Programming Assignment 3

Okky Mabruri

13/09/2020

credit : https://rpubs.com/SLamara

Introduction

The data for this assignment come from the Hospital Compare web site http://hospitalcompare.hhs.gov run by the U.S. Department of Health and Human Services. The purpose of the web site is to provide data and information about the quality of care at over 4,000 Medicare-certified hospitals in the U.S. This dataset essentially covers all major U.S. hospitals. This dataset is used for a variety of purposes, including determining whether hospitals should be fined for not providing high quality care to patients (see http://goo.gl/jAXFX for some background on this particular topic).

The Hospital Compare web site contains a lot of data and we will only look at a small subset for this assignment. The zip file for this assignment contains three files

- outcome-of-care-measures.csv: Contains information about 30-day mortality and readmission rates for heart attacks, heart failure, and pneumonia for over 4,000 hospitals.
- hospital-data.csv: Contains information about each hospital.
- Hospital_Revised_Flat files.pdf: Descriptions of the variables in each file (i.e the code book).

1 Plot the 30-day mortality rates for heart attack

Read the outcome data into R via theread.csv

```
setwd('ProgAssignment3-data/')
hospitaldata <- read.csv('hospital-data.csv')
outcome <-
  read.csv("outcome-of-care-measures.csv", colClasses = "character")
```

1.1 Check outcome data

```
nrow(outcome)
```

```
## [1] 4706
```

```
names(outcome[1:11])
```

```
##  [1] "Provider.Number"
##  [2] "Hospital.Name"
##  [3] "Address.1"
##  [4] "Address.2"
##  [5] "Address.3"
##  [6] "City"
```

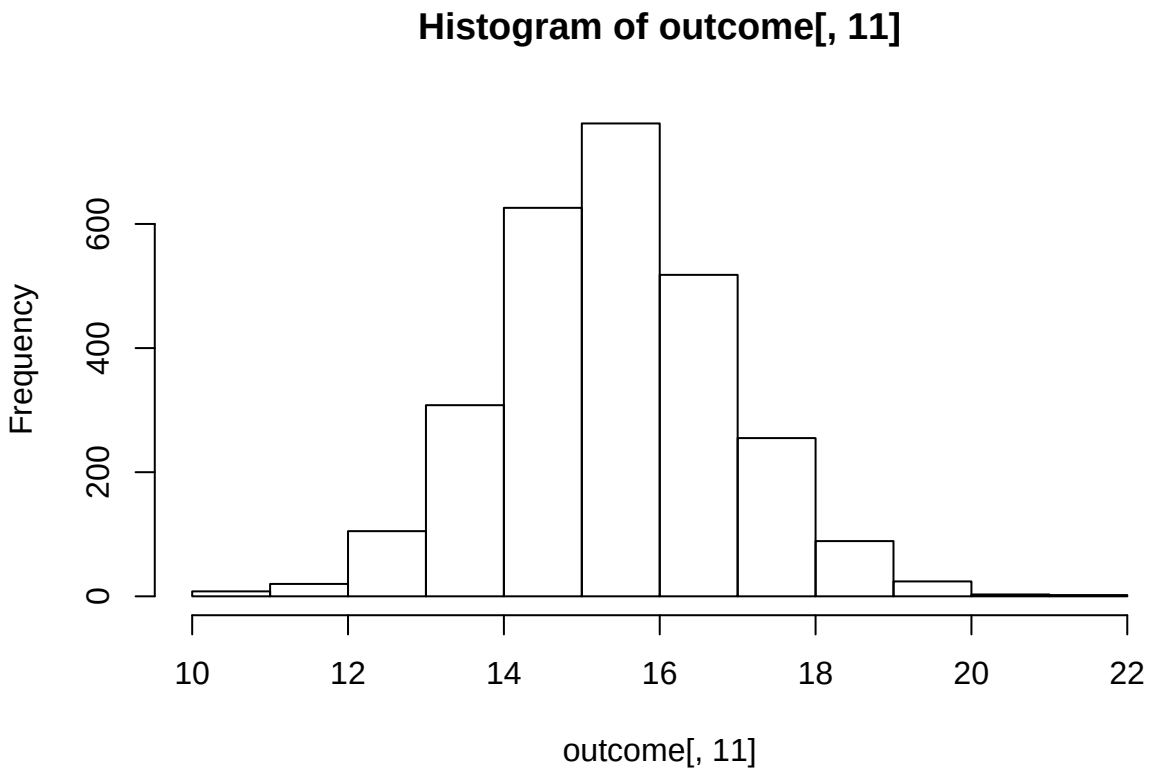
```
##  [7] "State"
##  [8] "ZIP.Code"
##  [9] "County.Name"
## [10] "Phone.Number"
## [11] "Hospital.30.Day.Death..Mortality..Rates.from.Heart.Attack"
```

