Analisis-Dataset-Pacientes-Covid19

Dani

28/4/2020

Antes de comenzar con el análisis de los datos de los pacientes con Covid19 se van a introducir las librerías necesarias.

library(xlsx)  
library(dplyr)  
library(ggplot2)  
library(tidyverse)

Se carga el primer dataset bruto de los datos de los pacientes.

archivo <- "C:/Users/Daniel/Desktop/TFG-github/TFG/modelo 1/datos\_usados/anexo datos pacientes/datos/datos\_pacientes.xlsx"  
datos\_pacientes <- read.xlsx(archivo,sheetIndex = 1,header = TRUE)

Primero se van a mostrar las variables que hay en el dataset, de las cuales muchas no nos van a ser necesarias para el modelado.

names(datos\_pacientes)

## [1] "id" "case\_in\_country" "reporting.date"   
## [4] "summary" "location" "country"   
## [7] "gender" "age" "symptom\_onset"   
## [10] "If\_onset\_approximated" "hosp\_visit\_date" "international\_traveler"  
## [13] "domestic\_traveler" "exposure\_start" "exposure\_end"   
## [16] "traveler" "visiting.Wuhan" "from.Wuhan"   
## [19] "death" "recovered" "symptom"   
## [22] "source" "link"

A continuación, se van a explicar dichas variables:

-*Id*: número de identificación para cada paciente

-*Case\_in\_country*: se trata del número de caso de esa persona en su país.

-*Reporting date*: fecha en la que se notificó el caso.

-*Summary*: resumen general del paciente, si visita Wuhán, síntomas, cuándo aparecen los síntomas…

-*Location*: provincia/ estado de la que pertenece el individuo.

-*Country*: país del que pertenece el individuo.

-*Gender*: sexo del individuo.

-*Age*: edad del individuo.

-*Symptom\_onset*: fecha de aparición de síntomas.

-*If\_onset\_approximated*: fecha de aparición aproximada.

-*hosp\_visit\_date*: fecha de visita al hospital.

-*International\_traveller*: indica si el individuo ha viajado (1) o no (0) fuera de las fronteras de su país.

-*Domestic\_traveller*: indica si el individuo ha viajado (1) o no (0) dentro de las fronteras de su país.

-*Exposure\_start*: fecha de inicio de exposición a la enfermedad.

-*Exposure\_end*: fecha de fin de la exposición.

-*Traveler*: indica si el individuo ha viajado o no.

-*Visiting\_Wuhan*: indica si el individuo visita Wuhán

-*From\_Wuhan*: indica si el individuo proviene de Wuhán

-*Death*: indica si el individuo ha muerto.

-*Recovered*: indica si el individuo se ha recuperado de la enfermedad.

-*Symtom*: síntomas que han aparecido en el individuo.

-*Source*: fuente de donde se consiguen los datos.

-*Link*: url de los datos

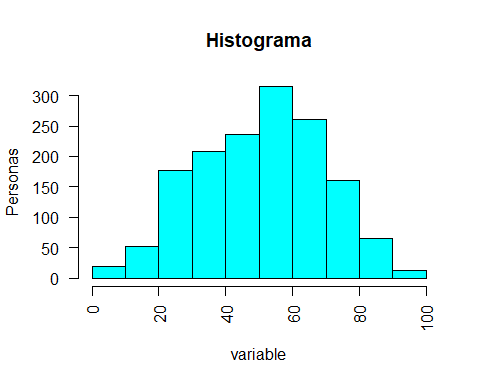
Se va a realizar una función para poder adquirir información de las diferentes variables que nos interesan.

representar\_variable <- function(variable){  
 if(is.numeric(variable)){  
 cat("La variable es de tipo numérica como podemos comprobar: ",class(variable),"\n")  
 c <- mean(variable, na.rm = TRUE)  
 cat("La media de la variable es: ",c,"\n")  
 d <- var(variable, na.rm = TRUE)  
 cat("La varianza de la variable es: ",d,"\n")  
 e<- sd(variable, na.rm = TRUE)  
 cat("La desviación estándar de la variable es: ",e,"\n")  
 f<- median(variable, na.rm = TRUE)  
 cat("La mediana de la variable es: ",f,"\n")  
 hist(variable, las=2, col="cyan", main="Histograma",ylab="Personas")  
 } else if(is.factor(variable)){  
 cat("La variable es de tipo factor como podemos comprobar: ",class(variable),"\n")  
 a1 <- levels(variable)  
 cat("A continuación, se obtendrán los diferentes valores existentes: ",a1,"\n")  
 cat("Ahora se va a ver la distribución \n")  
 pie(table(variable))  
 }  
}

