

# Assignment3

LingzhouAo

4/1/2018

## load data

```
library(shiny)
library(readr)
A2010 <- read.csv("bp appre 2010.csv", header = TRUE, stringsAsFactors = FALSE)

A2017 <- read.csv("bp appre 2017.csv", header = TRUE, stringsAsFactors = FALSE)

A2000.2017 <- read.csv("PB monthly summaries.csv", header = TRUE, stringsAsFactors = FALSE)
```

## clean data of BP Apprehensions 2010

```
rownames(A2010) <- A2010[,1]

A2010 <- subset(A2010, select = -c(Sector))

A2010 <- rbind(A2010, colSums(A2010))

rownames(A2010) <- c(rownames(A2010)[-length(rownames(A2010))], "Total")
```

## cbind rowSums to dataframd

```
A2010 <- cbind(A2010, rowSums(A2010))
```

## rename last column "Totals

```
colnames(A2010) <- c(colnames(A2010)[-length(colnames(A2010))], "Total")
```

## clean data of BP Apprehensions 2017

```
rownames(A2017) <- A2017[,1]

A2017 <- subset(A2017, select = -c(Sector))

A2017 <- rbind(A2017, colSums(A2017))

rownames(A2017) <- c(rownames(A2017)[-length(rownames(A2017))], "Total")
```

## cbind rowSums to dataframd

```
A2017 <- cbind(A2017,rowSums(A2017))
```

## rename last column "Totals"

```
colnames(A2017) <- c(colnames(A2017)[-length(colnames(A2017))], "Total")
```

## statistics

```
A2010_sector_mean <- apply(A2010[,1:12],1,mean)
A2010_sector_mean
```

##	Big Bend	Del Rio	El Centro	El Paso
##	440.6667	1224.5000	2713.5000	1020.9167
##	Laredo	Rio Grande Valley	San Diego	Tucson
##	2940.5833	4980.5000	5713.7500	17683.5000
##	Yuma	Total		
##	593.0000	37310.9167		

```
A2010_month_mean <- apply(A2010[1:9,],2,mean)
A2010_month_mean
```

##	October	November	December	January	February	March	April
##	4543.333	3646.111	2781.556	3864.889	4754.444	6817.889	6137.444
##	May	June	July	August	September	Total	
##	5227.222	3661.667	2845.444	2935.000	2532.889	49747.889	

```
A2017_sector_mean <- apply(A2017[,1:12],1,mean)
A2017_sector_mean
```

##	Big Bend	Del Rio	El Centro	El Paso
##	500.1667	1123.0000	1552.7500	2099.4167
##	Laredo	Rio Grande Valley	San Diego	Tucson
##	2121.6667	11463.5000	2173.8333	3221.4167
##	Yuma	Total		
##	1070.5833	25326.3333		

```
A2017_month_mean <- apply(A2017[1:9,],2,mean)
A2017_month_mean
```

##	October	November	December	January	February	March	April
##	5131.556	5245.667	4805.667	3508.444	2083.778	1355.000	1236.333
##	May	June	July	August	September	Total	
##	1613.222	1787.444	2020.778	2476.444	2504.111	33768.444	

```
sector<-cbind(A2010_sector_mean,A2017_sector_mean)
```