1.2 Make hist plot

```
outcome[, 11] <- as.numeric(outcome[, 11])
```

```
## Warning: NAs introduced by coercion
```

```
hist(outcome[, 11])
```



2 Finding the best hospital in a state

Handling ties. If there is a tie for the best hospital for a given outcome, then the hospital names should be sorted in alphabetical order and the first hospital in that set should be chosen (i.e. if hospitals “b”, “c”,and “f” are tied for best, then hospital “b” should be returned).The function should use the following template.

```
best <- function(state, outcome) {

  ## Read outcome data
  setwd('ProgAssignment3-data/')

  data <-
    read.csv("outcome-of-care-measures.csv", colClasses = "character")

  ## Check that state and outcome are valid
```

```
  if (!(state %in% data$State)) {
    result <- "invalid state"
  }
  else if (!outcome %in% c("heart attack", "heart failure", "pneumonia")) {
    result <- "invalid outcome"
  }
  else{
    keys <-
      c(
        "heart attack" = 11,
        "heart failure" = 17,
        "pneumonia" = 23
      )

    outcomeKey <- keys[outcome]

    ## Return hospital name in that state with lowest 30-day death rate

    dataPerState <- split(data, data$State)
    dataOurState <- dataPerState[[state]]
    dataOurState <-
      dataOurState[order(dataOurState["Hospital.Name"]),]
    dataOutcome <-
      suppressWarnings(as.numeric(dataOurState[, outcomeKey]))
    good <- complete.cases(dataOutcome)
    dataOutcome <- dataOutcome[good]
    dataOurState <- dataOurState[good, ]
    minimum <- min(dataOutcome)
    index <- match(minimum, dataOutcome)
    result <- dataOurState[index, 2]
  }
  result
}
```

```
best("TX", "heart attack")
```

```
## [1] "CYPRESS FAIRBANKS MEDICAL CENTER"
```

```
best("TX", "heart failure")
```

```
## [1] "FORT DUNCAN MEDICAL CENTER"
```

```
best("MD", "heart attack")
```

```
## [1] "JOHNS HOPKINS HOSPITAL, THE"
```

```
best("MD", "pneumonia")
```

```
## [1] "GREATER BALTIMORE MEDICAL CENTER"
```

```
best("BB", "heart attack")
```

```
## [1] "invalid state"
```

```
best("NY", "hert attack")
```

```
## [1] "invalid outcome"
```

3 Ranking hospitals by outcome in a state

Write a function called rank hospital that takes three arguments: the 2-character abbreviated name of a state (state), an outcome (outcome), and the ranking of a hospital in that state for that outcome (num).The function reads the outcome-of-care-measures.csv file and returns a character vector with the name of the hospital that has the ranking specified by the num argument. For example, the call

```
rankhospital("MD", "heart failure", 5)
```

would return a character vector containing the name of the hospital with the 5th lowest 30-day death rate for heart failure. The num argument can take values “best”, “worst”, or an integer indicating the ranking(smaller numbers are better). If the number given by num is larger than the number of hospitals in that state, then the function should return NA. Hospitals that do not have data on a particular outcome should be excluded from the set of hospitals when deciding the rankings.