La primera variable que se va a representar es la edad:

representar\_variable(datos\_pacientes$age)

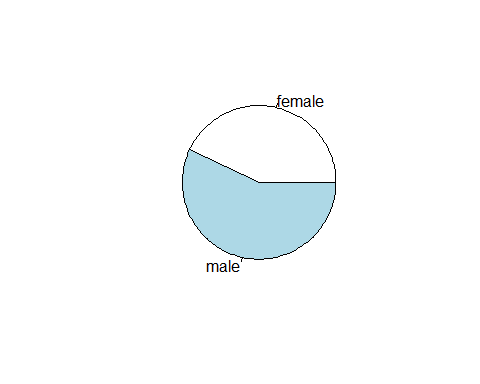
## La variable es de tipo numérica como podemos comprobar: numeric   
## La media de la variable es: 51.06967   
## La varianza de la variable es: 351.6186   
## La desviación estándar de la variable es: 18.7515   
## La mediana de la variable es: 54



La siguiente variable a representar es el género:

representar\_variable(datos\_pacientes$gender)

## La variable es de tipo factor como podemos comprobar: factor   
## A continuación, se obtendrán los diferentes valores existentes: female male   
## Ahora se va a ver la distribución



A continuación, se van a ver los posibles valores tomados en los síntomas.

head(levels(datos\_pacientes$symptom),20)

## [1] "back pain, fever, runny nose"   
## [2] "chest discomfort"   
## [3] "chills"   
## [4] "chills, cough, fever"   
## [5] "chills, fever, malaise"   
## [6] "chills, headache, fatigue"   
## [7] "cold, fever, pneumonia"   
## [8] "cough"   
## [9] "cough with sputum"   
## [10] "cough, chest pain"   
## [11] "cough, chill, muscle pain"   
## [12] "cough, chills, joint pain"   
## [13] "cough, chills, shortness of breath, diarrhea"   
## [14] "cough, diarrhea, vomiting"   
## [15] "cough, fever"   
## [16] "cough, fever, chills"   
## [17] "cough, fever, headache, nausea, vomiting, diarrhea"  
## [18] "cough, fever, joint pain"   
## [19] "cough, general malaise, joint muscle pain"   
## [20] "cough, itchy throat"

Se puede observar que los síntomas están recogidos de diferentes formas según el país por lo que habrá que normalizarlos. Así mismo, se hará lo mismo con las visitas al hospital.

levels(datos\_pacientes$hosp\_visit\_date)

## [1] "1/13/2020" "1/14/2020" "1/15/2020" "1/16/2020" "1/17/2020"   
## [6] "1/18/2020" "1/19/2020" "1/20/2020" "1/21/2020" "1/22/2020"   
## [11] "1/23/2020" "1/24/2020" "1/25/2020" "1/26/2020" "1/27/2020"   
## [16] "1/28/2020" "1/29/2020" "1/30/2020" "1/31/2020" "12/27/2019"  
## [21] "12/31/2019" "2//2020" "2/13/2020" "2/14/2020" "2/15/2020"   
## [26] "2/16/2020" "2/17/2020" "2/18/2020" "2/19/2020" "2/20/2020"   
## [31] "2/21/2020" "2/22/20" "2/22/2020" "2/23/2020" "2/24/2020"   
## [36] "2/25/2020" "2/26/2020" "2/27/2020" "2/28/2020" "2/29/2020"   
## [41] "43831" "43832" "43833" "43863" "43864"   
## [46] "43891" "43892" "43893" "43923" "43924"   
## [51] "43952" "43953" "43954" "43983" "43984"   
## [56] "43985" "44014" "44015" "44044" "44045"   
## [61] "44046" "44075" "44076" "44105" "44106"   
## [66] "44107" "44136" "44137" "44138" "44166"   
## [71] "44167"

Se puede comprobar que hay fechas que están completas en el formato mes/día/año. Sin embargo, hay otras que no están completas y en otras simplemente aparecen números. Viendo estos problemas, simplemente se creará otra columna en la que se añadirá 1 si visitó el hospital y NA si no.

