352.26 12431 1643.9
## - rm       1   1959.55 13038 1668.0
## - lstat    1   2718.88 13798 1696.7
##
## Step:  AIC=1585.76
## medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio +
##     black + lstat
##
##           Df Sum of Sq   RSS    AIC
## <none>                 11081 1585.8
## - chas     1    227.21 11309 1594.0
## - crim     1    245.37 11327 1594.8
## - zn       1    257.82 11339 1595.4
## - black    1    270.82 11352 1596.0
## - tax      1    273.62 11355 1596.1
## - rad      1    500.92 11582 1606.1
## - nox      1    541.91 11623 1607.9
```

```
## - ptratio  1   1206.45 12288 1636.0
## - dis      1   1448.94 12530 1645.9
## - rm       1   1963.66 13045 1666.3
## - lstat    1   2723.48 13805 1695.0

##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat, data = Boston)
##
## Coefficients:
## (Intercept)         crim           zn         chas          nox
##   36.341145    -0.108413     0.045845     2.718716   -17.376023
##          rm          dis          rad          tax      ptratio
##    3.801579    -1.492711     0.299608    -0.011778    -0.946525
##       black        lstat
##    0.009291    -0.522553
```

```r
step(lmbos,direction = "backward",k=log(506))
```

```
## Start:  AIC=1648.81
## medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + black + lstat
##
##           Df Sum of Sq   RSS    AIC
## - age      1      0.06 11079 1642.6
## - indus    1      2.52 11081 1642.7
## <none>                 11079 1648.8
## - chas     1    218.97 11298 1652.5
## - tax      1    242.26 11321 1653.5
## - crim     1    243.22 11322 1653.6
## - zn       1    257.49 11336 1654.2
## - black    1    270.63 11349 1654.8
## - rad      1    479.15 11558 1664.0
## - nox      1    487.16 11566 1664.4
## - ptratio  1   1194.23 12273 1694.4
## - dis      1   1232.41 12311 1696.0
## - rm       1   1871.32 12950 1721.6
## - lstat    1   2410.84 13490 1742.2
##
## Step:  AIC=1642.59
## medv ~ crim + zn + indus + chas + nox + rm + dis + rad + tax +
##     ptratio + black + lstat
##
##           Df Sum of Sq   RSS    AIC
## - indus    1      2.52 11081 1636.5
## <none>                 11079 1642.6
## - chas     1    219.91 11299 1646.3
## - tax      1    242.24 11321 1647.3
## - crim     1    243.20 11322 1647.3
```

```
## - zn       1    260.32 11339 1648.1
## - black    1    272.26 11351 1648.7
## - rad      1    481.09 11560 1657.9
## - nox      1    520.87 11600 1659.6
## - ptratio  1   1200.23 12279 1688.4
## - dis      1   1352.26 12431 1694.6
## - rm       1   1959.55 13038 1718.8
## - lstat    1   2718.88 13798 1747.4
##
## Step:  AIC=1636.48
## medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio +
##     black + lstat
##
##           Df Sum of Sq   RSS    AIC
## <none>                 11081 1636.5
## - chas     1    227.21 11309 1640.5
## - crim     1    245.37 11327 1641.3
## - zn       1    257.82 11339 1641.9
## - black    1    270.82 11352 1642.5
## - tax      1    273.62 11355 1642.6
## - rad      1    500.92 11582 1652.6
## - nox      1    541.91 11623 1654.4
## - ptratio  1   1206.45 12288 1682.5
## - dis      1   1448.94 12530 1692.4
## - rm       1   1963.66 13045 1712.8
## - lstat    1   2723.48 13805 1741.5

##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat, data = Boston)
##
## Coefficients:
## (Intercept)         crim           zn         chas          nox
##   36.341145    -0.108413     0.045845     2.718716   -17.376023
##          rm          dis          rad          tax      ptratio
##    3.801579    -1.492711     0.299608    -0.011778    -0.946525
##       black        lstat
##    0.009291    -0.522553

#(a)the modelP-values< 2.2e-16, so from the P value, it was a good model.
#for the variables selected based on P-value<0.05,
#choose all the varible except"indus" and "age"
#AIC for regression model was 1589.64 ,BIC for it was 1648.81.
#Adj R sqaure was 0.7338

#(b)
hist(Boston$medv)
```




```
#for the boston dataset, from the histogram, the medv variable was kind of ri
ght-skewness,
response_var<-log(Boston$medv)

lmbos2<-lm(response_var~crim+zn+indus+chas+nox+rm+age+dis+rad+tax+ptratio+bla
ck+lstat,data=Boston)
step(lmbos2,direction = "backward")