The maximum sector in 2010 is Tucson and the maximum sector in 2017 is Tio Grande Valley

```
sector2010<-as.data.frame(t(A2010[1:9,]))
as.character(colnames(sector2010))
```

```
## [1] "Big Bend"      "Del Rio"      "El Centro"
## [4] "El Paso"      "Laredo"      "Rio Grande Valley"
## [7] "San Diego"    "Tucson"      "Yuma"
```

```
sector2010
```

```
##           Big Bend Del Rio El Centro El Paso Laredo Rio Grande Valley
## October           530    1119     2589    1007    2613              4236
## November          421     897     2412     894    2130              3688
## December          373     697     2196     725    1802              2987
## January           433    1234     2688    1124    2526              3658
## February          484    1245     2836    1140    3173              4845
## March             660    1874     4408    1528    4433              7141
## April            575    1791     3419    1359    4528              7139
## May              493    1718     3126    1380    3813              7477
## June             415    1326     2440    1005    3475              5595
## July             280     767     2331     725    1857              3832
## August           295    1095     2075     732    2819              5329
## September        329     931     2042     632    2118              3839
## Total            5288   14694    32562   12251   35287              59766
##           San Diego Tucson Yuma
## October          5017   23197   582
## November         4738   16986   649
## December         4636   10907   711
## January          6413   16122   586
## February         6982   21266   819
## March            9061   31197  1059
## April            7115   28579   732
## May             5858   22572   608
## June            5092   13160   447
## July            5113   10303   401
## August          4528    9280   262
## September       4012    8633   260
## Total          68565  212202  7116
```

```
sector2017<-as.data.frame(t(A2017[1:9,]))
as.character(colnames(sector2017))
```

```
## [1] "Big Bend"      "Del Rio"      "El Centro"
## [4] "El Paso"      "Laredo"      "Rio Grande Valley"
## [7] "San Diego"    "Tucson"      "Yuma"
```

```
sector2017
```

```
##           Big Bend Del Rio El Centro El Paso Laredo Rio Grande Valley
## October           697    2106     2441    3973    3350              22642
## November          603    1880     1850    4105    3194              24686
## December          477    1817     1870    3948    2460              23418
## January           473    1243     1796    2779    2265              15580
## February          383    1104     1196    1575    1710              7855
```

```
## March      357      746      871      978      1256      4147
## April      413      589      849      906      1304      3942
## May        552      740      1134     1032     1722      4882
## June       378      761      1280     1180     1839      5817
## July       492      760      1478     1395     2120      7107
## August     563      798      1880     1782     2143      8650
## September  614      932      1988     1540     2097      8836
## Total      6002     13476     18633     25193     25460     137562
##           San Diego Tucson  Yuma
## October    2934     5924   2117
## November   2947     5912   2034
## December   3099     4303   1859
## January    2927     3357   1156
## February   1808     2589    534
## March      1356     2148    336
## April      1392     1487    245
## May        1724     2199    534
## June       1652     2632    548
## July       1764     2177    894
## August     2241     2913   1318
## September  2242     3016   1272
## Total      26086    38657  12847
```

```
t.test(sector2010$Tucson,sector2017$`Rio Grande Valley`)
```

```
##
## Welch Two Sample t-test
##
## data: sector2010$Tucson and sector2017$`Rio Grande Valley`
## t = 0.63546, df = 20.738, p-value = 0.5321
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -26125.62 49091.78
## sample estimates:
## mean of x mean of y
## 32646.46 21163.38
```

```
month<-cbind(A2010_month_mean,A2017_month_mean)
```

**The maximum three months in 2010 are March, April and May**

```
month2010<-as.data.frame((A2010[1:9,6:8]))
as.character(rownames(month2010))
```

```
## [1] "Big Bend"      "Del Rio"      "El Centro"
## [4] "El Paso"       "Laredo"       "Rio Grande Valley"
## [7] "San Diego"     "Tucson"      "Yuma"
```

```
month2010
```

```
##           March April  May
## Big Bend      660   575  493
## Del Rio       1874  1791 1718
## El Centro     4408  3419 3126
```

```
## El Paso          1528  1359  1380
## Laredo           4433  4528  3813
## Rio Grande Valley 7141  7139  7477
## San Diego        9061  7115  5858
## Tucson           31197 28579 22572
## Yuma             1059   732   608
```

```
month2010$sum<-apply(month2010,1,sum)
```

The maximum three months in 2017 are October, November and December

```
month2017<-as.data.frame((A2017[1:9,1:3]))
as.character(rownames(month2017))
```

```
## [1] "Big Bend"      "Del Rio"        "El Centro"
## [4] "El Paso"       "Laredo"         "Rio Grande Valley"
## [7] "San Diego"     "Tucson"         "Yuma"
```

```
month2017
```

```
##           October November December
## Big Bend      697       603       477
## Del Rio       2106      1880      1817
## El Centro     2441      1850      1870
## El Paso       3973      4105      3948
## Laredo        3350      3194      2460
## Rio Grande Valley 22642  24686  23418
## San Diego     2934      2947      3099
## Tucson        5924      5912      4303
## Yuma          2117      2034      1859
```

```
month2017$sum<-apply(month2017,1,sum)
t.test(month2010$sum,month2017$sum)
```

```
##
## Welch Two Sample t-test
##
## data: month2010$sum and month2017$sum
## t = 0.2727, df = 15.542, p-value = 0.7887
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -20374.89 26374.22
## sample estimates:
## mean of x mean of y
## 18182.56 15182.89
```

```
new_A2000.2017 <- apply(t(A2000.2017[,2:13]),1,rev)
ts <- as.vector(t(new_A2000.2017))
ts
```

```
## [1] 91410 76196 71252 185979 211328 220063 180050 166296 115093 113956
## [11] 114312 97744 82632 67709 55081 125090 152229 170580 142813 122927
## [21] 89131 83602 84648 59276 37812 32506 31501 79793 95724 126992
## [31] 121921 97424 78655 76661 82557 68263 61792 47731 37824 86925
```

```
## [41] 96869 98399 75359 88690 75530 79284 84486 72176 65391 57894
## [51] 43614 92521 110669 154981 135468 118726 94590 92165 93246 80017
## [61] 75913 65135 48406 93020 113775 143048 140062 115823 90786 94954
## [71] 96733 93741 83557 70975 52673 101195 125046 160696 126538 105450
## [81] 68366 59641 59751 58084 60713 51594 40527 71934 79268 114137
## [91] 104465 88504 71338 66782 59795 49581 51339 42209 31802 59028
## [101] 73483 89770 91566 69233 53854 49472 48541 44708 42938 32780
## [111] 25947 44502 49211 67342 58493 50884 46044 43843 43522 35359
## [121] 40890 32815 25034 34784 42790 61361 55237 47045 32955 25609
## [131] 26415 22796 26165 22405 19429 23926 28786 42014 36251 31236
## [141] 27166 23170 24166 22863 25612 23368 18983 25714 31579 42218
## [151] 40628 36966 30669 26978 27567 26591 28929 27636 23243 26921
## [161] 35042 47293 48212 43856 34436 33230 33797 31802 35312 31896
## [171] 29528 28668 36403 49596 51502 60683 57862 40708 31388 25825
## [181] 26450 24641 25019 21514 24376 29791 29750 31576 29303 28388
## [191] 30239 30286 32724 32838 37014 23758 26072 33316 38089 40337
## [201] 34450 33723 37048 39501 46184 47211 43251 31576 18754 12195
## [211] 11127 14519 16087 18187 22288 22537
```

```
ts2 <- ts(ts,frequency = 13,start = c(2000,10))
ts2
```

```
## Time Series:
```

```
## Start = c(2000, 10)
```

```
## End = c(2017, 4)
```