datos\_pacientes$logic\_hosp <- ifelse(datos\_pacientes$hosp\_visit\_date!=0,1,0)  
head(datos\_pacientes$logic\_hosp,25)

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

Ahora se van a buscar pacientes que tengan las características que nos interesan rellenadas, es decir, presenten edad, género y los síntomas.

datos\_pacientes\_creados <- datos\_pacientes[!is.na(datos\_pacientes$age),]  
datos\_pacientes\_creados <- datos\_pacientes\_creados[!is.na(datos\_pacientes$gender),]  
datos\_pacientes\_creados <- datos\_pacientes\_creados[!is.na(datos\_pacientes\_creados$symptom),]  
cat(c("Al principio el dataset presentaba", nrow(datos\_pacientes),"filas", "\n"))

## Al principio el dataset presentaba 3257 filas

cat(c("Tras haber eliminado las filas que tenían o la edad, el género o los síntomas vacíos, el dataset presenta", nrow(datos\_pacientes\_creados),"filas", "\n"))

## Tras haber eliminado las filas que tenían o la edad, el género o los síntomas vacíos, el dataset presenta 265 filas

A continuación se van a sustituir los valores NA de la variable logic\_hosp por 0. De esta forma el dataset generado tendrá un 1 si el paciente visitó el hospital y un 0 si no lo hizo.

datos\_pacientes\_creados$logic\_hosp[is.na(datos\_pacientes\_creados$logic\_hosp)] <- 0  
datos\_pacientes\_creados$death[is.na(datos\_pacientes\_creados$death)] <- 0  
datos\_pacientes\_creados$recovered[is.na(datos\_pacientes\_creados$recovered)] <- 0

Por último se eliminarán las columnas que no van a ser usadas y se va a generar un dataset nuevo con los datos de los pacientes que no presentan ningún campo que es interesante para el análisis en vacío.

datos\_pacientes\_creados <- select(datos\_pacientes\_creados,id,gender,age,symptom,logic\_hosp)  
dir\_archivo <- "C:/Users/Daniel/Desktop/TFG-github/TFG/modelo 1/datos\_usados/anexo datos pacientes/datos/datos\_covid19.xlsx"  
write.xlsx(datos\_pacientes\_creados,dir\_archivo)

Con este dataset generado se van a normalizar los síntomas generados, de tal forma que por cada paciente aparezcan todos los síntomas y “true” si lo padece o “false” en el caso contrario. Para ello, se usará Excel y se tienen que tener en cuenta ciertas consideraciones:

-La primera de ellas es que los datos provienen de diferentes países por lo que existen síntomas nombrados de diferentes formas. Así por ejemplo existían pacientes con el síntoma “tired” que en el dataset ha sido clasificado en “fatigue”, entre otros muchos ejemplos.

-La segunda es que en esta clasificación no se va a atender al grado del síntoma. Al normalizar los datos algunos eran del estilo “high fever”, el cual no tendría cabida en el dataset, por lo que se le añade un “true” a la variable “fever”.

-La tercera es que algunos síntomas podían ser englobados en otros y aparecían en pocos pacientes. Así que para no sobrecargar el dataset, no se ha añadido ese síntoma y al paciente que lo presentaba se le ha puesto el síntoma más general y que englobaba el mencionado anteriormente.

-Y por último, también ocurría el caso contrario a la tercera consideración. Algunos pacientes tenían “flu symptoms”, por lo que se ha optado en dividirlo en los síntomas que aparecen en el dataset.

De este modo tras haberlo normalizado el dataset obtenido es el siguiente:

head(datos\_pacientes\_sintomas)

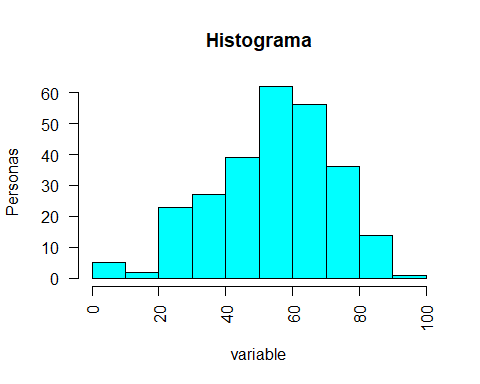
## id gender age fever cough difficult.in.breathing fatigue joint.pain  
## 1 195 female 28 true true true false false  
## 2 196 female 51 true true true false false  
## 3 197 male 37 true false false false false  
## 4 254 male 35 true false false false false  
## 5 268 male 45 true true false false false  
## 6 271 female 55 true true false false false  
## pneumonia vomiting diarrhea malaise muscle.pain sore.throat loss.of.apetit  
## 1 false false false false false false false  
## 2 false false false false false false false  
## 3 false false false false false false false  
## 4 false false false false false false false  
## 5 false false false false false false false  
## 6 false false false false false false false  
## nausea logic\_hosp  
## 1 false true  
## 2 false true  
## 3 false true  
## 4 false true  
## 5 false true  
## 6 false true

