Probabilidad

Selección de actividades resueltas

© José Fco. Martínez Boscá, Arnau Mir Torres, Lluís M. Pla Aragonés, Àngel J. Gil Estallo (Autores) & Ángel A. Juan (Editor)

© FUOC 2009

UOC Universitat Oberta de Catalunya

Introducción

Una de las aplicaciones más claras del cálculo de probabilidades a Ciencias de la Computación es la lucha contra el correo no deseado. O sea, ¿cómo detectamos si un correo recibido es no deseado o "spam"? O dicho de otra forma, ¿cómo diseñamos un filtro que nos separe el correo "spam" del que no lo es?

El profesor Paul Graham en la web http://www.paulgraham.com/spam.html explica que antes de probar métodos bayesianos o probabilísticos, usó métodos directos para reconocer el correo "spam" pero se encontró con la sorpresa que reconocer los "spams" cada vez se convertía en una tarea más ardua con la aparición de numerosos falsos positivos.[1]

La idea fundamental del filtro es el autoaprendizaje. O sea, cuánto más correo recibimos, el filtro aprende más y su fiabilidad va en alza.

Veamos un esbozo de cómo funciona. En primer lugar, suponemos que tenemos un "corpus" de correo "spam" y otro "corpus" de correo "ham" o correo válido. Hacemos un escaneo de todos los "tokens" o palabras de los dos "corpus" anteriores y contamos el número de veces que aparece cada "token" en cada "corpus". De esta forma, creamos dos tablas tipo "hash", una para cada "corpus", donde asignamos a cada "token" el número de veces que aparece en el "corpus" "spam" y el número de veces que aparece en el corpus "ham". Seguidamente, creamos una tercera tabla "hash" donde asignamos a cada "token" la probabilidad de que un correo que contenga el "token" sea "spam". Dicha probabilidad la calculamos de la forma siguiente. Sea b el valor que tiene "token" en la primera tabla "hash" y g el valor que tiene "token" en la segunda tabla "hash". O sea, b sería el número de veces que aparece "token" en el "corpus" "spam" y g sería el número de veces que aparece "token" en el corpus "ham". Sea nbad el número de correos "spam" en el corpus "spam" y ngood el número de correos "ham" en el corpus "ham". La probabilidad anterior es:

$$p(\text{SPAM} \mid \text{ email contiene token}) = \frac{p(\text{SPAM} \cap \text{email contiene token})}{p(\text{email contiene token})}$$

$$= \frac{\text{b/nbad}}{\text{b/nbad+g/ngood}}.$$

Si alguno de los cocientes anteriores (b/nbad o g/ngood) es mayor que 1, cogemos 1 como valor del cociente.