## Start:  AIC=-1667.19
## response_var ~ crim + zn + indus + chas + nox + rm + age + dis +
##     rad + tax + ptratio + black + lstat
##
##           Df Sum of Sq    RSS     AIC
## - age      1    0.0057 17.755 -1669.0
## - indus    1    0.0362 17.786 -1668.2
## <none>                 17.749 -1667.2
## - zn       1    0.1643 17.914 -1664.5
## - chas     1    0.3088 18.058 -1660.5
## - black    1    0.5339 18.283 -1654.2
## - tax      1    0.6235 18.373 -1651.7
## - nox      1    0.9351 18.684 -1643.2
## - rad      1    1.0413 18.791 -1640.3
## - rm       1    1.0637 18.813 -1639.7
## - dis      1    1.3639 19.113 -1631.7
## - ptratio  1    1.9270 19.676 -1617.0
## - crim     1    2.1995 19.949 -1610.1
## - lstat    1    7.3809 25.130 -1493.2
```

```
##
## Step:  AIC=-1669.03
## response_var ~ crim + zn + indus + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat
##
##           Df Sum of Sq    RSS     AIC
## - indus    1    0.0363 17.791 -1670.0
## <none>                 17.755 -1669.0
## - zn       1    0.1593 17.914 -1666.5
## - chas     1    0.3138 18.069 -1662.2
## - black    1    0.5431 18.298 -1655.8
## - tax      1    0.6205 18.376 -1653.7
## - nox      1    0.9645 18.720 -1644.3
## - rad      1    1.0356 18.791 -1642.3
## - rm       1    1.1452 18.900 -1639.4
## - dis      1    1.5471 19.302 -1628.8
## - ptratio  1    1.9224 19.677 -1619.0
## - crim     1    2.1988 19.954 -1612.0
## - lstat    1    8.1949 25.950 -1479.0
##
## Step:  AIC=-1670
## response_var ~ crim + zn + chas + nox + rm + dis + rad + tax +
##     ptratio + black + lstat
##
##           Df Sum of Sq    RSS     AIC
## <none>                 17.791 -1670.0
## - zn       1    0.1451 17.936 -1667.9
## - chas     1    0.3399 18.131 -1662.4
## - black    1    0.5344 18.326 -1657.0
## - tax      1    0.6139 18.405 -1654.8
## - nox      1    0.9350 18.726 -1646.1
## - rad      1    1.0088 18.800 -1644.1
## - rm       1    1.1171 18.909 -1641.2
## - dis      1    1.7385 19.530 -1624.8
## - ptratio  1    1.8862 19.678 -1621.0
## - crim     1    2.2229 20.014 -1612.4
## - lstat    1    8.1604 25.952 -1481.0

##
## Call:
## lm(formula = response_var ~ crim + zn + chas + nox + rm + dis +
##     rad + tax + ptratio + black + lstat, data = Boston)
##
## Coefficients:
## (Intercept)         crim           zn         chas          nox
##   4.0836823   -0.0103187    0.0010874    0.1051484   -0.7217440
##          rm          dis          rad          tax      ptratio
##   0.0906728   -0.0517059    0.0134457   -0.0005579   -0.0374259
##       black        lstat
##   0.0004127   -0.0286039
```

```
step(lmbos2,direction = "backward",k=log(506))

## Start:  AIC=-1608.02
## response_var ~ crim + zn + indus + chas + nox + rm + age + dis +
##     rad + tax + ptratio + black + lstat
##
##           Df Sum of Sq    RSS     AIC
## - age      1    0.0057 17.755 -1614.1
## - indus    1    0.0362 17.786 -1613.2
## - zn       1    0.1643 17.914 -1609.6
## <none>                 17.749 -1608.0
## - chas     1    0.3088 18.058 -1605.5
## - black    1    0.5339 18.283 -1599.2
## - tax      1    0.6235 18.373 -1596.8
## - nox      1    0.9351 18.684 -1588.3
## - rad      1    1.0413 18.791 -1585.4
## - rm       1    1.0637 18.813 -1584.8
## - dis      1    1.3639 19.113 -1576.8
## - ptratio  1    1.9270 19.676 -1562.1
## - crim     1    2.1995 19.949 -1555.1
## - lstat    1    7.3809 25.130 -1438.3
##
## Step:  AIC=-1614.09
## response_var ~ crim + zn + indus + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat
##
##           Df Sum of Sq    RSS     AIC
## - indus    1    0.0363 17.791 -1619.3
## - zn       1    0.1593 17.914 -1615.8
## <none>                 17.755 -1614.1
## - chas     1    0.3138 18.069 -1611.5
## - black    1    0.5431 18.298 -1605.1
## - tax      1    0.6205 18.376 -1602.9
## - nox      1    0.9645 18.720 -1593.5
## - rad      1    1.0356 18.791 -1591.6
## - rm       1    1.1452 18.900 -1588.7
## - dis      1    1.5471 19.302 -1578.0
## - ptratio  1    1.9224 19.677 -1568.3
## - crim     1    2.1988 19.954 -1561.2
## - lstat    1    8.1949 25.950 -1428.3
##
## Step:  AIC=-1619.28
## response_var ~ crim + zn + chas + nox + rm + dis + rad + tax +
##     ptratio + black + lstat
##
##           Df Sum of Sq    RSS     AIC
## - zn       1    0.1451 17.936 -1621.4
## <none>                 17.791 -1619.3
## - chas     1    0.3399 18.131 -1615.9
## - black    1    0.5344 18.326 -1610.5
```