Handling ties. It may occur that multiple hospitals have the same 30-day mortality rate for a given causeof death. In those cases ties should be broken by using the hospital name.

Note that Cypress Fairbanks Medical Center and Detar Hospital Navarro both have the same 30-day rate(8.7). However, because Cypress comes before Detar alphabetically, Cypress is ranked number 3 in this scheme and Detar is ranked number 4. One can use the order function to sort multiple vectors in this manner (i.e. where one vector is used to break ties in another vector).

```
rankhospital <- function(state, outcome, num = "best") {
  ## Read outcome data

  setwd('ProgAssignment3-data/')

  data <-
    read.csv("outcome-of-care-measures.csv", colClasses = "character")

  ## Check that state and outcome are valid

  if (!(state %in% data$State)) {
    result <- "invalid state"
  }
  else if (!outcome %in% c("heart attack", "heart failure", "pneumonia")) {
    result <- "invalid outcome"
  }
  else {
    keys <-
      c(
        "heart attack" = 11,
        "heart failure" = 17,
        "pneumonia" = 23
      )
    outcomeKey <- keys[outcome]


    ## Return hospital name in that state with the given rank
    ## 30-day death rate

    dataPerState <- split(data, data$State)
    dataOurState <- dataPerState[[state]]
    dataOutcome <-
      suppressWarnings(as.numeric(dataOurState[, outcomeKey]))
```

```
    good <- complete.cases(dataOutcome)
    dataOutcome <- dataOutcome[good]
    dataOurState <- dataOurState[good, ]
    dataOurState <-
      dataOurState[order(dataOutcome, dataOurState["Hospital.Name"]), ]
    if (grepl("^[0-9]+$", num)) {
      if (as.numeric(num) > length(dataOutcome)) {
        result <- NA
      }
      else {
        result <- dataOurState[as.numeric(num), "Hospital.Name"]
      }
    }
    else if (num == "best") {
      result <- dataOurState[1, "Hospital.Name"]
    }
    else if (num == "worst") {
      result <- dataOurState[length(dataOutcome), "Hospital.Name"]
    }
    else
      result <- NA
  }
  result
}
```

```
rankhospital("TX", "heart failure", 4)
```

```
## [1] "DETAR HOSPITAL NAVARRO"
```

```
rankhospital("MD", "heart attack", "worst")
```

```
## [1] "HARFORD MEMORIAL HOSPITAL"
```

```
rankhospital("MN", "heart attack", 5000)
```

```
## [1] NA
```

4 Ranking hospitals in all states

I implement a function rankAll which takes as arguments the outcome name (outcome) and hospital ranking (num) and returns a 2-column data frame containing the hospital in each state that has the ranking specified in num.

The function returns a value for every state (some may be NA). The first column in the data frame contains the hospital name and the second one contains the 2-character abbreviation for the state name. Hospitals that do not have data on a particular outcome are excluded from the set of hospitals when deciding the rankings.

Although it is possible to call the rankHospital function from the previous section, I decided, for didactic purposes, not using it.

```
rankall <- function(outcome, num = "best") {
  setwd('ProgAssignment3-data/')

  dataAll <-
    data.frame(hospital = character(), state = character())
```

```r
## Read outcome data

data <-
  read.csv("outcome-of-care-measures.csv", colClasses = "character")

## Check that outcome and num are valid

if (!outcome %in% c("heart attack", "heart failure", "pneumonia")) {
  dataAll <- "invalid outcome"
}
else {
  keys <-
    c(
      "heart attack" = 11,
      "heart failure" = 17,
      "pneumonia" = 23
    )
  outcomeKey <- keys[outcome]

  ## For each state, find the hospital of the given rank

  dataPerState <- split(data, data$State)
  for (stat in names(dataPerState)) {
    dataOurState <- dataPerState[[stat]]
    dataOutcome <-
      suppressWarnings(as.numeric(dataOurState[, outcomeKey]))
    good <- complete.cases(dataOutcome)
    dataOutcome <- dataOutcome[good]
    dataOurState <- dataOurState[good, ]
    dataOurState <-
      dataOurState[order(dataOutcome, dataOurState["Hospital.Name"]),]

    if (num == "best") {
      numState <- c(1)
    } else {
      if (num == "worst") {
        numState <- length(dataOutcome)
      } else {
        numState <- num
      }
    }

    dataPart <-
      data.frame(hospital = dataOurState[numState, "Hospital.Name"],
                 state = stat,
                 row.names = stat)
    dataAll <- rbind(dataAll, dataPart)
  }
}

## Return a data frame with the hospital names and the (abbreviated) state name

dataAll
```

```
}
```

