1 Plot the 30-day mortality rates for heart attack

Read the outcome data into R via the read.csv function and look at the \_rst few rows.

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

> head(outcome)

There are many columns in this dataset. You can see how many by typing ncol(outcome) (you can see

the number of rows with the nrow function). In addition, you can see the names of each column by typing

names(outcome) (the names are also in the PDF document.

To make a simple histogram of the 30-day death rates from heart attack (column 11 in the outcome dataset),

run

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

> ## You may get a warning about NAs being introduced; that is okay

> hist(outcome[, 11])

Because we originally read the data in as character (by specifying colClasses = "character" we need to

coerce the column to be numeric. You may get a warning about NAs being introduced but that is okay.

There is nothing to submit for this part.

2 Finding the best hospital in a state

Write 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 \_le and returns a character vector

with the name of the hospital that has the best (i.e. lowest) 30-day mortality for the speci\_ed 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.

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 \_rst 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

## Check that state and outcome are valid

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

## rate

}

The function should check the validity of its arguments. If an invalid state value is passed to best, the

function should throw an error via the stop function with the exact message \invalid state". If an invalid

outcome value is passed to best, the function should throw an error via the stop function with the exact

message \invalid outcome".

Here is some sample output from the function.

> source("best.R")

> 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")

Error in best("BB", "heart attack") : invalid state

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

Error in best("NY", "hert attack") : invalid outcome

Save your code for this function to a \_le named best.R.

Use the submit script provided to submit your solution to this part. There are 3 tests that need to be passed

for this part of the assignment.

3 Ranking hospitals by outcome in a state

Write 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 \_le and returns a character vector with the name

of the hospital that has the ranking speci\_ed 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 cause

of death. In those cases ties should be broken by using the hospital name. For example, in Texas (\TX"),

the hospitals with lowest 30-day mortality rate for heart failure are shown here.

> head(texas)

Hospital.Name Rate Rank

3935 FORT DUNCAN MEDICAL CENTER 8.1 1

4085 TOMBALL REGIONAL MEDICAL CENTER 8.5 2

4103 CYPRESS FAIRBANKS MEDICAL CENTER 8.7 3

3954 DETAR HOSPITAL NAVARRO 8.7 4

4010 METHODIST HOSPITAL,THE 8.8 5

3962 MISSION REGIONAL MEDICAL CENTER 8.8 6

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).

The function should use the following template.

rankhospital <- function(state, outcome, num = "best") {

## Read outcome data

## Check that state and outcome are valid

## Return hospital name in that state with the given rank

## 30-day death rate

}

The function should check the validity of its arguments. If an invalid state value is passed to best, the

function should throw an error via the stop function with the exact message \invalid state". If an invalid

outcome value is passed to best, the function should throw an error via the stop function with the exact

message \invalid outcome".

Here is some sample output from the function.

> source("rankhospital.R")

> 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

Save your code for this function to a \_le named rankhospital.R.

Use the submit script provided to submit your solution to this part. There are 4 tests that need to be passed

for this part of the assignment.

4 Ranking hospitals in all states

Write a function called rankall that takes two arguments: an outcome name (outcome) and a hospital rank-

ing (num). The function reads the outcome-of-care-measures.csv \_le and returns a 2-column data frame

containing the hospital in each state that has the ranking speci\_ed 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 \_rst 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.

Handling ties. The rankall function should handle ties in the 30-day mortality rates in the same way

that the rankhospital function handles ties.

The function should use the following template.

rankall <- function(outcome, num = "best") {

## Read outcome data

## Check that state and outcome are valid

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

## Return a data frame with the hospital names and the

## (abbreviated) state name

}

NOTE: For the purpose of this part of the assignment (and for e\_ciency), your function should NOT call

the rankhospital function from the previous section.

The function should check the validity of its arguments. 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.

Here is some sample output from the function.

> source("rankhospital.R")

> 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

Save your code for this function to a \_le named rankhospital.R.

Use the submit script provided to submit your solution to this part. There are 4 tests that need to be passed

for this part of the assignment.

4 Ranking hospitals in all states

Write a function called rankall that takes two arguments: an outcome name (outcome) and a hospital rank-

ing (num). The function reads the outcome-of-care-measures.csv \_le and returns a 2-column data frame

containing the hospital in each state that has the ranking speci\_ed 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 \_rst 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.

Handling ties. The rankall function should handle ties in the 30-day mortality rates in the same way

that the rankhospital function handles ties.

The function should use the following template.

rankall <- function(outcome, num = "best") {

## Read outcome data

## Check that state and outcome are valid

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

## Return a data frame with the hospital names and the

## (abbreviated) state name

}

NOTE: For the purpose of this part of the assignment (and for e\_ciency), your function should NOT call

the rankhospital function from the previous section.

The function should check the validity of its arguments. 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.

Here is some sample output from the function.

> source("rankall.R")

> 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

Save your code for this function to a \_le named rankall.R.

Use the submit script provided to submit your solution to this part. There are 3 tests that need to be passed

for this part of the assignment.