```
## - tax      1    0.6139 18.405 -1608.3
## - nox      1    0.9350 18.726 -1599.6
## - rad      1    1.0088 18.800 -1597.6
## - rm       1    1.1171 18.909 -1594.7
## - dis      1    1.7385 19.530 -1578.3
## - ptratio  1    1.8862 19.678 -1574.5
## - crim     1    2.2229 20.014 -1565.9
## - lstat    1    8.1604 25.952 -1434.5
##
## Step:  AIC=-1621.4
## response_var ~ crim + chas + nox + rm + dis + rad + tax + ptratio +
##     black + lstat
##
##           Df Sum of Sq    RSS     AIC
## <none>                 17.936 -1621.4
## - chas     1    0.3388 18.275 -1618.2
## - tax      1    0.5229 18.459 -1613.1
## - black    1    0.5386 18.475 -1612.7
## - rad      1    0.9601 18.897 -1601.2
## - nox      1    1.0250 18.961 -1599.5
## - rm       1    1.2650 19.201 -1593.1
## - dis      1    1.6967 19.633 -1581.9
## - crim     1    2.1377 20.074 -1570.7
## - ptratio  1    2.5632 20.500 -1560.0
## - lstat    1    8.1516 26.088 -1438.1

##
## Call:
## lm(formula = response_var ~ crim + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat, data = Boston)
##
## Coefficients:
## (Intercept)         crim         chas          nox           rm
##   4.1000969   -0.0100763    0.1049848   -0.7515379    0.0954516
##         dis          rad          tax      ptratio        black
##  -0.0442395    0.0130841   -0.0005050   -0.0409840    0.0004143
##       lstat
##  -0.0285881
```

```
summary(lmbos2)
```

```
##
## Call:
## lm(formula = response_var ~ crim + zn + indus + chas + nox +
##     rm + age + dis + rad + tax + ptratio + black + lstat, data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73361 -0.09747 -0.01657  0.09629  0.86435
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.1020423  0.2042726  20.081  < 2e-16 ***
## crim        -0.0102715  0.0013155  -7.808 3.52e-14 ***
## zn           0.0011725  0.0005495   2.134 0.033349 *
## indus        0.0024668  0.0024614   1.002 0.316755
## chas         0.1008876  0.0344859   2.925 0.003598 **
## nox         -0.7783993  0.1528902  -5.091 5.07e-07 ***
## rm           0.0908331  0.0167280   5.430 8.87e-08 ***
## age          0.0002106  0.0005287   0.398 0.690567
## dis         -0.0490873  0.0079834  -6.149 1.62e-09 ***
## rad          0.0142673  0.0026556   5.373 1.20e-07 ***
## tax         -0.0006258  0.0001505  -4.157 3.80e-05 ***
## ptratio     -0.0382715  0.0052365  -7.309 1.10e-12 ***
## black        0.0004136  0.0001075   3.847 0.000135 ***
## lstat       -0.0290355  0.0020299 -14.304  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1899 on 492 degrees of freedom
## Multiple R-squared:  0.7896, Adjusted R-squared:  0.7841
## F-statistic: 142.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

```
#so we use the log of the variable to do the regression
#As the result, the AIC and for regression were -1667 and -1608 respectively,

#adjusted R square was 0.7841
#so we can conclude that when we log our response variable, the regression re
sult was better
#because of the AIC AND BIC were both negative, and higher the R square resul
t.
#for the variable we selected, based on p-value<0.05, the result would be sam
e that "age" and "indus"
#should be not selected.
```

