Analyzing hospitals

Arin Parsa

6/7/2020

Contents

Hospitals Data set

"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. 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_Flatfiles.pdf: Descriptions of the variables in each file (i.e the code book)." (source: R Programming course, Week 3 assignment by JHU)

Unzip data file

```
unzip("hospitals.zip")
```

Load outcomes.csv and explore

```
outcomes <- read.csv("data/outcome-of-care-measures.csv")

names(outcomes)
```

```
##  [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"
## [12] "Comparison.to.U.S..Rate...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Attack"
## [13] "Lower.Mortality.Estimate...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Attack"
## [14] "Upper.Mortality.Estimate...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Attack"
## [15] "Number.of.Patients...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Attack"
## [16] "Footnote...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Attack"
## [17] "Hospital.30.Day.Death..Mortality..Rates.from.Heart.Failure"
## [18] "Comparison.to.U.S..Rate...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Failure"
## [19] "Lower.Mortality.Estimate...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Failure"
## [20] "Upper.Mortality.Estimate...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Failure"
## [21] "Number.of.Patients...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Failure"
## [22] "Footnote...Hospital.30.Day.Death..Mortality..Rates.from.Heart.Failure"
## [23] "Hospital.30.Day.Death..Mortality..Rates.from.Pneumonia"
## [24] "Comparison.to.U.S..Rate...Hospital.30.Day.Death..Mortality..Rates.from.Pneumonia"
## [25] "Lower.Mortality.Estimate...Hospital.30.Day.Death..Mortality..Rates.from.Pneumonia"
## [26] "Upper.Mortality.Estimate...Hospital.30.Day.Death..Mortality..Rates.from.Pneumonia"
## [27] "Number.of.Patients...Hospital.30.Day.Death..Mortality..Rates.from.Pneumonia"
## [28] "Footnote...Hospital.30.Day.Death..Mortality..Rates.from.Pneumonia"
## [29] "Hospital.30.Day.Readmission.Rates.from.Heart.Attack"
## [30] "Comparison.to.U.S..Rate...Hospital.30.Day.Readmission.Rates.from.Heart.Attack"
## [31] "Lower.Readmission.Estimate...Hospital.30.Day.Readmission.Rates.from.Heart.Attack"
## [32] "Upper.Readmission.Estimate...Hospital.30.Day.Readmission.Rates.from.Heart.Attack"
## [33] "Number.of.Patients...Hospital.30.Day.Readmission.Rates.from.Heart.Attack"
## [34] "Footnote...Hospital.30.Day.Readmission.Rates.from.Heart.Attack"
## [35] "Hospital.30.Day.Readmission.Rates.from.Heart.Failure"
## [36] "Comparison.to.U.S..Rate...Hospital.30.Day.Readmission.Rates.from.Heart.Failure"
## [37] "Lower.Readmission.Estimate...Hospital.30.Day.Readmission.Rates.from.Heart.Failure"
## [38] "Upper.Readmission.Estimate...Hospital.30.Day.Readmission.Rates.from.Heart.Failure"
## [39] "Number.of.Patients...Hospital.30.Day.Readmission.Rates.from.Heart.Failure"
## [40] "Footnote...Hospital.30.Day.Readmission.Rates.from.Heart.Failure"
## [41] "Hospital.30.Day.Readmission.Rates.from.Pneumonia"
## [42] "Comparison.to.U.S..Rate...Hospital.30.Day.Readmission.Rates.from.Pneumonia"
## [43] "Lower.Readmission.Estimate...Hospital.30.Day.Readmission.Rates.from.Pneumonia"
## [44] "Upper.Readmission.Estimate...Hospital.30.Day.Readmission.Rates.from.Pneumonia"
## [45] "Number.of.Patients...Hospital.30.Day.Readmission.Rates.from.Pneumonia"
## [46] "Footnote...Hospital.30.Day.Readmission.Rates.from.Pneumonia"
```

```
dim(outcomes)
```

```
## [1] 4706   46
```

```
nrow(outcomes)
```

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

```
ncol(outcomes)
```

```
## [1] 46
```

