

# Carga inicial datos: Physionet 2012 UCI data mortality

*Ricardo Alberich, Irene Garcia*

## Contents

<b>1</b>	<b>Introducción: Physionet 2012 UCI data mortality</b>	<b>1</b>
1.1	Enlaces de interés . . . . .	1
<b>2</b>	<b>Ingesta de datos</b>	<b>1</b>
2.1	Modelo de datos . . . . .	1
2.2	Carga set_a . . . . .	2
2.3	En resumen tenemos . . . . .	3
2.4	Unificar: series, perfiles y scores . . . . .	4

## 1 Introducción: Physionet 2012 UCI data mortality

En el concurso del congreso “Computers in Cardiology” (ahora “Computing in Cardiology”) del año 2012 propuso un caso de estudio como reto: *Predicción de la tasa de mortalidad de los pacientes de una UCI*

Resto de años mas recientes

- <https://physionet.org/content/challenge-2018/>
- <https://physionet.org/content/challenge-2019/>

### 1.1 Enlaces de interés

**HR:** Heart Rate bpm beats per minut

**GCS:** Glasgow Comma Score (scale 3-15)

**RespRate:** Respiration rate (bpm) breaths for one minute

## 2 Ingesta de datos

### 2.1 Modelo de datos

```
# Cargamos los datos
path = "data_basic_physionet/set-a/"# path training
# Creamos un vector con los nombres de los archivos
lista_pacientes_set_a = dir(path) # lista ficheros pacientes
# Printamos número de archivos que leemos
length(lista_pacientes_set_a) # número pacientes en training

## [1] 4000

# Mostramos como ejemplo el nombre del documento 1 de los datos
lista_pacientes_set_a[1]

## [1] "132539.txt"
```

```

data_paciente_132539=read_csv("data_basic_physionet/set-a/132539.txt", col_types =cols(Time=col_time(f
Value=col_double()))
str(data_paciente_132539)
glimpse(data_paciente_132539)
class(data_paciente_132539)
head(data_paciente_132539,30)

```

## 2.2 Carga set\_a

```

# lista path's a cada fichero de paciente
list_files = paste0(path,lista_pacientes_set_a)
# Función leer paciente
# Leemos el tiempo como carácter y después haremos un ajuste que nos lo simplifique todo a minutos.
leer_paciente = function(file){read_csv(file, col_types = cols(Time = col_character(),
                        Parameter = col_character(),
                        Value = col_double())) %>%
# Separamos las horas de los minutos de la columna Time para acto seguido poner una sola columna
# llamada Time_min sólo con los minutos en que se tomaron los datos.
                        separate(Time,into = c("H","M"),sep = ":") %>%
                        mutate(Time_Minutes = as.numeric(H)*60+as.numeric(M)) %>%
                        select(Time_Minutes,Parameter,Value)}

#leer_paciente(list_files[1])
raw_data = lapply(list_files,leer_paciente)# lista de los datos por paciente
#extraer perfiles "RecordID" "Age" "Gender" "Height" "Weight" "ICUType"
perfil = function(data_paciente){
  data_paciente %>%
  filter(Parameter %in% c("RecordID","Age","Gender","Height","ICUType","Weight")) %>%
  select(-Time_Minutes) %>%
  distinct(Parameter,.keep_all = TRUE) %>%
  spread(Parameter,Value) }

## ejemplo
#perfil(data_paciente_132539)
## Guardo todos los datos del perfil de cada paciente
perfiles = lapply(raw_data,perfil) %>%
  bind_rows() %>%
  select(RecordID, Age, Gender, Height,Weight,ICUType)

glimpse(perfiles)

## Observations: 4,000
## Variables: 6
## $ RecordID <dbl> 132539, 132540, 132541, 132543, 132545, 132547, 13254...
## $ Age <dbl> 54, 76, 44, 68, 88, 64, 68, 78, 64, 74, 64, 71, 66, 8...
## $ Gender <dbl> 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, ...
## $ Height <dbl> -1.0, 175.3, -1.0, 180.3, -1.0, 180.3, 162.6, 162.6, ...
## $ Weight <dbl> -1.0, 76.0, 56.7, 84.6, -1.0, 114.0, 87.0, 48.4, 60.7...
## $ ICUType <dbl> 4, 2, 3, 3, 3, 1, 3, 3, 3, 2, 3, 2, 3, 1, 1, 2, 3, 3,...

## Ler series
## se modifica error de time

serie_UCI_parameter <- function(paciente,parameters){ paciente %>%

```

```

arrange(Parameter,Time_Minutes) %>%
filter(Parameter %in% parameters) %>%
add_column(RecordID=paciente[1,3]$Value) }

##ejemplo
parameters = c("HR","RespRate","GCS")
serie_paciente1 = serie_UCI_parameter(raw_data[[1]],parameters)
serie_paciente1

## # A tibble: 92 x 4
##   Time_Minutes Parameter Value RecordID
##   <dbl> <chr> <dbl> <dbl>
## 1         7 GCS      15  132539
## 2       217 GCS      15  132539
## 3       457 GCS      15  132539
## 4       697 GCS      15  132539
## 5       937 GCS      15  132539
## 6      1177 GCS      15  132539
## 7      1417 GCS      15  132539
## 8      1657 GCS      15  132539
## 9      1897 GCS      14  132539
## 10     2137 GCS      15  132539
## # ... with 82 more rows

# paso parámetros y apilo
parameters=c("HR","RespRate","GCS","Bilirubin","NIMAP","MechVent","PaCO2","Cholesterol","TropT","SaO2",
series_parameters = lapply(raw_data,FUN=function(x) serie_UCI_parameter(x,parameters)) %>%
bind_rows()
glimpse(series_parameters)

## Observations: 647,685
## Variables: 4
## $ Time_Minutes <dbl> 7, 217, 457, 697, 937, 1177, 1417, 1657, 1897, 21...
## $ Parameter <chr> "GCS", "GCS", "GCS", "GCS", "GCS", "GCS", "GCS", ...
## $ Value <dbl> 15, 15, 15, 15, 15, 15, 15, 15, 14, 15, 15, 15, 1...
## $ RecordID <dbl> 132539, 132539, 132539, 132539, 132539, 132539, 1...