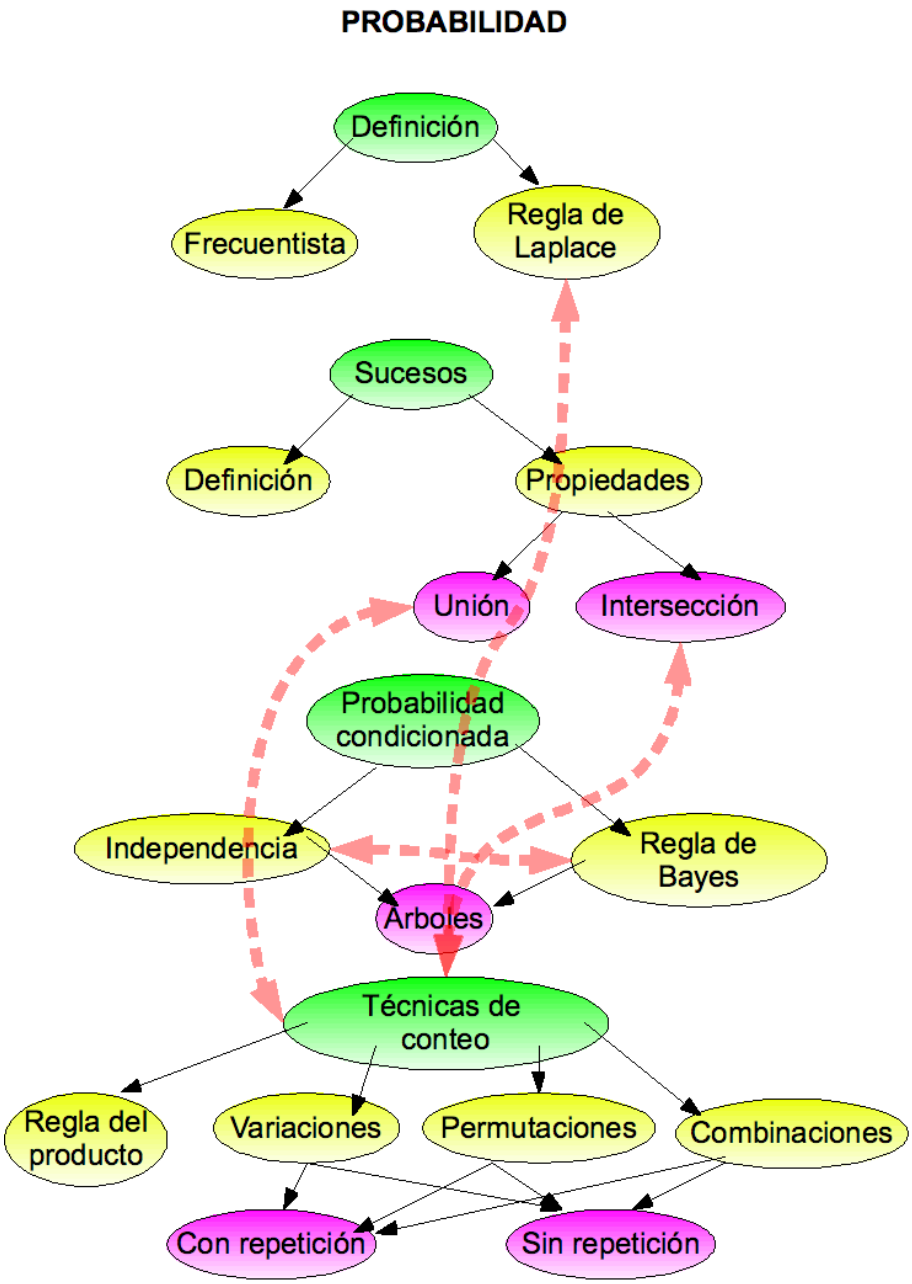
Una vez creadas las tres tablas "hash", veamos cómo funciona nuestro filtro. Imaginemos que entra un correo nuevo en nuestro servidor de correo. Nuestro filtro escanea el mensaje y lo divide en "tokens". Luego considera los 15 tokens más interesantes; por "más interesantes" queremos decir los 15 "tokens" cuyo valor en la tercera tabla "hash" o probabilidad de que el mensajes sea "spam" estén más separados del valor 0.5, que sería el valor neutro de que la probabilidad de que un mensaje que contenga el "token" sea "spam". Una vez separados estos 15 "tokens" más interesantes, hallamos la probabilidad "combinada" de que el mensaje sea "spam" (ver http://www.mathpages.com/home/kmath267.htm para entender el concepto de probabilidad "combinada"). Para hallar la probabilidad anterior, sea $p_1, p_2, \ldots, p_{15}$ las probabilidades de que el mensaje entrante sea "spam" con respecto al "token" i, para i=1,…,15. La probabilidad de que el mensaje sea "spam" sería:

$$p(\text{SPAM}) = \frac{p_1 \cdot p_2 \cdots p_{15}}{p_1 \cdot p_2 \cdots p_{15} + (1-p_1)\cdot(1-p_2)\cdots(1-p_{15})}.$$

El usuario asigna un "score" al filtro de forma que todos los mensajes que superan dicho "score", el filtro los considera "spam". De esta forma, el filtro separa los mensajes "spam" de los "ham".

1 Un falso positivo es un mensaje de correo que el filtro detecta como spam cuando en realidad no lo es.

Mapa conceptual



Actividades

Actividad 1: Relación entre el tiempo de computación de un programa informático y el sistema operativo usado.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

El tiempo de computación (en segundos) de un determinado programa informático ejecutado de forma independiente 100 veces en una misma máquina vale: 4.67, 4.94, 5.09, 4.74, 4.63, 4.62, 4.53, 4.89, 5.12, 4.78, 4.51, 5.17, 4.53, 4.64, 4.57, 4.92, 5.15, 4.51, 4.57, 4.86, 4.64, 4.66, 4.98, 4.71, 5.07, 5.14, 4.54, 4.90, 4.88, 4.91, 5.16, 4.99, 5.19, 4.62, 4.56, 4.81, 5.10, 5.12, 4.69, 4.77, 5.04, 4.61, 4.72, 4.85, 5.20, 4.55, 4.52, 4.83, 5.09, 4.76, 4.64, 4.86, 4.68, 5.03, 4.57, 5.17, 4.56, 4.99, 4.95, 4.92, 4.70, 4.89, 5.01, 4.60, 4.65, 4.95, 4.79, 4.55, 5.01, 4.92, 4.60, 4.63, 4.77, 4.93, 4.85, 4.70, 4.78, 4.68, 5.02, 4.87, 4.72, 4.66, 4.66, 4.83, 4.87, 4.66, 5.08, 4.83, 4.75, 5.11, 4.81, 4.66, 4.68, 5.03, 5.02, 5.04, 4.82, 4.62, 4.92, 4.90.
En los valores en que les centésimas son pares, el programa se ejecutó con el sistema operativo VENTANAS y en los demás casos, con el sistema operativo LINCAT.
Agrupamos la variable “tiempo de computación” en intervalos de amplitud 0.231 empezando con el valor 4.51.
Se pide:

a) Construid la tabla de contingencia que permita estudiar la relación entre las dos variables “tiempo de computación agrupado” y “sistema operativo usado”. Para la construcción de las tablas, podemos usar la función “table”.
b) Calculad la probabilidad de que un valor del tiempo de computación escogido al azar computado con el sistema operativo VENTANAS esté en el tercer intervalo.
c) Hallad la probabilidad de que un valor del tiempo de computación escogido al azar esté en el último intervalo y que esté computado con el sistema operativo LINCAT.
d) Los sucesos “computado con LINCAT” y “estar en el segundo intervalo”, ¿son independientes? ¿Por qué?

Solución

a) En primer lugar introducimos los datos en R:

```
OBS<-c(4.67, 4.94, 5.09, 4.74, 4.63, 4.62, 4.53, 4.89, 5.12, 4.78, 4.51, 5.17, 4.53,
       4.64, 4.57, 4.92, 5.15, 4.51, 4.57, 4.86, 4.64, 4.66, 4.98, 4.71, 5.07, 5.14,
       4.54, 4.90, 4.88, 4.91, 5.16, 4.99, 5.19, 4.62, 4.56, 4.81, 5.10, 5.12, 4.69,
       4.77, 5.04, 4.61, 4.72, 4.85, 5.20, 4.55, 4.52, 4.83, 5.09, 4.76, 4.64, 4.86,
       4.68, 5.03, 4.57, 5.17, 4.56, 4.99, 4.95, 4.92, 4.70, 4.89, 5.01, 4.60, 4.65,
       4.95, 4.79, 4.55, 5.01, 4.92, 4.60, 4.63, 4.77, 4.93, 4.85, 4.70, 4.78, 4.68,
       5.02, 4.87, 4.72, 4.66, 4.66, 4.83, 4.87, 4.66, 5.08, 4.83, 4.75, 5.11, 4.81,
       4.66, 4.68, 5.03, 5.02, 5.04, 4.82, 4.62, 4.92, 4.90)
OBS

##   [1] 4.67 4.94 5.09 4.74 4.63 4.62 4.53 4.89 5.12 4.78 4.51 5.17 4.53 4.64
##  [15] 4.57 4.92 5.15 4.51 4.57 4.86 4.64 4.66 4.98 4.71 5.07 5.14 4.54 4.90
##  [29] 4.88 4.91 5.16 4.99 5.19 4.62 4.56 4.81 5.10 5.12 4.69 4.77 5.04 4.61
##  [43] 4.72 4.85 5.20 4.55 4.52 4.83 5.09 4.76 4.64 4.86 4.68 5.03 4.57 5.17
##  [57] 4.56 4.99 4.95 4.92 4.70 4.89 5.01 4.60 4.65 4.95 4.79 4.55 5.01 4.92
##  [71] 4.60 4.63 4.77 4.93 4.85 4.70 4.78 4.68 5.02 4.87 4.72 4.66 4.66 4.83
##  [85] 4.87 4.66 5.08 4.83 4.75 5.11 4.81 4.66 4.68 5.03 5.02 5.04 4.82 4.62
##  [99] 4.92 4.90
```

A continuación creamos la variable “SIS_OP” mediante la instrucción round(100*OBS,0) %% 2 donde "%%" es el operador de módulo y round(...,0) redondea a enteros. Por lo tanto round(100*OBS;0) pasará a valores enteros la columna que nos indica el tiempo de computación multiplicada por 100. El operador "%%" nos da el resto de dividir un entero entre otro. Por tanto, round(100*OBS,0) %% 2 nos dará 1 si las centésimas de la variable OBS son impares y 0 si son pares. Por tanto, si OBS es 1, el programa se ejecutó con LINCAT y si es 0, con VENTANAS. Obtenemos

```
SIS_OP<-round(100*OBS,0) %% 2
SIS_OP##   [1] 1 0 1 0 1 0 1 1 0 0 1 1 1 0 1 0 1 1 1 0 0 0 0 1 1 0 0 0 0 1 0 1 1 0 0
```

```
##  [36] 1 0 0 1 1 0 1 0 1 0 1 0 1 1 0 0 0 0 1 1 1 0 1 1 0 0 1 1 0 1 1 1 1 1 0
##  [71] 0 1 1 1 1 1 0 0 0 0 1 0 0 0 1 1 0 0 1 1 1 1 1 0 0 1 0 0 0 0 0 0 0
```