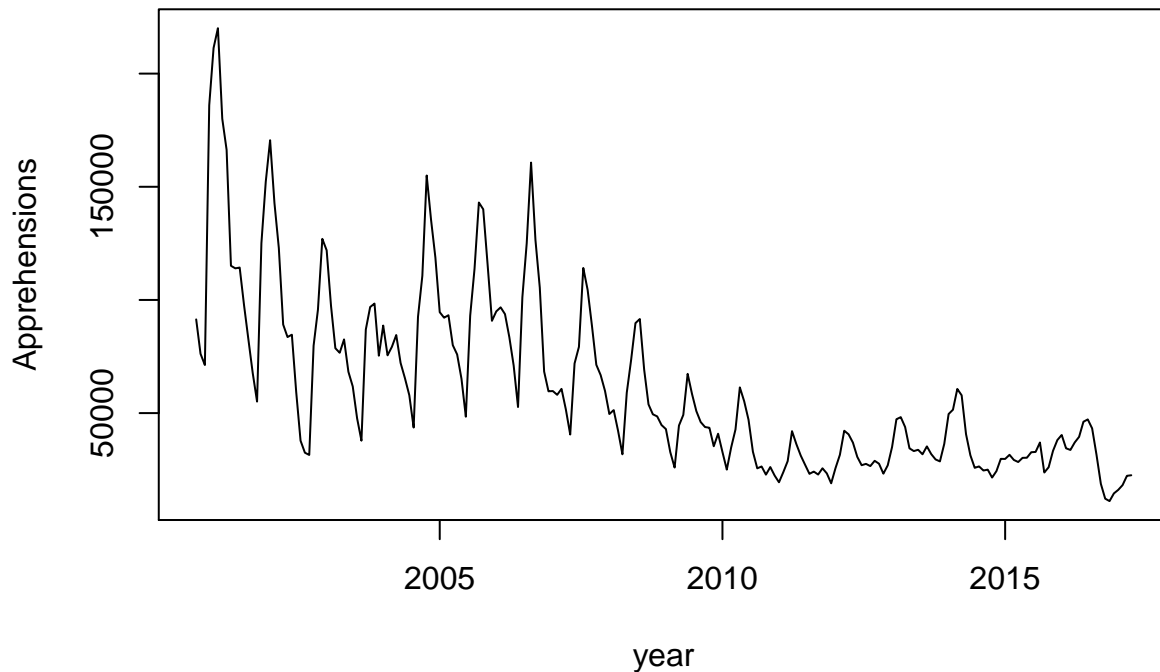
```
## Frequency = 13
```

```
## [1] 91410 76196 71252 185979 211328 220063 180050 166296 115093 113956
## [11] 114312 97744 82632 67709 55081 125090 152229 170580 142813 122927
## [21] 89131 83602 84648 59276 37812 32506 31501 79793 95724 126992
## [31] 121921 97424 78655 76661 82557 68263 61792 47731 37824 86925
## [41] 96869 98399 75359 88690 75530 79284 84486 72176 65391 57894
## [51] 43614 92521 110669 154981 135468 118726 94590 92165 93246 80017
## [61] 75913 65135 48406 93020 113775 143048 140062 115823 90786 94954
## [71] 96733 93741 83557 70975 52673 101195 125046 160696 126538 105450
## [81] 68366 59641 59751 58084 60713 51594 40527 71934 79268 114137
## [91] 104465 88504 71338 66782 59795 49581 51339 42209 31802 59028
## [101] 73483 89770 91566 69233 53854 49472 48541 44708 42938 32780
## [111] 25947 44502 49211 67342 58493 50884 46044 43843 43522 35359
## [121] 40890 32815 25034 34784 42790 61361 55237 47045 32955 25609
## [131] 26415 22796 26165 22405 19429 23926 28786 42014 36251 31236
## [141] 27166 23170 24166 22863 25612 23368 18983 25714 31579 42218
## [151] 40628 36966 30669 26978 27567 26591 28929 27636 23243 26921
## [161] 35042 47293 48212 43856 34436 33230 33797 31802 35312 31896
## [171] 29528 28668 36403 49596 51502 60683 57862 40708 31388 25825
## [181] 26450 24641 25019 21514 24376 29791 29750 31576 29303 28388
## [191] 30239 30286 32724 32838 37014 23758 26072 33316 38089 40337
## [201] 34450 33723 37048 39501 46184 47211 43251 31576 18754 12195
## [211] 11127 14519 16087 18187 22288 22537
```

```
new_A2000.2017_mean <- apply(new_A2000.2017,1,mean)
new_A2000.2017_mean
```

```
## [1] 136973.25 102976.50 77484.08 75422.08 94940.17 97616.33 89331.00
## [8] 71553.17 58750.42 45072.08 37310.92 27298.08 29739.42 34533.08
## [15] 39947.58 27611.08 34072.50 25326.33
```

```
ts.plot(ts2, gpars=list(xlab="year", ylab="Apprehensions", lty=c(1:3)))
```



