Pauta Evaluación 2

Natalie Julian

¿Cómo saber si una cuenta es *fake*?

Instagram es una red social lanzada al mundo el 10 de Octubre de 2010. La dinámica es simple: los usuarios comparten fotos y videos de forma instantánea, pudiendo interactuar con sus seguidores y amigos. Instagram tiene más de 500 millones de usuarios activos por mes. Sin embargo, una problemática no menor es que existen muchos usuarios que crean cuentas falsas, es decir, son usuarios *Fake*. Generalmente, este tipo de cuentas son creadas con fines maliciosos, como realizar spam, espiar o subir contenido inapropiado a Instagram, desagradable para los demás usuarios de la red social.

Usualmente se diseñan algoritmos de detección de perfiles falsos en base a sus características y comportamiento. El archivo `fake` contiene datos de cuentas de Instagram que el algoritmo ya ha clasificado como *fake* o real:

- **profilepic** Indica con un 1 si la cuenta tiene foto de perfil o con un 0 si no
- **description** Indica la cantidad de caracteres usados en la descripción de la cuenta
- **private** Indica si la cuenta es privada o pública
- **posts** Indica la cantidad de publicaciones de la cuenta
- **followers** Indica la cantidad de seguidores de la cuenta
- **follows** Indica la cantidad de cuentas seguidas
- **Fake** Indica si la cuenta correspondía a una cuenta falsa o no

a) Cargue los datos del archivo `fake` en R. ¿Qué función utilizó para cargar los datos? ¿En qué paquete viene incorporada esta función? ¿Por qué es necesario cargar paquetes? Comente.

Una solución

cargar y guardar correctamente los datos

decir que la función que carga los datos es import() decir que la función viene del paquete rio

```
library(rio) #Carga la librería/paquete llamado rio

datos<-import("fake.csv") #Se utiliza la función import del paquete rio
```

comentar por qué es necesario cargar paquetes

Es necesario cargar paquetes para poder acceder a funciones que no se encuentran por defecto en RStudio. Muchas veces, funciones más específicas para gráficos u otros.

b) ¿En qué formato se lee la variable *followers*? ¿Por qué se ha leído en este formato? Realice la modificación correspondiente de manera que el formato sea el adecuado. Verifique que se realizó correctamente la modificación.

Una solución

comentar que followers posee un formato de tipo character

```
str(datos$followers) #Muestra el formato de la variable
```

```
##  chr [1:120] "488" "35" "328" "14890" "225" "362" "213" "552" "122" "834" ...
```

Ya sabemos que cuando una variable numérica se está leyendo en formato character es porque debe existir texto en algunas de las observaciones. ¿Cuáles?

comentar que la variable followers toma el valor "Ninguno" por esto se lee en formato character

```
#Forma 1: con unique()

unique(datos$followers) #Muestra los valores que toma esta variable
```

```
##   [1] "488"     "35"      "328"     "14890"   "225"     "362"     "213"
##   [8] "552"     "122"     "834"     "229"     "1913"    "200"     "130"
##  [15] "192"     "498"     "96"      "202"     "175"     "223"     "189"
##  [22] "486"     "464"     "150"     "2983"    "116"     "155537"  "248"
##  [29] "348"     "4021842" "366"     "1064"    "81267"   "400"     "361"
##  [36] "228"     "855"     "777"     "264"     "1572"    "510"     "1027419"
##  [43] "710"     "2267"    "2055"    "814"     "668"     "87"      "461"
##  [50] "602517"  "62"      "341"     "717"     "386"     "673"     "654"
##  [57] "751"     "209"     "573"     "284"     "Ninguno" "45"      "19"
##  [64] "69"      "22"      "31"      "9"       "23"      "17"      "46"
##  [71] "16"      "21"      "52"      "24"      "13"      "227"     "10"
##  [78] "57"      "1789"    "309"     "1742"    "1906"    "39"      "119"
##  [85] "2997"    "772"     "129"     "94"      "37"      "75"      "42"
##  [92] "145"     "128"     "88"      "1987"    "100"     "214"     "415"
##  [99] "926"     "238"     "193"     "49"      "74"      "114"     "833"
## [106] "219"
```

```
#Forma 2: con table()
table(datos$followers) #Muestra tabla de frecuencias de esta variable
```

```
##
##      10     100 1027419    1064     114     116     119     122     128     129
##       1       1       1       1       1       1       1       1       1       1
##      13     130     145   14890     150  155537    1572      16      17    1742
##       2       1       1       1       2       1       1       1       2       1
##     175    1789     189      19    1906    1913     192     193    1987     200
##       1       1       1       1       1       1       2       1       1       1
##     202    2055     209      21     213     214     219      22     223     225
##       1       1       1       2       1       1       1       1       1       1
##    2267     227     228     229      23     238      24     248     264     284
##       1       2       1       1       1       1       1       1       1       1
##    2983    2997     309      31     328     341     348      35     361     362
##       1       1       1       1       1       2       1       1       1       1
##     366      37     386      39     400 4021842     415      42      45      46
##       1       1       1       2       1       1       1       1       2       2
##     461     464     486     488      49     498     510      52     552      57
##       1       1       1       1       1       1       1       1       1       1
##     573  602517      62     654     668     673      69     710     717      74
##       1       1       2       1       1       1       2       1       1       1
```

```
##      75     751     772     777   81267     814     833     834     855      87
##       1       1       1       1       1       1       1       1       1       1
##      88       9     926      94      96 Ninguno
##       2       1       1       1       1       2
```