A continuación creamos la variable TEMP_AGRUP (tiempo de computación agrupado). Los intervalos son los siguientes: [4.51,4.51+0.231), [4.51+0.231,4.51+2*0.231) [4.51+20.231,4.51+3*0.231). Como son abiertos por la derecha usaremos la función "cut" con la opción "right=FALSE" (si es necesario se puede pedir ayuda con "> help(cut)").

```
TEMP_AGRUP<-cut(OBS,breaks=c(4.51, 4.51+0.231,4.51+2*0.231,4.51+3*0.231),
                labels=c("T_bajo","T_medio","T_alto"),right=FALSE)
TEMP_AGRUP

##   [1] T_bajo  T_medio T_alto  T_bajo  T_bajo  T_bajo  T_bajo  T_medio
##   [9] T_alto  T_medio T_bajo  T_alto  T_bajo  T_bajo  T_bajo  T_medio
##  [17] T_alto  T_bajo  T_bajo  T_medio T_bajo  T_bajo  T_alto  T_bajo
##  [25] T_alto  T_alto  T_bajo  T_medio T_medio T_medio T_alto  T_alto
##  [33] T_alto  T_bajo  T_bajo  T_medio T_alto  T_alto  T_bajo  T_medio
##  [41] T_alto  T_bajo  T_bajo  T_medio T_alto  T_bajo  T_bajo  T_medio
##  [49] T_alto  T_medio T_bajo  T_medio T_bajo  T_alto  T_bajo  T_alto
##  [57] T_bajo  T_alto  T_medio T_medio T_bajo  T_medio T_alto  T_bajo
##  [65] T_bajo  T_medio T_medio T_medio T_bajo  T_alto  T_medio T_bajo  T_bajo
##  [73] T_medio T_medio T_medio T_bajo  T_medio T_bajo  T_alto  T_medio
##  [81] T_bajo  T_bajo  T_bajo  T_medio T_medio T_bajo  T_alto  T_medio
##  [89] T_medio T_alto  T_medio T_bajo  T_bajo  T_alto  T_alto  T_alto
##  [97] T_medio T_bajo  T_medio T_medio
## Levels: T_bajo T_medio T_alto
```

Para poder ver los datos en forma de columna podemos incorporar las variables a una "data.frame" y escribimos las primeras 10 filas:

```
DADES1<-data.frame(OBS,SIS_OP,TEMP_AGRUP)
head(DADES1,10)

##     OBS SIS_OP TEMP_AGRUP
## 1  4.67      1     T_bajo
## 2  4.94      0    T_medio
## 3  5.09      1     T_alto
## 4  4.74      0     T_bajo
## 5  4.63      1     T_bajo
## 6  4.62      0     T_bajo
## 7  4.53      1     T_bajo
## 8  4.89      1    T_medio
## 9  5.12      0     T_alto
## 10 4.78      0    T_medio
```

A continuación se muestra la tabla de contingencia usando las indicaciones del enunciado, y usando "addmargins" para obtener los totales por filas y columnas

```
TA<-table(SIS_OP,TEMP_AGRUP)
TA

##       TEMP_AGRUP
## SIS_OP T_bajo T_medio T_alto
##      0     25      14     12
##      1     16      19     14

TAM<-addmargins(TA)
TAM

##       TEMP_AGRUP
## SIS_OP T_bajo T_medio T_alto Sum
##    0       25      14     12  51
##    1       16      19     14  49
##    Sum     41      33     26 100
```

b) La probabilidad pedida vale: $$p(\text{T\_alto/VENTANAS}) = \frac{12}{51} \simeq 0{,}24.$$

c) La probabilidad pedida vale: $p(\text{T\_alto} \cap \text{LINCAT}) = \frac{14}{100} = 0{,}14.$

d) Hallemos las probabilidades siguientes: $p(\text{T\_medio})$, $p(\text{LINCAT})$ y $p(\text{T\_medio} \cap \text{LINCAT})$: $p(\text{T\_medio}) = \frac{33}{100} = 0,33$, $p(\text{LINCAT}) = \frac{49}{100} = 0,49$, $p(\text{T\_medio} \cap \text{LINCAT}) = \frac{19}{100} = 0{,}19$, pero $p(\text{T\_medio} \cap \text{LINCAT}) \neq p(\text{T\_medio}) \cdot p(\text{LINCAT})$ ya que $0{,}19 \neq 0,49 \cdot 0,33 = 0{,}1617$. Por tanto, no son independientes.