Una vez se tiene un dataset con la información que se necesita se va a profundizar en el análisis.

Se empezará de nuevo con la edad

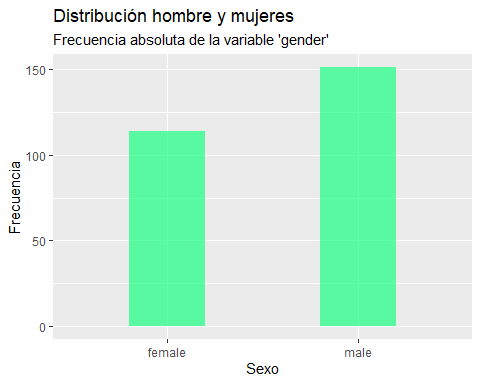
representar\_variable(datos\_pacientes\_sintomas$age)

## La variable es de tipo numérica como podemos comprobar: numeric   
## La media de la variable es: 54.39245   
## La varianza de la variable es: 319.3227   
## La desviación estándar de la variable es: 17.8696   
## La mediana de la variable es: 55



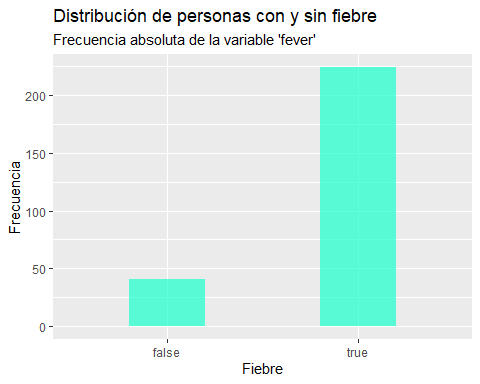
Como se puede observar se ha producido un cambio en el análisis de la edad. A continuación se va a observar la distribución de la población respecto al género. Se puede observar que hay más hombres que mujeres en el dataset.

ggplot(datos\_pacientes\_sintomas, aes(x = datos\_pacientes\_sintomas$gender)) +  
 geom\_bar(width = 0.4, fill=rgb(0.1,1,0.5,0.7)) +  
 scale\_x\_discrete("Sexo") + # configuración eje X (etiqueta del eje)  
 scale\_y\_continuous("Frecuencia") +  
 labs(title = "Distribución hombre y mujeres",  
 subtitle = "Frecuencia absoluta de la variable 'gender'")



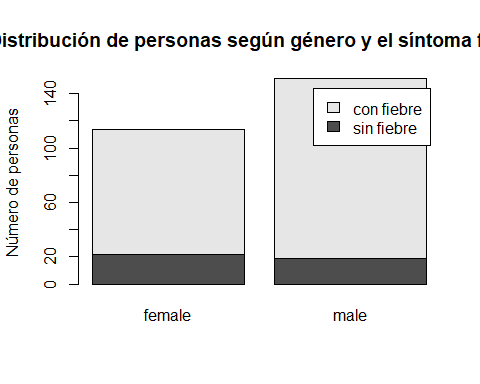
Ahora se va a realizar la distribución de la población respecto a si presenta el síntoma fiebre o no.

ggplot(datos\_pacientes\_sintomas, aes(x = datos\_pacientes\_sintomas$fever)) +  
 geom\_bar(width = 0.4, fill=rgb(0.1,1,0.8,0.7)) +  
 scale\_x\_discrete("Fiebre") + # configuración eje X (etiqueta del eje)  
 scale\_y\_continuous("Frecuencia") +  
 labs(title = "Distribución de personas con y sin fiebre",  
 subtitle = "Frecuencia absoluta de la variable 'fever'")



Además, se va a mirar esta distribución respecto al género de cada paciente.