realizar la modificación correspondiente

En este caso, era necesario interpretar que followers contiene la cantidad de seguidores de la cuenta, por lo tanto "Ninguno" corresponde a tener 0 seguidores.

```
datos$followers[which(datos$followers=="Ninguno")]<-0 #Aquellos registros Ninguno los llena con 0

datos$followers<-as.numeric(datos$followers) #Redefine el formato a numerico
```

verificar que la variable posee el formato que corresponde

```
str(datos$followers) #Ahora posee el formato correspondiente


##  num [1:120] 488 35 328 14890 225 ...
```

c) La variable *profilepic* es una variable categórica que indica si la cuenta tiene o no tiene foto de perfil. Sin embargo, está codificada de forma numérica (unos y ceros). ¿Por qué pudiera ser relevante detectar aquellas categorías codificadas numéricamente previo al análisis? ¿Qué errores podrían cometerse si no realizó este alcance? Comente. Recodifique esta variable en las categorías que corresponde. ¿Cuántos usuarios tenían foto de perfil? ¿Cuántos no?

Una solución

comentar por qué es relevante detectar cuáles variables no están codificadas como corresponde

En este caso, en esta variable se indicaba con 0 si no tenía foto de perfil y con 1 si tenía foto de perfil, si no notáramos este detalle podríamos estar calculando erróneamente estadísticas de esta variable que se encuentra codificada con números pero que en realidad corresponde a categorías.

recodificar de manera adecuada la variable

```
#Valores 0 indican que no tiene foto de perfil:
datos$profilepic[which(datos$profilepic==0)]<-"Sin foto de perfil"

#Valores 1 indican que tiene foto de perfil:
datos$profilepic[which(datos$profilepic==1)]<-"Con foto de perfil"
```

En este ejercicio es irrelevante si redefinieron la variable o crearon otra nueva, lo importante es que estuviera incluida en los datos fake.

comentar cuántos usuarios o cuentas tenían foto de perfil y cuántos no

```
table(datos$profilepic)


##
## Con foto de perfil Sin foto de perfil
##                 91                 29
```

En los datos hay 91 cuentas de Instagram con foto de perfil y 29 sin foto de perfil.

d) Se cree que en general, las cuentas *fake* no suelen tener foto de perfil. Muestre en una tabla cómo se distribuye el tener o no tener foto de perfil en las cuentas *fake* y *reales*. Comente. ¿Le parece que tener foto de perfil asegura que una cuenta sea real?

Una solución

obtener cuántas cuentas fake hay con foto de perfil y sin foto de perfil y lo mismo pero con cuentas reales

```
#Forma 1: Tabla de dobe entrada:
table(datos$profilepic, datos$fake)
```

```
##
##                      Fake Not Fake
##   Con foto de perfil   31       60
##   Sin foto de perfil   29        0
```

```
#Forma 2:

table(datos$profilepic[which(datos$fake=="Fake")]) #Filtro cuentas fake y veo tabla de profilepic
```

```
##
## Con foto de perfil Sin foto de perfil
##                 31                 29
```

```
table(datos$profilepic[which(datos$fake=="Not Fake")]) #Filtro cuentas reales y veo tabla de profilepic
```

```
##
## Con foto de perfil
##                 60
```

comentar sobre las tablas o valores obtenidos

Hay 29 cuentas fake que no tenían foto de perfil y 31 cuentas fake que tenian foto de perfil. Mientras que las cuentas reales, todas tenian foto de perfil.

concluir finalmente si tener foto de perfil asegura que la cuenta sea real

Lo que se puede concluir es que existen cuentas que tenían foto de perfil y que a pesar de esto, eran cuentas fake, por lo tanto no aseguraría realmente el tener foto de perfil que la cuenta sea real.

e) El cuociente de seguidores entre seguidos de una cuenta es:

$$\text{cuociente} = \frac{\text{followers}}{\text{follows}}$$

Este cuociente se utiliza como una variable relevante para clasificar si la cuenta es fake, menores valores del cuociente indicarían que una cuenta es *fake*. Calcule la variable *cuociente* y agréguela a los datos. ¿Qué rango de valores recorre dicho cuociente para las cuentas fake? ¿y para las cuentas reales? Compare. ¿Por qué tendría sentido realcionar menores valores del cuociente con cuentas fake? ¿Cómo se puede interpretar el cuociente en términos del contexto? Conjeture.