Actividad 2: Inmersión de las tecnologías de la información y comunicación en los municipios.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

En la taba siguiente se recogen los resultados de unas encuestas a diferentes municipios sobre el uso de las TICs el año 2007. De cada municipio tenemos 4 valores: LLAR_ORD (proporción de hogares que tienen ordenador), LLAR_BA (proporción de hogares que tienen banda ancha), USU_ORD (proporción de habitantes que han utilizado el último mes el ordenador) y USU_COR (proporción de habitantes que han utilizado el último mes el correo electrónico). En este problema estudiaremos y compararemos las variables LLAR_ORD y USU_ORD.

LLAR_ORD	LLAR_BA	USU_ORD	USU_COR
64,6188	39,6013	58,9384	45,4938
70,6249	45,6342	64,395	47,4064
64,2484	43,5412	60,4109	48,6097
71,2663	33,7878	52,4718	35,7512
57,6435	32,1987	50,8143	36,3483
64,7489	44,0721	61,5918	43,5046
66,7538	40,5427	56,4004	40,8341
60,4349	41,1113	61,7787	45,5469
60,0854	24,4543	51,2101	33,9002
58,7004	42,2022	57,772	42,7404
63,9532	51,1428	65,7906	50,6901
69,3859	41,2327	61,8521	44,8027
65,0687	49,9265	58,7896	49,5121
60,9454	40,1733	55,2561	40,1754
57,5605	32,8864	62,7563	39,3537
77,3762	41,3325	60,6667	45,946
59,2103	45,6849	63,6107	50,0219
64,2766	32,5023	53,0574	37,4951
67,4305	45,7729	60,4322	42,6967
72,1898	41,6095	62,6842	47,4671
75,7838	47,3866	65,0358	50,2751
63,6186	26,5791	54,3564	36,6763
61,9027	30,9928	59,2544	43,6985
65,3377	43,654	61,1756	53,0282
74,99	34,9094	51,7311	36,2705
65,1809	33,6252	56,6723	38,1024
65,3551	36,09	60,0942	41,2713
68,4741	47,8383	62,4837	49,5373
59,2229	35,9583	56,1627	42,2148
61,2693	30,5968	50,1595	35,9874
62,2441	39,0514	56,7887	41,2898
64,6091	42,3303	60,1923	45,2172
67,4023	38,2239	62,53	46,0961
65,2089	38,4087	62,6382	45,2434
62,7509	37,3973	54,9593	37,0856
71,8195	41,1555	65,9579	47,5231
56,7604	23,373	39,5549	25,8601
63,3384	34,6439	67,8957	45,6673
58,2328	34,372	56,5405	41,8765
64,2704	48,2378	63,2496	50,1087
62,503	46,8367	62,343	48,9784

Queremos estudiar ahora la relación entre las variables LLAR_BA y USU_COR. Para hacerlo clasificaremos los municipios según el nivel de implantación de la banda ancha y la proporción de usuarios del correo electrónico.
Tabulando los datos iniciales crearemos dos nuevas variables
a) NIV_BA: toma los valores “1_bajo” (menos del 30%), “2_suf” (entre el 30% y el 40%), “3_med” (entre el 40% y el 50%) y “4_bue” (más del 50%),
b) NIV_COR: toma los valores “1_bajo” (menos del 30%), “2_suf” (entre el 30% y el 40%), “3_med” (entre el 40% y el 50%) y “4_bue” (más del 50%).

1) Usando la función “cut” cread las variables NIV_BA y NIV_COR.
2) Construid la tabla de contingencia que permita estudiar la relación entre las dos variables NIV_BA i NIV_COR. Para la construcción de las tablas, podemos usar la función “table”.
3) Utilizando la regla de Laplace, según la cual “probabilidad = casos favorables/casos posibles” y sobre la base de los resultados de la tabla obtenida anteriormente responded a las preguntas siguientes:
    a. Calculad la probabilidad que un municipio tenga un nivel bajo de implantación de la banda ancha.
    b. Calculad la probabilidad que un municipio tenga un nivel bajo de implantación de la banda ancha y un nivel no bajo de usuarios de correo electrónico.
    c. Calculad la probabilidad que un municipio tenga un nivel medio (med) de implantación de la banda ancha si tiene un nivel medio (med) de usuarios de correo electrónico.