```

## 2.3 En resumen tenemos

```

#set-a
glimpse(perfiles)

## Observations: 4,000
## Variables: 6
## $ RecordID <dbl> 132539, 132540, 132541, 132543, 132545, 132547, 13254...
## $ Age <dbl> 54, 76, 44, 68, 88, 64, 68, 78, 64, 74, 64, 71, 66, 8...
## $ Gender <dbl> 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1,...
## $ Height <dbl> -1.0, 175.3, -1.0, 180.3, -1.0, 180.3, 162.6, 162.6, ...
## $ Weight <dbl> -1.0, 76.0, 56.7, 84.6, -1.0, 114.0, 87.0, 48.4, 60.7...
## $ ICUType <dbl> 4, 2, 3, 3, 3, 1, 3, 3, 3, 2, 3, 2, 3, 1, 1, 2, 3, 3,...

glimpse(series_parameters)

## Observations: 647,685
## Variables: 3

```

```
## $ Time_min <dbl> 7, 217, 457, 697, 937, 1177, 1417, 1657, 1897, 2137,...
## $ Parameter <chr> "GCS", "GCS", "GCS", "GCS", "GCS", "GCS", "GCS", "GC...
## $ Value <dbl> 15, 15, 15, 15, 15, 15, 15, 15, 14, 15, 15, 15, 15, ...
```

## 2.4 Unificar: series, perfiles y scores

Nos faltan los scores clásicos que se utilizan en las ICU. Estos están en el fichero Outcome-a.txt para el set-a

```
scoresApath = "data_basic_physionet/Outcomes-a.txt"
scoresA = read_csv(scoresApath)
```

```
## Parsed with column specification:
## cols(
##   RecordID = col_double(),
##   `SAPS-I` = col_double(),
##   SOFA = col_double(),
##   Length_of_stay = col_double(),
##   Survival = col_double(),
##   `In-hospital_death` = col_double()
## )
```

```
glimpse(scoresA)
```

```
## Observations: 4,000
## Variables: 6
## $ RecordID <dbl> 132539, 132540, 132541, 132543, 132545, 13...
## $ `SAPS-I` <dbl> 6, 16, 21, 7, 17, 14, 14, 19, 11, 14, 15, ...
## $ SOFA <dbl> 1, 8, 11, 1, 2, 11, 4, 8, 0, 6, 2, 7, 2, 7...
## $ Length_of_stay <dbl> 5, 8, 19, 9, 4, 6, 9, 6, 17, 8, 13, 7, 22,...
## $ Survival <dbl> -1, -1, -1, 575, 918, 1637, -1, 5, 38, -1,...
## $ `In-hospital_death` <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, ...
```

```
Scores_perfilesA = inner_join(perfiles,scoresA,"RecordID")
glimpse(Scores_perfilesA)
```

```
## Observations: 4,000
## Variables: 11
## $ RecordID <dbl> 132539, 132540, 132541, 132543, 132545, 13...
## $ Age <dbl> 54, 76, 44, 68, 88, 64, 68, 78, 64, 74, 64...
## $ Gender <dbl> 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, ...
## $ Height <dbl> -1.0, 175.3, -1.0, 180.3, -1.0, 180.3, 162...
## $ Weight <dbl> -1.0, 76.0, 56.7, 84.6, -1.0, 114.0, 87.0,...
## $ ICUType <dbl> 4, 2, 3, 3, 3, 1, 3, 3, 3, 2, 3, 2, 3, 1, ...
## $ `SAPS-I` <dbl> 6, 16, 21, 7, 17, 14, 14, 19, 11, 14, 15, ...
## $ SOFA <dbl> 1, 8, 11, 1, 2, 11, 4, 8, 0, 6, 2, 7, 2, 7...
## $ Length_of_stay <dbl> 5, 8, 19, 9, 4, 6, 9, 6, 17, 8, 13, 7, 22,...
## $ Survival <dbl> -1, -1, -1, 575, 918, 1637, -1, 5, 38, -1,...
## $ `In-hospital_death` <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, ...
```

### 2.4.1 Extracción factores de las series

genero una tabla con resúmenes de las variables por paciente: media, desviación típica

```
series_summary = series_parameters %>%
  group_by(RecordID,Parameter) %>%
```

```

summarise(count = n(),mean = mean(Value,na.rm = TRUE),
          sd = sd(Value,na.rm=TRUE)) %>%
gather(Stat, Value, count:sd) %>%
ungroup() %>%
transmute(RecordID,ParameterStat = paste0(Parameter,"_",Stat),Value) %>%
spread(ParameterStat, Value)

```

```

data_tidy = Scores_perfilesA %>% inner_join(series_summary)

```

```

## Joining, by = "RecordID"

```

```

hola= series_parameters %>% spread(Parameter,Value)

```

```

import=c()

```

```

chart.Correlation(hola[], histogram=TRUE, pch=19)

```