Homework_1-Q2.R

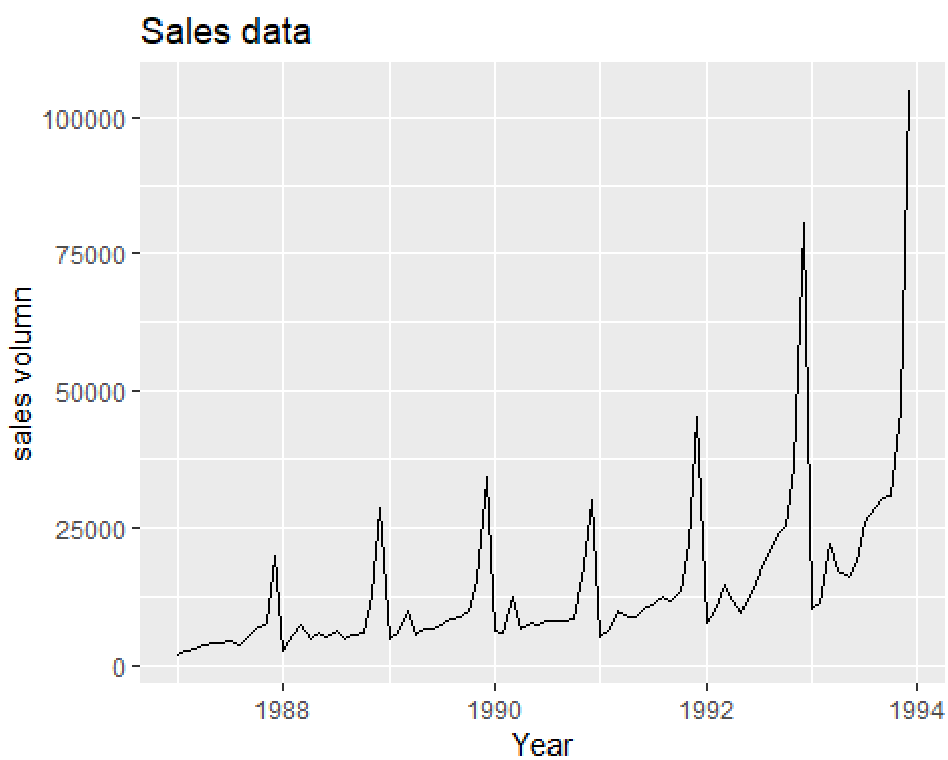
zeshi yang

Thu Feb 07 22:27:44 2019

```
#install.packages("fpp")
library(fpp)

## Warning: package 'fpp' was built under R version 3.5.2

## Loading required package: forecast

## Warning: package 'forecast' was built under R version 3.5.2

## Loading required package: fma

## Warning: package 'fma' was built under R version 3.5.2

## Loading required package: expsmooth

## Warning: package 'expsmooth' was built under R version 3.5.2

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: tseries

## Warning: package 'tseries' was built under R version 3.5.2

library(tseries)
fancy

##            Jan       Feb       Mar       Apr       May       Jun       Jul
## 1987   1664.81   2397.53   2840.71   3547.29   3752.96   3714.74   4349.61
## 1988   2499.81   5198.24   7225.14   4806.03   5900.88   4951.34   6179.12
## 1989   4717.02   5702.63   9957.58   5304.78   6492.43   6630.80   7349.62
## 1990   5921.10   5814.58  12421.25   6369.77   7609.12   7224.75   8121.22
## 1991   4826.64   6470.23   9638.77   8821.17   8722.37  10209.48  11276.55
## 1992   7615.03   9849.69  14558.40  11587.33   9332.56  13082.09  16732.78
## 1993  10243.24  11266.88  21826.84  17357.33  15997.79  18601.53  26155.15
##            Aug       Sep       Oct       Nov       Dec
## 1987   3566.34   5021.82   6423.48   7600.60  19756.21
```

```
## 1988   4752.15   5496.43   5835.10  12600.08  28541.72
## 1989   8176.62   8573.17   9690.50  15151.84  34061.01
## 1990   7979.25   8093.06   8476.70  17914.66  30114.41
## 1991  12552.22  11637.39  13606.89  21822.11  45060.69
## 1992  19888.61  23933.38  25391.35  36024.80  80721.71
## 1993  28586.52  30505.41  30821.33  46634.38 104660.67
```

```r
#(a) Produce a time plot of the data and describe the patterns
#in the graph. Identify any unusual or unexpected fluctuations in the time
series.
autoplot(fancy, xlab="Year", ylab="sales volumn", main="Sales data")
```




```r
#FROM the chare, it can be seen that a seasonal pattern that every March, it
will be a
#increase of sales, and the December always peak. the sales increaseWith the
time going on.

#(b) Use R function "tslm" to fit a regression model to the logarithms
#of these sales data with a linear trend and seasonal component.

#log of sales data
fancy_log <- log(fancy)

season_dummy <- rep(0, length(fancy))

# Create seasonal Dummy Variable
season_dummy[seq_along(season_dummy)%%12 == 3] <- 1
```

```
#Skip first year March
season_dummy[3] <- 0

#transfer seasonal dummy to time series data
season_dummy <- ts(season_dummy, frequency = 12, start = c(1987,1))
fancy_data <- data.frame(fancy_log, season_dummy)
#generate the time series model
tslm_fancy <- tslm(fancy_log~trend + season + season_dummy, data =
fancy_data)
summary(tslm_fancy)#show the statistic results

##
## Call:
## tslm(formula = fancy_log ~ trend + season + season_dummy, data =
fancy_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33673 -0.12757  0.00257  0.10911  0.37671
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.6196670  0.0742471 102.626  < 2e-16 ***