Testing rankAll

```
head(rankall("heart attack", 20), 10)
```

```
##                                hospital state
## AK                                    <NA>    AK
## AL      D W MCMILLAN MEMORIAL HOSPITAL    AL
## AR   ARKANSAS METHODIST MEDICAL CENTER    AR
## AZ JOHN C LINCOLN DEER VALLEY HOSPITAL    AZ
## CA               SHERMAN OAKS HOSPITAL    CA
## CO            SKY RIDGE MEDICAL CENTER    CO
## CT             MIDSTATE MEDICAL CENTER    CT
## DC                                    <NA>    DC
## DE                                    <NA>    DE
## FL      SOUTH FLORIDA BAPTIST HOSPITAL    FL
```

```
tail(rankall("pneumonia", "worst"), 3)
```

```
##                                      hospital state
## WI MAYO CLINIC HEALTH SYSTEM - NORTHLAND, INC    WI
## WV                      PLATEAU MEDICAL CENTER    WV
## WY           NORTH BIG HORN HOSPITAL DISTRICT    WY
```

```
tail(rankall("heart failure"), 10)
```

```
##                                                          hospital state
## TN                         WELLMONT HAWKINS COUNTY MEMORIAL HOSPITAL    TN
## TX                                       FORT DUNCAN MEDICAL CENTER    TX
## UT VA SALT LAKE CITY HEALTHCARE - GEORGE E. WAHLEN VA MEDICAL CENTER    UT
## VA                                         SENTARA POTOMAC HOSPITAL    VA
## VI                            GOV JUAN F LUIS HOSPITAL & MEDICAL CTR    VI
## VT                                             SPRINGFIELD HOSPITAL    VT
## WA                                       HARBORVIEW MEDICAL CENTER    WA
## WI                                  AURORA ST LUKES MEDICAL CENTER    WI
## WV                                       FAIRMONT GENERAL HOSPITAL    WV
## WY                                     CHEYENNE VA MEDICAL CENTER    WY
```

Programming Assignment 3: Quiz

```
best("SC", "heart attack")
```

```
## [1] "MUSC MEDICAL CENTER"
```

```
best("NY", "pneumonia")
```

```
## [1] "MAIMONIDES MEDICAL CENTER"
```

```
best("AK", "pneumonia")
```

```
## [1] "YUKON KUSKOKWIM DELTA REG HOSPITAL"
```

```
rankhospital("NC", "heart attack", "worst")
```

```
## [1] "WAYNE MEMORIAL HOSPITAL"
```

```
rankhospital("WA", "heart attack", 7)
```

```
## [1] "YAKIMA VALLEY MEMORIAL HOSPITAL"
```

```
rankhospital("TX", "pneumonia", 10)
```

```
## [1] "SETON SMITHVILLE REGIONAL HOSPITAL"
```

```
rankhospital("NY", "heart attack", 7)
```

```
## [1] "BELLEVUE HOSPITAL CENTER"
```

```
r <- rankall("heart attack", 4)
as.character(subset(r, state == "HI")$hospital)
```

```
## [1] "CASTLE MEDICAL CENTER"
```

```
r <- rankall("pneumonia", "worst")
as.character(subset(r, state == "NJ")$hospital)
```

```
## [1] "BERGEN REGIONAL MEDICAL CENTER"
```

```
r <- rankall("heart failure", 10)
as.character(subset(r, state == "NV")$hospital)
```

```
## [1] "RENOWN SOUTH MEADOWS MEDICAL CENTER"
```