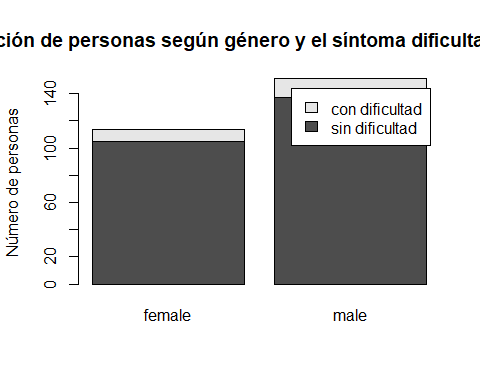
barplot(table(datos\_pacientes\_sintomas$fever,datos\_pacientes\_sintomas$gender), legend.text = c("sin fiebre", "con fiebre"), ylab="Número de personas", main = "Distribución de personas según género y el síntoma fiebre")



Se puede observar que existen un número mayor de hombres que de mujeres que presentan el síntoma fiebre.

Esto mismo se va a hacer para diferentes síntomas: dificultad al respirar, neumonía y tos.

barplot(table(datos\_pacientes\_sintomas$difficult.in.breathing,datos\_pacientes\_sintomas$gender), legend.text = c("sin dificultad", "con dificultad"), ylab="Número de personas", main = "Distribución de personas según género y el síntoma dificultad al respirar")



Se puede observar que existen más hombres que mujeres con dificultad al respirar. Esto podría ser debido a que hay más hombres que mujeres, por lo que se va a realizar la proporción.

Para ello se creará una tabla y se dividirá por el número de mujeres y hombres respectivamente.

tabla\_dificultad <- table(datos\_pacientes\_sintomas$difficult.in.breathing,datos\_pacientes\_sintomas$gender)  
numero\_mujeres <- as.numeric(sum((datos\_pacientes\_sintomas$gender=="female")\*1))  
numero\_hombres <- as.numeric(sum((datos\_pacientes\_sintomas$gender=="male")\*1))

tabla\_dificultad[1,1] <- tabla\_dificultad[1,1]/numero\_mujeres  
tabla\_dificultad[2,1] <- tabla\_dificultad[2,1]/numero\_mujeres  
tabla\_dificultad[1,2] <- tabla\_dificultad[1,2]/numero\_hombres  
tabla\_dificultad[2,2] <- tabla\_dificultad[2,2]/numero\_hombres

Y así se conseguiría la tabla correspondiente

tabla\_dificultad

##   
## female male  
## false 0.92105263 0.90728477  
## true 0.07894737 0.09271523

De esta forma se observa como es más probable que los hombres tengan dicho síntoma. A continuación, se va a repetir el proceso con la neumonía.

tabla\_neumonia <- table(datos\_pacientes\_sintomas$pneumonia,datos\_pacientes\_sintomas$gender)

tabla\_neumonia[1,1] <- tabla\_neumonia[1,1]/numero\_mujeres  
tabla\_neumonia[2,1] <- tabla\_neumonia[2,1]/numero\_mujeres  
tabla\_neumonia[1,2] <- tabla\_neumonia[1,2]/numero\_hombres  
tabla\_neumonia[2,2] <- tabla\_neumonia[2,2]/numero\_hombres

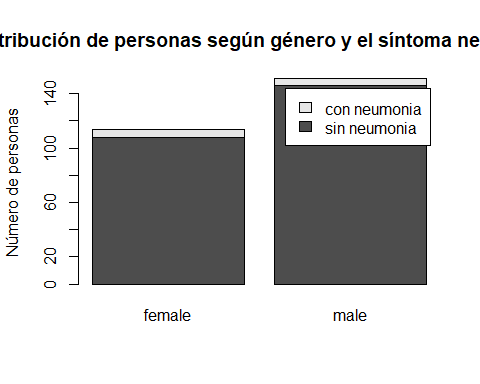
Ahora se va a ver la proporción de la neumonía en la población según el género

tabla\_neumonia

##   
## female male  
## false 0.94736842 0.96688742  
## true 0.05263158 0.03311258

Una vez visto los valores, se va a realizar la representacción.

barplot(table(datos\_pacientes\_sintomas$pneumonia,datos\_pacientes\_sintomas$gender), legend.text = c("sin neumonia", "con neumonia"), ylab="Número de personas", main = "Distribución de personas según género y el síntoma neumonía")



Según los datos de esta población se puede ver como es más probable la neumonía en mujeres que en hombres.