## trend        0.0220198  0.0008268  26.634  < 2e-16 ***
## season2      0.2514168  0.0956790   2.628 0.010555 *
## season3      0.2660828  0.1934044   1.376 0.173275
## season4      0.3840535  0.0957075   4.013 0.000148 ***
## season5      0.4094870  0.0957325   4.277 5.88e-05 ***
## season6      0.4488283  0.0957647   4.687 1.33e-05 ***
## season7      0.6104545  0.0958039   6.372 1.71e-08 ***
## season8      0.5879644  0.0958503   6.134 4.53e-08 ***
## season9      0.6693299  0.0959037   6.979 1.36e-09 ***
## season10     0.7473919  0.0959643   7.788 4.48e-11 ***
## season11     1.2067479  0.0960319  12.566  < 2e-16 ***
## season12     1.9622412  0.0961066  20.417  < 2e-16 ***
## season_dummy 0.5015151  0.1964273   2.553 0.012856 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.179 on 70 degrees of freedom
## Multiple R-squared:  0.9567, Adjusted R-squared:  0.9487
## F-statistic:   119 on 13 and 70 DF,  p-value: < 2.2e-16

#from the results, we can see that Multiple R-square was 0.9567,
#Adjusted R-squared was 0.9487



#(c) Use multiple regression with trend variable and seasonal dummy variables
```

```
to
#redo the regression as shown in the lecture example.
#Check to see that you obtain the same results as tslm.

#x1 is the trend
x1=1:length(fancy)
#x2-x12 are seasonal dummy variables
x2=rep(c(0,1,0,0,0,0,0,0,0,0,0,0),7)
x3=rep(c(0,0,1,0,0,0,0,0,0,0,0,0),7)
x4=rep(c(0,0,0,1,0,0,0,0,0,0,0,0),7)
x5=rep(c(0,0,0,0,1,0,0,0,0,0,0,0),7)
x6=rep(c(0,0,0,0,0,1,0,0,0,0,0,0),7)
x7=rep(c(0,0,0,0,0,0,1,0,0,0,0,0),7)
x8=rep(c(0,0,0,0,0,0,0,1,0,0,0,0),7)
x9=rep(c(0,0,0,0,0,0,0,0,1,0,0,0),7)
x10=rep(c(0,0,0,0,0,0,0,0,0,1,0,0),7)
x11=rep(c(0,0,0,0,0,0,0,0,0,0,1,0),7)
x12=rep(c(0,0,0,0,0,0,0,0,0,0,0,1),7)
#create the multiple regresion
fancy_lm<-lm(fancy_log~x1+x2+x3+x4+x5+x6+x7+x8+x9+x10+x11+x12)
#show the statistic results
summary(fancy_lm)

##
## Call:
## lm(formula = fancy_log ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 +
##     x9 + x10 + x11 + x12)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41644 -0.12619  0.00608  0.11389  0.38567
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.6058604  0.0768740  98.939  < 2e-16 ***
## x1          0.0223930  0.0008448  26.508  < 2e-16 ***
## x2          0.2510437  0.0993278   2.527 0.013718 *
## x3          0.6952066  0.0993386   6.998 1.18e-09 ***
## x4          0.3829341  0.0993565   3.854 0.000252 ***
## x5          0.4079944  0.0993817   4.105 0.000106 ***
## x6          0.4469625  0.0994140   4.496 2.63e-05 ***
## x7          0.6082156  0.0994534   6.116 4.69e-08 ***
## x8          0.5853524  0.0995001   5.883 1.21e-07 ***
## x9          0.6663446  0.0995538   6.693 4.27e-09 ***
## x10         0.7440336  0.0996148   7.469 1.61e-10 ***
## x11         1.2030164  0.0996828  12.068  < 2e-16 ***
## x12         1.9581366  0.0997579  19.629  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.1858 on 71 degrees of freedom
## Multiple R-squared:  0.9527, Adjusted R-squared:  0.9447
## F-statistic: 119.1 on 12 and 71 DF,  p-value: < 2.2e-16
```