Solución

1) Primero introducimos los datos en R:

```
LLAR_ORD<-c(64.6188,70.6249,64.2484,71.2663,57.6435,64.7489,66.7538,60.4349,60.0854,
            58.7004,63.9532,69.3859,65.0687,60.9454,57.5605,77.3762,59.2103,64.2766,
            67.4305,72.1898,75.7838,63.6186,61.9027,65.3377,74.99,65.1809,65.3551,
            68.4741,59.2229,61.2693,62.2441,64.6091,67.4023,65.2089,62.7509,71.8195,
            56.7604,63.3384,58.2328,64.2704,62.503)
LLAR_BA<-c(39.6013,45.6342,43.5412,33.7878,32.1987,44.0721,40.5427,41.1113,24.4543,
           42.2022,51.1428,41.2327,49.9265,40.1733,32.8864,41.3325,45.6849,32.5023,
           45.7729,41.6095,47.3866,26.5791,30.9928,43.654,34.9094,33.6252,36.09,
           47.8383,35.9583,30.5968,39.0514,42.3303,38.2239,38.4087,37.3973,41.1555,
           23.373,34.6439,34.372,48.2378,46.8367)
USU_ORD<-c(58.9384,64.395,60.4109,52.4718,50.8143,61.5918,56.4004,61.7787,51.2101,
           57.772,65.7906,61.8521,58.7896,55.2561,62.7563,60.6667,63.6107,53.0574,
           60.4322,62.6842,65.0358,54.3564,59.2544,61.1756,51.7311,56.6723,60.0942,
           62.4837,56.1627,50.1595,56.7887,60.1923,62.53,62.6382,54.9593,65.9579,
           9.5549,67.8957,56.5405,63.2496,62.343)
USU_COR<-c(45.4938,47.4064,48.6097,35.7512,36.3483,43.5046,40.8341,45.5469,33.9002,
           42.7404,50.6901,44.8027,49.5121,40.1754,39.3537,45.946,50.0219,37.4951,
           42.6967,47.4671,50.2751,36.6763,43.6985,53.0282,36.2705,38.1024,41.2713,
           49.5373,42.2148,35.9874,41.2898,45.2172,46.0961,45.2434,37.0856,47.5231,
           25.8601,45.6673,41.8765,50.1087,48.9784)
```

A continuación creamos las variables NIV_BA y NIV_COR.

```
NIV_BA<-cut(LLAR_BA,breaks=c(0,30,40,50,100),
      labels=c("1_bajo","2_suf","3_med","4_bue"),right=FALSE)


NIV_BA

##  [1] 2_suf  3_med  3_med  2_suf  2_suf  3_med  3_med  3_med  1_bajo 3_med
## [11] 4_bue  3_med  3_med  3_med  2_suf  3_med  3_med  2_suf  3_med  3_med
## [21] 3_med  1_bajo 2_suf  3_med  2_suf  2_suf  2_suf  3_med  2_suf  2_suf
## [31] 2_suf  3_med  2_suf  2_suf  2_suf  3_med  1_bajo 2_suf  2_suf  3_med
## [41] 3_med
## Levels: 1_bajo 2_suf 3_med 4_bue

NIV_COR<-cut(USU_COR,breaks=c(0,30,40,50,100),
labels=c("1_bajo","2_suf","3_med","4_bue"),right=FALSE)
NIV_COR
```

```
##  [1] 3_med  3_med  3_med  2_suf  2_suf  3_med  3_med  3_med  2_suf  3_med
## [11] 4_bue  3_med  3_med  3_med  2_suf  3_med  4_bue  2_suf  3_med  3_med
## [21] 4_bue  2_suf  3_med  4_bue  2_suf  2_suf  3_med  3_med  3_med  2_suf
## [31] 3_med  3_med  3_med  3_med  2_suf  3_med  1_bajo 3_med  3_med  4_bue
## [41] 3_med
## Levels: 1_bajo 2_suf 3_med 4_bue
```