## shiny app

```
options(shiny.sanitize.errors = FALSE)
ui <- fluidPage(
  titlePanel("Assignment3 Plots"),

  sidebarLayout(position = "left",
    sidebarPanel("Check box",
      checkboxInput("comparebysector",
        "compare by sector",
        value = T),
      checkboxInput("comparebymonth",
        "compare by month",
        value = T),
      checkboxInput("timeseries",
        "time series",
        value = T)
    ),
    mainPanel("Main panel",
      fluidRow(
        splitLayout(cellWidths = c("50%", "50%"),
          plotOutput("graph1"),
          plotOutput("graph2"))
      ),
      tabsetPanel(

        tabPanel("Compare by sector", plotOutput("graph3")),
        tabPanel("Compare by month", plotOutput("graph4")),
```

```

        tabPanel("time series", plotOutput("graph5")),
        tabPanel("T-Test", textOutput("ttest")),

        p()
    )

    )

))

server <- function(input, output) {
  set.seed(1234)
  pt3 <- reactive({
    if(input$comparebysector){
      return(barplot(t(sector),
                     beside = TRUE,
                     legend.text = c("2010", "2017"),
                     main = "compare by sector"))
    }
    else{
      return(NULL)
    }
  })

  pt4 <- reactive({
    if(input$comparebymonth){
      return(barplot(t(month),
                     beside = TRUE,
                     legend.text = c("2010", "2017"),
                     main = "compare by month"))
    }
    else{
      return(NULL)
    }
  })

  pt5 <- reactive({
    if(input$timeseries){
      return(ts.plot(ts2, gpars = list(xlab="year", ylab="appre")))
    }
    else{
      return(NULL)
    }
  })

  output$graph1 <- renderPlot({
    barplot(A2017[1:9, 13], names.arg = rownames(A2017)[1:9],
            las=2,
            axisnames=TRUE,
            main="2017 Border Patrol Apprehensions by Sector",
            border="blue",
            col="red")
  })
}

```



```

})
output$graph2 <- renderPlot({
  barplot(A2010[1:9,13], names.arg = rownames(A2010)[1:9],
    las=2,
    axisnames=TRUE,
    main="2010 Border Patrol Apprehensions by Sector",
    border="blue",
    col="yellow")
})

output$graph3 <- renderPlot({pt3()})
output$graph4 <- renderPlot({pt4()})
output$graph5 <- renderPlot({pt5()})

output$ttest <- renderPrint({
  return(t.test(month2010$sum,month2017$sum))
})
}

```

## Run the application

```
shinyApp(ui = ui, server = server)
```

Shiny applications not supported in static R Markdown documents