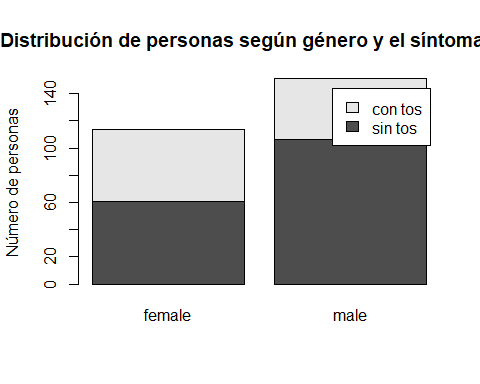
Por último, se va a realizar el mismo proceso con el síntoma tos.

tabla\_tos <- table(datos\_pacientes\_sintomas$pneumonia,datos\_pacientes\_sintomas$gender)  
tabla\_tos[1,1] <- tabla\_tos[1,1]/numero\_mujeres  
tabla\_tos[2,1] <- tabla\_tos[2,1]/numero\_mujeres  
tabla\_tos[1,2] <- tabla\_tos[1,2]/numero\_hombres  
tabla\_tos[2,2] <- tabla\_tos[2,2]/numero\_hombres

tabla\_tos

##   
## female male  
## false 0.94736842 0.96688742  
## true 0.05263158 0.03311258

barplot(table(datos\_pacientes\_sintomas$cough,datos\_pacientes\_sintomas$gender), legend.text = c("sin tos", "con tos"), ylab="Número de personas", main = "Distribución de personas según género y el síntoma tos")



Se vuelve a dar la coincidencia que el síntoma tos es más probable en las mujeres que en los hombres.

Por último, se va a estudiar el porcentaje de cada síntoma en la población redondeando a partir del segundo decimal que es lo que realmente va a servir a la hora de la realización del modelado, ya que se introducirá un porcentaje al agente de tener dicho síntoma.

tabla\_porcentajes <- data.frame (cbind(nombres,valores))  
tabla\_porcentajes

## nombres valores  
## 1 Porcentaje de fiebre 84.53  
## 2 Porcentaje de tos 36.98  
## 3 Porcentaje dificultad al respirar 8.68  
## 4 Porcentaje de fatiga 7.55  
## 5 Porcentaje de dolor articular 4.91  
## 6 Porcentaje de neumonía 4.15  
## 7 Porcentaje de vómitos 3.02  
## 8 Porcentaje de diarrea 2.26  
## 9 Porcentaje de malestar 19.25  
## 10 Porcentaje de dolor muscular 5.66  
## 11 Porcentajde de dolor de garganta 10.57  
## 12 Porcentaje de pérdida de apetito 1.13  
## 13 Porcentaje de nausea 1.13  
## 14 Porcentaje visita hospital 79.25

Además, se va a crear otro dataset en el que se añadan también otras características del primer dataset inicial.

vector\_id <- datos\_pacientes\_sintomas$id  
data\_auxiliar <- datos\_pacientes[vector\_id,]  
  
datos\_pacientes\_sintomas$traveler<- ifelse(data\_auxiliar$traveler!=0,"true","false")  
datos\_pacientes\_sintomas$traveler[is.na(datos\_pacientes\_sintomas$traveler)] <- "false"  
  
datos\_pacientes\_sintomas$death<- ifelse(data\_auxiliar$death!=0,"true","false")  
  
datos\_pacientes\_sintomas$recovered<- ifelse(data\_auxiliar$recovered!=0,"true","false")

A continuación se va a exportar el dataset en un archivo .xlsx.

dir\_archivo <- "C:/Users/Daniel/Desktop/TFG-github/TFG/modelo 1/datos\_usados/anexo datos pacientes/datos/datos\_pacientes\_sintomas\_completos.xlsx"  
write.xlsx(datos\_pacientes\_sintomas,dir\_archivo)

También se va a conseguir el porcentjaje de recuperados, muertos y viajantes.

tabla\_porcentajes2 <- data.frame (cbind(nombres2,valores2))  
tabla\_porcentajes2

## nombres2 valores2  
## 1 Porcentaje muerte 3.02  
## 2 Porcentaje recuperación 7.17  
## 3 Porcentaje viajeros 14.72

Viendo los resultados obtenidos, se puede concluir que existen muchos pacientes del dataset que seguían infectados el día de descarga del conjunto de datos.