

# Programación Actuarial III

## Caso: Calidad de Hospitales

### Introducción

Deberás descargar el archivo zip de base de datos que contiene la información para este caso. Y extráela en el directorio que servirá como tu directorio de trabajo. Al iniciar R, recuerda cambiar tu directorio de trabajo al directorio en el que extrajiste los archivos.

Los datos de este caso provienen del sitio web de *Hospital Compare* (<http://hospitalcompare.hhs.gov>), administrado por el Departamento de Salud y Servicios Humanos de Estados Unidos (U.S. Department of Health and Human Services). El propósito del sitio es proveer datos e información acerca de la calidad del cuidado en más de 4,000 hospitales certificados por Medicare en USA. Esta base de datos cubre la mayoría de los hospitales de Estados Unidos. Esta base de datos se ocupa para diversos fines, incluido la determinación de si un hospital debe ser multado por no proveer cuidado de calidad a sus pacientes. (vea [http:// goo.gl/JAXFX](http://goo.gl/JAXFX) para más detalles de este tema en particular).

El sitio web contiene una gran cantidad de datos y lo que haremos será buscar en un subconjunto de ellos para este caso. El archivo zip contiene tres archivos:

- **outcome-of-care-measures.csv**: contiene información acerca de la mortalidad por 30 días y las tasas de readmisión en ataques al corazón, fallas cardíacas y neumonía en más de 4,000 hospitales.
- **hospital-data.csv**: contiene información acerca de cada hospital.
- **Hospital\_Revised\_Flatfiles.pdf**: describe las variables de cada archivo. (code book)

Una descripción de las variables se incluye en el archivo PDF llamado *Hospital\_Revised\_Flatfiles.pdf*. Este documento contiene información acerca de muchos otros archivos que no están incluidos en este caso. Se debe prestar atención a las variables de número 19 ("Outcome of Care Measures.csv") y el número 11 ("Hospital\_Data.csv"). Puede ser útil tener a la mano este documento impreso, al menos las páginas de las tablas 11 y 19. En particular, los números de las variables de cada tabla indican el índice que le corresponde a esa variable. (i.e. "Hospital Name" es la columna número 2 del archivo **outcome-of-care-measures.csv**)

### Datos

El archivo se puede descargar de:

<https://www.dropbox.com/s/gntzvdap87tpf9h/Calidad%20de%20Hospitales%20-%20data.zip?dl=0>

#### 1. Graficar la tasa de mortalidad de 30 días por ataque al corazón

Lea la información en R con la función `read.csv` y revise las primeras filas.

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

```
> head(outcome)
```

Hay muchas columnas en esta base. Puedes usar la función `ncol(outcome)` para saber la cantidad de columnas. Además de esto, puedes usar la función `names(outcome)` para saber los nombres de cada columna.

Para hacer un histograma simple de las tasas de mortalidad de 30 días por ataque al corazón (columna 11 de la base `outcome`) debemos correr:

```
> outcome[, 11] <- as.numeric(outcome[, 11])
> ## You may get a warning about NAs being introduced; that is okay
> hist(outcome[, 11])
```

Hasta acá no hay nada que entregar.

## 2. Encontrar el mejor hospital en un estado.

Escribe una función llamada `mejor` que reciba dos argumentos: la abreviatura de dos caracteres del nombre de un estado de USA y el nombre de un resultado (`outcome`). Esta función debe leer el archivo `outcome-of-care-measures.csv` y regresar un vector con el nombre del hospital que tiene la mejor (la más baja) tasa de mortalidad de 30 días para el resultado especificado en ese estado. El nombre del hospital aparece en la variable `Hospital.Name`. Los resultados pueden ser: “ataque al corazón” (heart attack), “falla cardíaca” (heart failure) o “neumonía” (pneumonia). Los hospitales que no tengan información de un resultado en específico, deberán ser excluidos de la clasificación.

Manejo de empates. En el caso de un empate por el hospital con mejor resultado, se muestra el primero en orden alfabético. (por ejemplo, si “b”, “f” y “x” están empatados en el mejor resultado, se muestra únicamente “b”)

La función debe parecerse a esta:

```
mejor <- function(estado, resultado) {
  # Lectura de datos

  # Revisión de la validez de estado y resultado

  # Regresa el nombre del hospital con la tasa más baja de mortalidad de 30 días
}
```

La función debe revisar la validez de sus argumentos. Si un estado inexistente es colocado en la función, la función debe enviar un error mediante la función `stop` que diga “estado inválido”. Si un resultado inválido es colocado en la función, la función debe enviar un error mediante la función `stop` que diga “resultado inválido”.