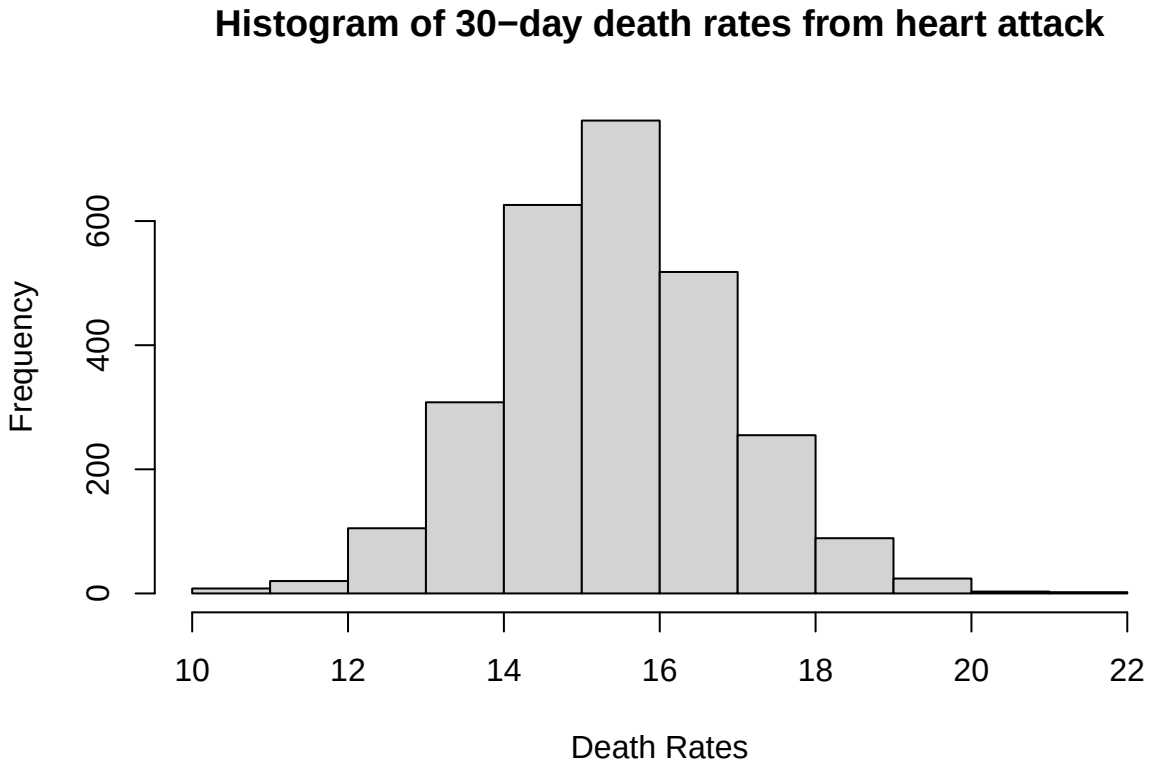
```
#summary(outcomes)
#str(outcomes)
#head(outcomes)
```

Histogram of 30-day death rates from heart attack

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

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

```
hist(outcomes[, 11], xlab = "Death Rates", main = "Histogram of 30-day death rates from heart attack")
```



Find the best hospital in state

A function called best that take two arguments: the 2-character abbreviated name of a state and an outcome name.

The function reads the outcome-of-care-measures.csv and returns a character vector with the name of the hospital that has the best (i.e. lowest) 30-day mortality for the specified outcome in that state.

The hospital name is the name provided in the Hospital.Name variable. The outcomes can be one of "heart attack", heart failure", or pneumonia". Hospitals that do not have data on a particular outcome should be excluded from the set of hospitals when deciding the rankings.

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

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

  ## Read outcome data
  outcome_df <- read.csv("data/outcome-of-care-measures.csv", colClasses = "character")
```

```r
  ## Check that state and outcome are valid
  if(!(state %in% outcome_df$State)) {
    stop('Invalid State')
  }

  if(!(outcome %in% c("heart attack", "heart failure", "pneumonia"))) {
    stop('Invalid Outcome')
  }

  if(outcome == "heart attack") {
    col <- 11
  }
  if(outcome == "heart failure") {
    col <- 17
  }
  if(outcome == "pneumonia") {
    col <- 23
  }
  ## Create a data frame of the hospitals in the state we want and the outcome for the condition we wan
  hospital_df <- outcome_df[outcome_df$State == state, c(2, col)]