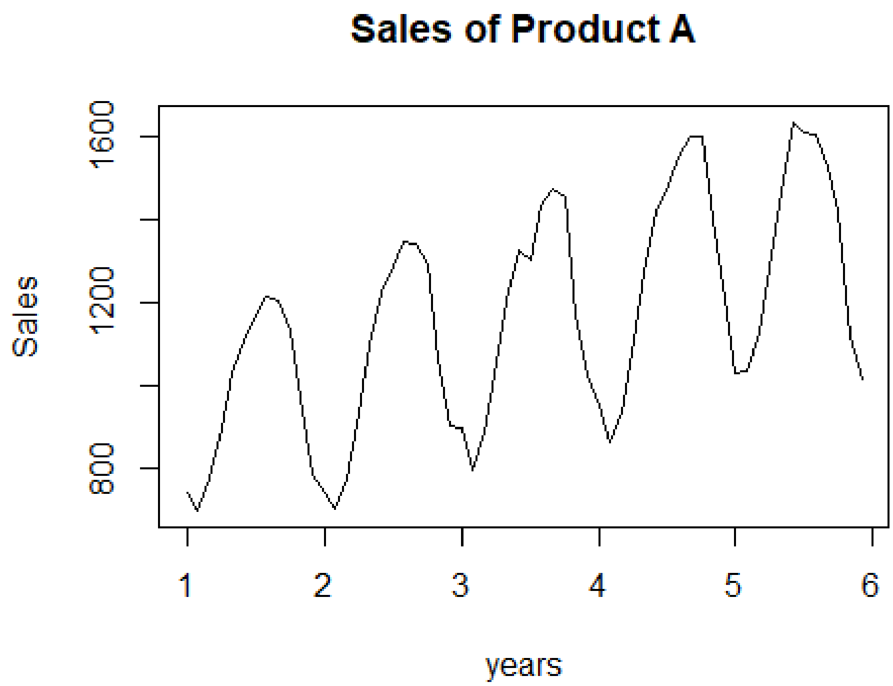
```
#WE can see the Multiple R-squared:  0.9527,    Adjusted R-squared:  0.9447
#which is almost the same result with the tslm regression.
```