Para poder ver los datos en forma de columna podemos incorporar las variables a una "data.frame" y escribimos las primeras 10 filas:

```
DADES2<-data.frame(LLAR_BA,USU_COR, NIV_BA,NIV_COR)
head(DADES2,10)

##    LLAR_BA USU_COR NIV_BA NIV_COR
## 1  39.6013 45.4938  2_suf   3_med
## 2  45.6342 47.4064  3_med   3_med
## 3  43.5412 48.6097  3_med   3_med
## 4  33.7878 35.7512  2_suf   2_suf
## 5  32.1987 36.3483  2_suf   2_suf
## 6  44.0721 43.5046  3_med   3_med
## 7  40.5427 40.8341  3_med   3_med
## 8  41.1113 45.5469  3_med   3_med
## 9  24.4543 33.9002 1_bajo   2_suf
## 10 42.2022 42.7404  3_med   3_med
```

2) Creamos la tabla de contingencia

```
TA_BA_COR<-addmargins(table(NIV_BA,NIV_COR))
TA_BA_COR

##         NIV_COR
## NIV_BA   1_bajo 2_suf 3_med 4_bue Sum
##   1_bajo      1     2     0     0   3
##   2_suf       0     8     9     0  17
##   3_med       0     0    16     4  20
##   4_bue       0     0     0     1   1
##   Sum         1    10    25     5  41
```

3) a. La probabilidad pedida es: $p(1\_\text{bajo}(\text{NIV\_BA})) = \frac{3}{41} \simeq 0{,}073.$

b. La probabilidad pedida es:

$$p(1\_\text{bajo}(\text{NIV\_BA}) \cap (1\_\text{bajo}(\text{NIV\_COR}))^c) =$$

$$p(1\_\text{bajo}(\text{NIV\_BA}))\text{-}p(1\_\text{bajo}(\text{NIV\_BA}) \cap 1\_\text{bajo}(\text{NIV\_COR})) = \frac{3}{41} - \frac{1}{41} = \frac{2}{41} \simeq 0{,}049.$$

c. La probabilidad pedida es: $p(3\_\text{med}(\text{NIV\_BA}) | \, 3\_\text{med}(\text{NIV\_COR})) = \frac{16}{25} = 0{,}64.$

Actividad 3: Número de mensajes no deseados que recibe una empresa.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

La siguiente tabla nos indica el número de mensajes “SPAM” que reciben en un día cualquiera los empleados de una determinada empresa:

Número “SPAM”	0	1	2	3	4	5	6	7
Número de empleados	7	11	10	7	1	2	1	1

La tabla anterior se ha de interpretar así: 7 empleados no reciben ningún “SPAM” en el día considerado, 11 empleados reciben 1 “SPAM” en el día considerado, etc. Consideramos la variable X= “número de mensajes “SPAM” que recibe un empleado cualquiera de esta empresa por día”. Clasificamos los empleados de esta empresa de la forma siguiente: A: “empleados

que reciben hasta 2 “SPAMS” por día”. B: “empleados que reciben de 3 a 5 “SPAMS” por día (ambos inclusive)” y C: “empleados que reciben 6 o más “SPAMS” por día”. Nos dicen también que los únicos usuarios del sistema operativo Ventanas de la empresa son los 8 empleados que reciben cero o 7 “SPAMS” por día. Se pide:

a) Construid la tabla de contingencia que permita estudiar la relación entre las dos variables “número de spams recibidos” y “sistema operativo usado”.
b) Encontrad la probabilidad de que un empleado escogido al azar use “Ventanas” y sea del grupo A.
c) Encontrad la probabilidad de que un empleado que no use “Ventanas” sea del grupo C.
d) ¿Los sucesos “usar Ventanas” y “ser del grupo B” son independientes? ¿Por qué?

Solución

a) En primer lugar, introducimos el número de mensajes SPAM y cuántos empleados reciben este número de correos

```
SPAM<-c(0,1,2,3,4,5,6,7)
SPAM

## [1] 0 1 2 3 4 5 6 7

EMPLEADOS<-c(7,11,10,7,1,2,1,1)
EMPLEADOS

## [1]  7 11 10  7  1  2  1  1
```

A continuación, creamos la variable de estudio, “número de SPAMS que recibe un empleado” repitiendo los valores de la variable “SPAM” tantas veces como indica la variable “EMPLEADOS”; esto se puede hacer con la función “rep” mediante la que repetimos el valor 0, 7 veces, el valor 1, 11 veces y así sucesivamente.

```
OBS_SPAM<-rep(SPAM,EMPLEADOS)
OBS_SPAM

##  [1] 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3
## [36] 4 5 5 6 7
```

A continuación creamos la variable NUM_SPAMS que valdrá A si un empleado recibe hasta 2 SPAMS por día, B si de 3 a 5 SPAMS por día (ambos inclusive) y C si recibe 6 o más “SPAMS” por día”. Para hacerlo,

```
NUM_SPAMS<-cut(OBS_SPAM,breaks=c(-1, 2.5,5.5, 7.5), labels=c("A","B","C"),right=FALSE)
NUM_SPAMS

##  [1] A A A A A A A A A A A A A A A A A A A A A A A A A A A B B B B B B B
## [36] B B B C C
## Levels: A B C
```