  ## Redefine hospital_df names to be simpler
  names(hospital_df) <- c("Hospital", "Mortality")

  ## Change column class to numeric (Note: It will also coerce non-numerics into N/As )
  hospital_df$Mortality <- as.numeric(hospital_df$Mortality)

  ## Remove N/As
  hospital_df <- na.omit(hospital_df)

  ## Order the hospital names alphabetically
  hospital_df <- hospital_df[order(hospital_df$Hospital),]

  ## Return the hospital that has the lowest 30-day mortality rate
  best_hospital <- hospital_df[which.min(hospital_df$Mortality), 1]
  return(best_hospital)
}

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

```
## Warning in best("TX", "heart attack"): NAs introduced by coercion
## [1] "CYPRESS FAIRBANKS MEDICAL CENTER"
```

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

```
## Warning in best("TX", "heart failure"): NAs introduced by coercion
## [1] "FORT DUNCAN MEDICAL CENTER"
```

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

```
## Warning in best("MD", "heart attack"): NAs introduced by coercion
## [1] "JOHNS HOPKINS HOSPITAL, THE"
```

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

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

```r
# best("BB", "heart attack")
# best("NY", "hert attack")
```

Ranking hospitals by outcome in a state

A function called rankhospital 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 and returns a character vector with the name of the hospital that has the ranking specified by the num argument.

The num argument can take values "best", "worst", or an integer indicating the ranking. 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. It may occur that multiple hospitals have the same 30-day mortality rate for a given cause of death. In those cases ties should be broken by using the hospital name.

```r
rankhospital <- function(state, outcome, num = "best") {
  ## Read outcome data
  outcome_df <- read.csv("data/outcome-of-care-measures.csv", colClasses = "character")
  ## Check that state and outcome are valid
  if(!(state %in% outcome_df$State)) {
    stop('Invalid State')
  }

  if(!(outcome %in% c("heart attack", "heart failure", "pneumonia"))) {
    stop('Invalid Outcome')
  }

  if(outcome == "heart attack") {
    col <- 11
  }
  if(outcome == "heart failure") {
    col <- 17
  }
  if(outcome == "pneumonia") {
    col <- 23
  }
  ## Create a data frame of the hospitals in the state we want and the outcome for the condition we wan
  hospital_df <- outcome_df[outcome_df$State == state, c(2, col)]

  ## Redefine hospital_df names to be simpler
  names(hospital_df) <- c("Hospital", "Mortality")

  ## Change column class to numeric (Note: It will also coerce non-numerics into N/As )
  hospital_df$Mortality <- as.numeric(hospital_df$Mortality)

  ## Remove N/As
  hospital_df <- na.omit(hospital_df)

  ## Order the hospital names alphabetically
  hospital_df <- hospital_df[order(hospital_df$Hospital),]
```

```
  ## Order the Mortality column from least to greatest
  hospital_df <- hospital_df[order(hospital_df$Mortality),]

  ## Creating Rank column
  hospital_df <- cbind(hospital_df, c(1:nrow(hospital_df)))
  colnames(hospital_df)[3] <- "Rank"

  ## Return hospital name in that state with the given rank
  if (num == "best") {
    return(hospital_df[hospital_df$Rank == 1, 1])
  }

  if (num == "worst") {
    return(hospital_df[hospital_df$Rank == nrow(hospital_df), 1])
  }

  if (num > nrow(hospital_df)) {
    return("NA")
  }

  return(hospital_df[hospital_df$Rank == num, 1])
}

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

```
## Warning in rankhospital("TX", "heart failure", 4): NAs introduced by coercion
```

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

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

```
## Warning in rankhospital("MD", "heart attack", "worst"): NAs introduced by
## coercion
```

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

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

```
## Warning in rankhospital("MN", "heart attack", 5000): NAs introduced by coercion
```

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

Ranking hospitals in all states

A function called rankall that takes two arguments: an outcome name (outcome) and a hospital ranking (num). The function reads the outcome-of-care-measures.csv and returns a 2-column data frame containing the hospital in each state that has the ranking specified in num. For example the function call rankall("heart attack", "best") would return a data frame containing the names of the hospitals that are the best in their respective states for 30-day heart attack death rates. The function should return a value for every state (some may be NA). The first column in the data frame is named hospital, which contains the hospital name, and the second column is named state, which contains the 2-character abbreviation for the state name.

Hospitals that do not have data on a particular outcome should be excluded from the set of hospitals when deciding the rankings.

If an invalid outcome value is passed to rankall, the function should throw an error via the stop function with the exact message invalid outcome". The num variable 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.