Homework_1-q3.R

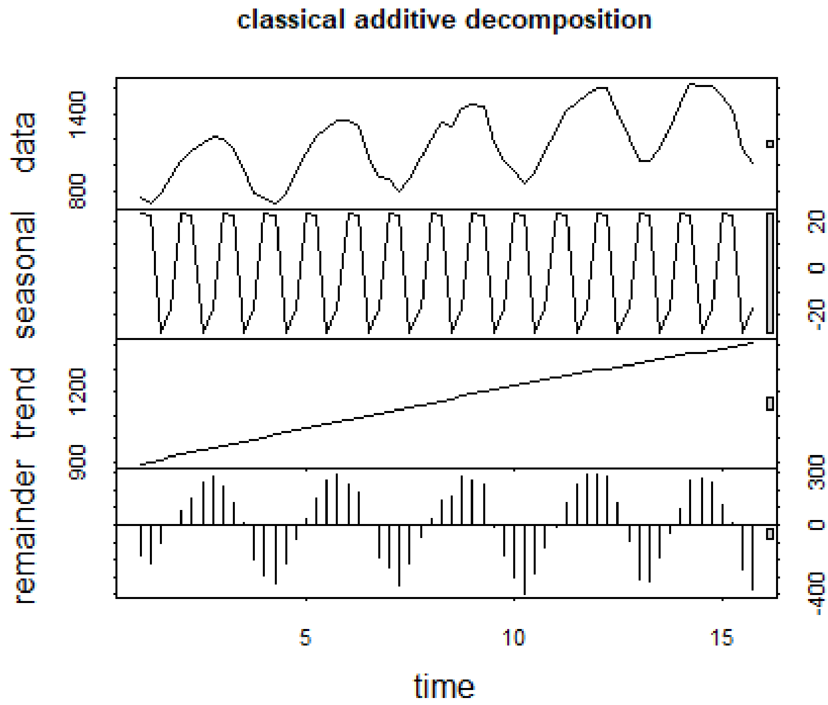
zeshi yang

Thu Feb 07 23:39:44 2019

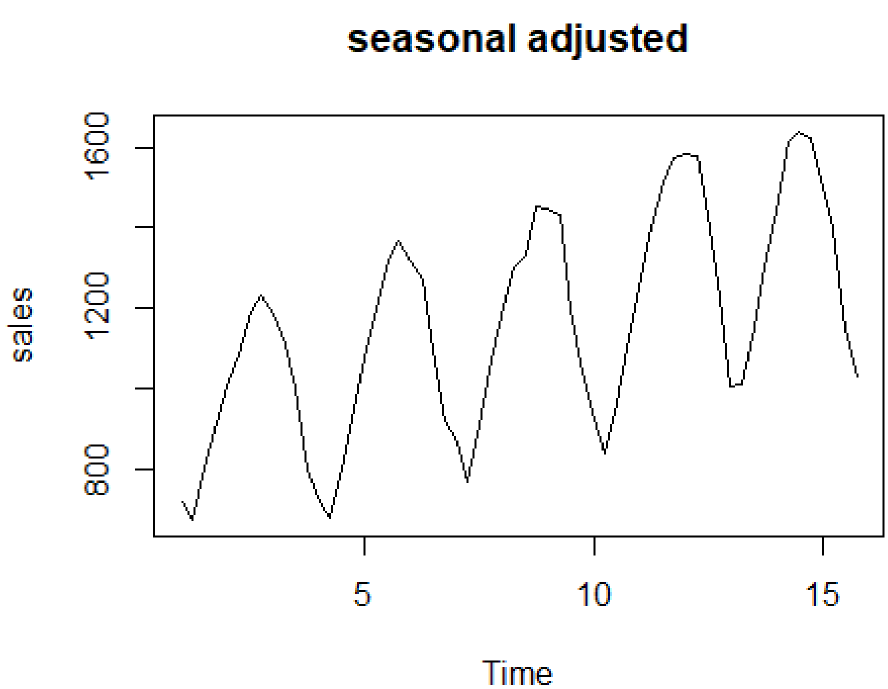
```
library(fpp)

## Warning: package 'fpp' was built under R version 3.5.2

## Loading required package: forecast

## Warning: package 'forecast' was built under R version 3.5.2

## Loading required package: fma

## Warning: package 'fma' was built under R version 3.5.2

## Loading required package: expsmooth

## Warning: package 'expsmooth' was built under R version 3.5.2

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: tseries

## Warning: package 'tseries' was built under R version 3.5.2

library(tseries)
head(plastics)

##    Jan  Feb  Mar  Apr  May  Jun
## 1  742  697  776  898 1030 1107

#(a) Plot the time series of sales of product A. Can you
#identify seasonal fluctuations or a trend?

plot(plastics,xlab="years",ylab="Sales", main="Sales of Product A")
```



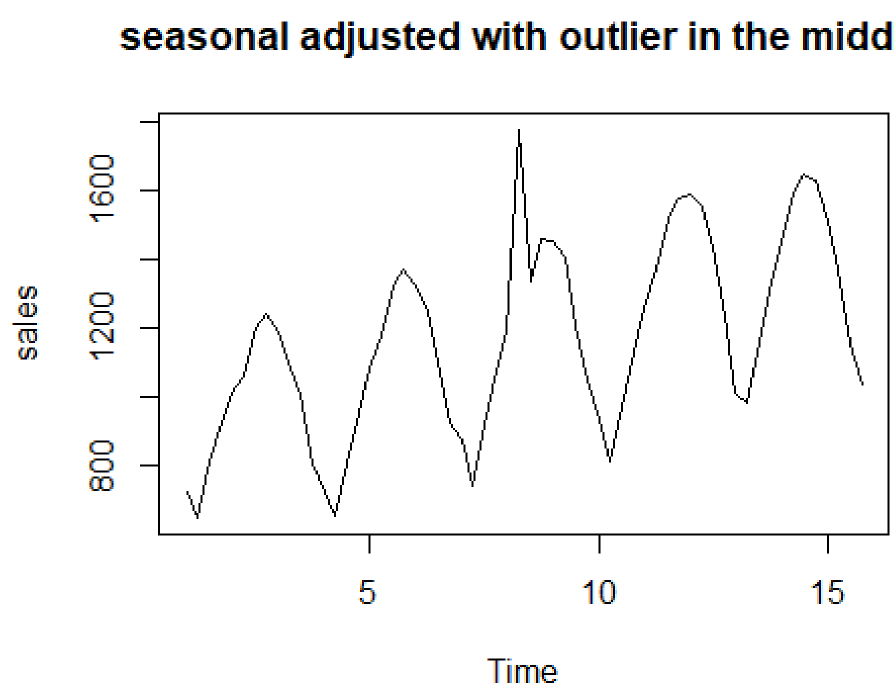

```
#(b) Perform a classical additive decomposition using "stl" function. Plot
out the decomposition for s.window="periodic", t.window=50.
ts_pl<-ts(plastics, frequency = 4)#requirement for additive decomponent
p_model<- stl(ts_pl, t.window=50, s.window="periodic")
plot(p_model,main = "classical additive decomposition")
```