Algunos resultados que deberías obtener son:

```
> source("mejor.R")
> best("TX", "ataque")
[1] "CYPRESS FAIRBANKS MEDICAL CENTER"

> best("TX", "falla")
[1] "FORT DUNCAN MEDICAL CENTER"

> best("MD", "ataque")
[1] "JOHNS HOPKINS HOSPITAL, THE"
```

```
> best("MD", "neumonia")
[1] "GREATER BALTIMORE MEDICAL CENTER"

> best("BB", "ataque")
Error in best("BB", "ataque") : invalid state

> best("NY", "atakue")
Error in best("NY", "atakue") : invalid outcome
```

Este código debe ser guardado como “mejor.R”.

### 3. Jerarquía de hospitales por resultado en un estado

Escribe una función llamada **rankhospital** que tome 3 argumentos: la abreviatura del nombre del estado (estado), un resultado (resultado) y el ranking de un hospital en ese estado para ese resultado (num). Esta función lee el archivo **outcome-of-care-measures.csv** y regresa un vector de caracteres con el nombre del hospital que tiene el rango especificado por el argumento num. Por ejemplo, la llamada:

```
rankhospital("MD", "falla",5)
```

debe devolver un vector de caracteres que contenga el nombre del 5° hospital con tasa de mortalidad más baja para falla cardiaca. El argumento num puede recibir los valores de “mejor”, “peor” o un entero indicando el ranking (entre más baja la tasa el hospital es mejor). Si el número dado en la función es mayor que la cantidad de hospitales en ese estado, la función debe regresar el valor de NA. Los hospitales que no tienen datos en particular de un resultado deben ser excluidos de la lista de hospitales al decidir los rankings.

Manejo de empates. Puede ser que varios hospitales tengan la misma tasa de mortalidad de 30 días dada una causa de muerte. En esos casos, los empates se deciden usando la regla del nombre de hospital. Por ejemplo, en Texas los hospitales de tasas más bajas son:

```
> 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
```

Observa que Cypress Fairbanks Medical Center y Detar Hospital Navarro tienen ambos la misma tasa (8.7). Sin embargo, Cypress está antes de Detar en el orden alfabético, por ello Cypress tiene posición 3 y Detar 4. Se permite usar la función **order** para ordenar vectores.

La función debe verse de la siguiente manera:

```
rankhospital <- function(estado, resultado, num = "best") {
  # Lectura de datos

  # Revisión de la validez de estado y resultado

  # Regresa el nombre del hospital con el puesto dado de la tasa más
  # baja de mortalidad de 30 días
```

```
}
```

La función debe revisar la validez de sus argumentos. Si un estado inexistente es colocado en la función, la función debe enviar un error mediante la función **stop** que diga “estado inválido”. Si un resultado inválido es colocado en la función, la función debe enviar un error mediante la función **stop** que diga “resultado inválido”.

Algunos resultados que deberías obtener son:

```
> source("rankhospital.R")
> rankhospital("TX", "falla", 4)
[1] "DETAR HOSPITAL NAVARRO"

> rankhospital("MD", "ataque", "peor")
[1] "HARFORD MEMORIAL HOSPITAL"

> rankhospital("MN", "ataque", 5000)
[1] NA
```

Guarde su código como “rankhospital.R”.

#### 4. Ranking de hospitales en todos los estados

Escriba una función llamada **rankingcompleto** que reciba dos argumentos: un resultado (resultado) y un número de clasificación en el ranking (num). La función lee el archivo **outcome-of-care-measures.csv** y regresa un data frame de 2 columnas conteniendo el hospital de cada estado que se encuentre clasificado en esa posición específica. Por ejemplo, la llamada:

```
rankingcompleto ("ataque", "mejor")
```

debe regresar un data frame conteniendo los nombres de los hospitales que resultaron con más baja tasa de mortalidad de 30 días en su estado. Esta función debe regresar un valor para cada estado, en algunos podría ser NA. La primera columna del data frame se llama hospital (contiene los nombres de los hospitales) y la segunda columna se llama estado (contiene la abreviatura de dos caracteres del nombre del estado). Los hospitales que no tienen información acerca de un resultado en particular deben ser excluidos del análisis al decidir la clasificación.

Manejo de empates. La función debe manejar los empates de la misma manera que lo hace rankhospital.

La función debe verse de la siguiente manera:

```
rankingcompleto <- function(resultado, num = "mejor") {
  # Lectura de datos

  # Revisión de la validez de estado y resultado

  # Para cada estado, encuentra el hospital con la posición dada.
  # Regresa un data frame con el nombre del hospital y la abreviatura
  # del nombre del estado al que pertenece.
}
```

Nota: Esta función no debe llamar a la función rankhospital.

La función debe revisar la validez de sus argumentos. Si un estado inexistente es colocado en la función, la función debe enviar un error mediante la función **stop** que diga “estado inválido”. Si un

resultado inválido es colocado en la función, la función debe enviar un error mediante la función **stop** que diga “**resultado inválido**”.

Algunos resultados que deberías obtener son:

```
> source("rankingcompleto.R")
> head(rankall("ataque", 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("neumonia", "peor"), 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("falla"), 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
```

Guarde su código como “rankingcompleto.R”.