```
rankall <- function(outcome, num = "best") {
  ## Read outcome data
  outcome_df <- read.csv("data/outcome-of-care-measures.csv", colClasses = "character")
  ## Check that outcome is valid
  if(!(outcome %in% c("heart attack", "heart failure", "pneumonia"))) {
    stop('Invalid Outcome')
  }

  if(outcome == "heart attack") {
    col <- 11
  }
  if(outcome == "heart failure") {
    col <- 17
  }
  if(outcome == "pneumonia") {
    col <- 23
  }
  ## Create a data frame of the hospitals in the state we want and the outcome for the condition we wan
  hospital_df <- outcome_df[,c(2, 7, col)]

  ## Redefine hospital_df names to be simpler
  names(hospital_df) <- c("hospital", "state", "mortality")

  ## Change column class to numeric (Note: It will also coerce non-numerics into N/As )
  hospital_df$mortality <- as.numeric(hospital_df$mortality)

  ## Remove N/As
  hospital_df <- na.omit(hospital_df)

  ## Order the dataset columns
  hospital_df <- hospital_df[order(hospital_df$state, hospital_df$mortality, hospital_df$hospital),]

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

  ## Create a list that contains the abbreviation for each state
  states <- unique(hospital_df$state)

  ## Create an empty vector for the number of hospitals in a state to be added
  state_rank <- c()

  ## Create an empty vector for the worst hospitals in each state
  worst <- c()

  desired_hospitals <- data.frame(matrix(vector(), 0, 2))
  names(desired_hospitals) <- c("hospital", "state")

  ## Loop through unique states and create the data frame for hospital and state
  ## based on rank (index of the row in the ordered dataframe of state hospitals by mortality
  for (i in states) {
    ## Subset of rows by state
    state_rows <- hospital_df[hospital_df$state == i,]
```

```
    if(num == "best") {
      desired_hospitals <- rbind(desired_hospitals, c(state_rows[1, 1], i))
    }
    else if(num == "worst") {
      desired_hospitals <- rbind(desired_hospitals, c(state_rows[nrow(state_rows), 1],i))
    }
    ## if num is greater than number of hospitals in a state, return NA
    ## although R will coerce them if we don't have this statement, I'm
    ## explicitly coding for sake of clarity
    else if(num > nrow(state_rows)) {
       desired_hospitals <- rbind(desired_hospitals, c("<NA>", i))
     }
    else {
      desired_hospitals <- rbind(desired_hospitals, c(state_rows[num, 1],i))
    }
  }

  names(desired_hospitals) <- c("hospital", "state")

  return(desired_hospitals)

}

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

```
## Warning in rankall("heart attack", 20): NAs introduced by coercion
```

```
##                              hospital state
## 1                                <NA>    AK
## 2      D W MCMILLAN MEMORIAL HOSPITAL    AL
## 3   ARKANSAS METHODIST MEDICAL CENTER    AR
## 4 JOHN C LINCOLN DEER VALLEY HOSPITAL    AZ
## 5               SHERMAN OAKS HOSPITAL    CA
## 6            SKY RIDGE MEDICAL CENTER    CO
```

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

```
## Warning in rankall("pneumonia", "worst"): NAs introduced by coercion
```

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

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

```
## Warning in rankall("heart failure"): NAs introduced by coercion
```

```
##                                                          hospital state
## 45                        WELLMONT HAWKINS COUNTY MEMORIAL HOSPITAL    TN
## 46                                    FORT DUNCAN MEDICAL CENTER    TX
## 47 VA SALT LAKE CITY HEALTHCARE - GEORGE E. WAHLEN VA MEDICAL CENTER    UT
## 48                                     SENTARA POTOMAC HOSPITAL    VA
## 49                          GOV JUAN F LUIS HOSPITAL & MEDICAL CTR    VI
## 50                                        SPRINGFIELD HOSPITAL    VT
## 51                                     HARBORVIEW MEDICAL CENTER    WA
## 52                                AURORA ST LUKES MEDICAL CENTER    WI
```

```
## 53                                  FAIRMONT GENERAL HOSPITAL    WV
## 54                                CHEYENNE VA MEDICAL CENTER    WY
```