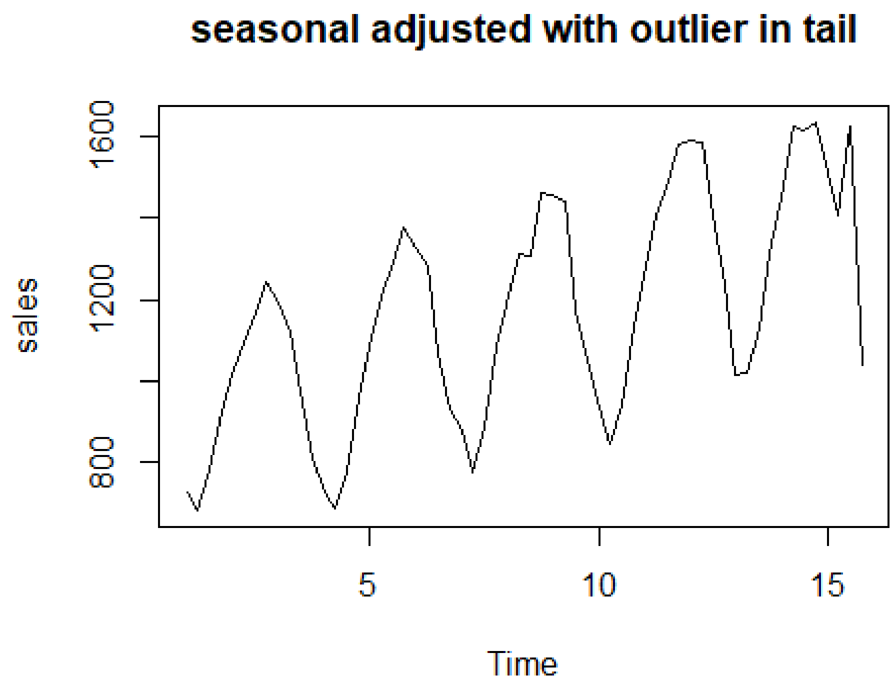

```r
#(c) Compute and plot the seasonally adjusted data for s.window="periodic",
t.window=50.
plot(seasadj(p_model),ylab="sales",main="seasonal adjusted")
```



```
#(d) Change one observation to be an outlier (pick one data point and add 500
to its value. For instance, if you picked July of the third year, the current
value is 1303,then the modified value will be 1803) and recompute the
seasonally adjusted data. What is the effect of the outlier. Again, you need
to do this for s.window="periodic", t.window=50.

plastic2 <- plastics
plastic2[30] = plastic2[30] + 500#choose  the data point =30,which is the
middle.
#decompose again using additive decomposition
p_de2 <- ts(plastic2, frequency = 4)
p_model2<-stl(p_de2, t.window=50, s.window="periodic")
plot(seasadj(p_model2),ylab="sales",main="seasonal adjusted with outlier in
the middle")
```




```
#Does it make any difference if the outlier is near the end rather than in
the middle of the time series? Try it out.

#to anwser this question, add outlier to the data point in the tail to see
how it affects
p3<- plastics
p3[59] = p3[59] + 500#choose  the data point =59,which is the middle.
#decompose again using additive decomposition
p_de3 <- ts(p3, frequency = 4)
p_model3<-stl(p_de3, t.window=50, s.window="periodic")
plot(seasadj(p_model3),ylab="sales",main="seasonal adjusted with outlier in
tail")
```




```
#Conclusion: there is difference if the outlier is near the end rather than
in the middle of the time series, if there was an outlier increase in that
month, so the adjusted seasonal chart observes a spike in that month.
```