Una solución

calcular el cuociente

```
datos$followers/datos$follows
```

```
##   [1] 8.079470e-01 5.833333e+00 4.910180e-01 2.020627e+00 6.320225e-01
##   [6] 8.537736e-01 8.385827e-01 1.059501e+00 8.531469e-01 2.329609e+00
##  [11] 4.654472e-01 4.387615e+00 4.576659e-01 2.090032e-01 1.361702e+00
##  [16] 1.477745e+00 1.923848e-01 3.338843e-01 8.793970e-01 3.213256e-01
##  [21] 6.847826e-01 5.638051e-01 1.264305e+00 9.554140e-01 5.473394e+00
##  [26] 8.405797e-01 1.114961e+02 5.061224e-01 1.002882e+00 7.293874e+02
##  [31] 6.630435e-01 1.856894e+00 8.438941e+01 8.908686e-01 6.423488e-01
##  [36] 6.589595e-01 5.662252e+00 5.250000e+00 1.748344e+00 4.486301e-01
##  [41] 2.756757e+00 3.506549e+03 1.293260e+00 4.864807e+00 2.069486e+00
##  [46] 7.326733e-01 1.104132e+00 2.175000e+00 4.369668e-01 1.250035e+03
##  [51] 1.319149e+00 1.244526e+00 3.215247e+00 1.063361e+00 1.184859e+00
##  [56] 1.222430e+00 1.301560e+00 7.572464e-01 1.208861e+00 5.623762e-01
##  [61] 0.000000e+00 7.031250e-01 6.333333e-01 9.942363e-02 2.682927e-01
##  [66] 2.500000e-01 3.600000e-01 9.942363e-02 6.969697e-01 5.000000e-01
##  [71] 1.210526e+00 8.000000e+00 0.000000e+00 2.100000e+01 3.466667e+00
##  [76] 1.200000e+01 5.909091e-01 6.430595e-01 4.166667e-01 2.555556e+00
##  [81] 2.590909e+00 1.491036e-01 2.907525e-01 7.031250e-01 6.774194e-01
##  [86] 1.236000e+00 2.822424e-01 8.952560e-01 1.203704e-01 5.151515e-01
##  [91] 4.920635e-01 3.400000e-01 3.922775e+00 2.383452e-01 1.402174e-01
##  [96] 8.952381e-01 6.379310e-01 1.363636e+00 2.400000e-01 7.178218e-01
## [101] 2.012579e-01 1.222222e+00 2.666041e-01 6.172840e-01 2.581423e-01
## [106] 2.925258e-01 2.038217e-01 2.871972e-01 2.184477e-01 1.723389e-01
## [111] 2.884903e-01 2.085106e-01 1.857143e+00 2.740741e-01 1.157895e+00
## [116] 1.405672e-01 9.146341e-01 2.332027e-01 1.292035e-01 5.735294e-01
```

agregar a los datos/tabla de datos

```
datos$cuociente<-datos$followers/datos$follows
```

calcular rango de valores del cuociente para cuentas fake y reales

```
#Forma 1:

summary(datos$cuociente[which(datos$fake=="Fake")])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.2371  0.4544  1.3243  0.8952 21.0000
```

```
summary(datos$cuociente[which(datos$fake=="Not Fake")])
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    0.192    0.662    1.084   96.088    2.096 3506.549
```

```
#Forma 2:

range(datos$cuociente[which(datos$fake=="Fake")])
```

```
## [1]  0 21
```

```
range(datos$cuociente[which(datos$fake=="Not Fake")])
```

```
## [1]    0.1923848 3506.5494881
```

```r
#Forma 3:

min(datos$cuociente[which(datos$fake=="Fake")]);max(datos$cuociente[which(datos$fake=="Fake")])


## [1] 0
## [1] 21


min(datos$cuociente[which(datos$fake=="Not Fake")]);max(datos$cuociente[which(datos$fake=="Not Fake")])


## [1] 0.1923848
## [1] 3506.549
```

comentar sobre los rangos

Se puede observar que efectivamente el rango de valores para el cuociente es mayor para las cuentas reales que para las cuentas fake.

conjeturar

Naturalmente, dado que el cuociente corresponde a la cantidad de seguidores respecto a seguidos, como las cuentas fake usualmente se utilizan para realizar spam, espiar u otros fines maliciosos, es natural que no tengan tantos seguidores pero que sí sigan más cuentas. En cuentas reales se esperaría que tengan más seguidores, correspondiente a amistades, familiares u otros.

BONUS) Instale y cargue el paquete `dplyr`. Luego corra las siguientes líneas de código:

```r
datos %>%
  group_by(fake) %>%
   summarise(Min=min(cuociente),
             Max=max(cuociente))
```

¿Qué resultado observa? Explique con sus palabras qué cree que realiza cada línea de código para llegar al resultado obtenido.

Una solución

```r
#install.packages("dplyr")

library(dplyr)

datos %>%
  group_by(fake) %>%
   summarise(Min=min(cuociente),
             Max=max(cuociente))


## # A tibble: 2 x 3
##   fake       Min   Max
##   <chr>    <dbl> <dbl>
## 1 Fake     0       21
## 2 Not Fake 0.192 3507.
```

instalar y cargar el paquete dplyr

notar que entrega los rangos de valores calculados en el item e)

dar ideas de lo que realiza el comando, que agrupa por tipo de cuenta y luego calcula minimo y maximo del cuociente