A continuación creamos la variable VENTANAS de una forma muy similar a la creación de la variable anterior pero como R no admite “labels” repetidas, crearemos una variable con los valores “SI”, “NO” y “SI2” y luego convertiremos el “SI2” en “SI”:

```
VENTANAS3<-cut(OBS_SPAM,breaks=c(0, 0.5 ,6.5, 8), labels=c("SI","NO","SI2"),right=FALSE)
VENTANAS3

##  [1] SI  SI  SI  SI  SI  SI  SI  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO
## [18] NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO  NO
## [35] NO  NO  NO  NO  NO  SI2
## Levels: SI NO SI2

VENTANAS<-replace(VENTANAS3,VENTANAS3=="SI2","SI")
VENTANAS
```

```
##  [1] SI SI SI SI SI SI SI NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO
## [24] NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO SI
## Levels: SI NO SI2
```

En forma de columnas

```
DADES3<-data.frame(OBS_SPAM,NUM_SPAMS,VENTANAS)
head(DADES3,10)

##    OBS_SPAM NUM_SPAMS VENTANAS
## 1         0         A       SI
## 2         0         A       SI
## 3         0         A       SI
## 4         0         A       SI
## 5         0         A       SI
## 6         0         A       SI
## 7         0         A       SI
## 8         1         A       NO
## 9         1         A       NO
## 10        1         A       NO
```

Como siempre creamos la tabla de contingencia con “table”

```
TA_VAN<-addmargins(table(NUM_SPAMS, VENTANAS))
TA_VAN

##          VENTANAS
## NUM_SPAMS SI NO SI2 Sum
##       A    7 21   0  28
##       B    0 10   0  10
##       C    1  1   0   2
##       Sum  8 32   0  40
```

b) La probabilidad pedida vale: $p(\mathrm{SI} \cap \mathrm{A}) = \frac{7}{40} = 0{,}175.$

c) La probabilidad pedida vale: $p(\mathrm{C} \mid \mathrm{NO}) = \frac{1}{32} = 0{,}03125.$

d) Para estudiar la independencia, hallemos las probabilidades siguientes: $p(\mathrm{SI})$, $p(\mathrm{B})$ y $p(\mathrm{SI} \cap \mathrm{B})$: $p(\mathrm{SI}) = \frac{8}{40} = 0{,}2$, $p(\mathrm{B}) = \frac{10}{40} = 0{,}25$, $p(\mathrm{SI} \cap \mathrm{B}) = \frac{0}{40} = 0.$ Como $p(\mathrm{SI} \cap \mathrm{B}) \neq p(\mathrm{SI}) \cdot p(\mathrm{B})$ ya que $0 \neq 0{,}2 \cdot 0{,}25,$ concluimos que no son independientes.

Actividad 4: Cómputo del tiempo de CPU.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

En la tabla siguiente se muestran los resultados de un test que consiste en ejecutar aleatoriamente diferentes programas en un ordenador y medir el tiempo de CPU consumido (en milisegundos) para cada programa (variable TEMP_CPU). También conocemos la longitud del código de cada uno de los programas ejecutados (Variable LONG_CODI).

TEMP_CPU	LONG_CODI
127	146
83	80
85	60
93	90
103	58
80	88
71	106

112	150
116	38
62	195
123	121
90	109
103	148
63	96
116	92
103	50
98	19
71	84
103	90
101	72
91	97
125	147
117	108
126	135
112	111
91	79
89	121
105	194
110	41
149	110
98	120
112	45
131	109
147	155
92	169
85	268
55	97
89	81
52	78
91	47
66	108
76	180
102	40
97	184
87	29
111	192
70	63
143	117
108	73
81	53
107	103
103	44
99	91
135	131
123	107
103	36
129	56
115	85
80	23
93	71
117	133
90	48
94	48
70	74
83	82
109	80
65	38
115	107
100	51
86	78
89	200
134	96
96	155
67	61
138	78
117	31
75	54
111	87
111	152
104	140
66	56
126	112
121	136
101	41

118	148
67	171
117	114
92	73
107	71
122	196
48	39
97	67
94	40
125	81
120	169
112	39
97	85
89	58
112	37
87	48

En este problema os planteamos estudiar la relación entre el tiempo de ejecución del programa y la longitud del código desde el punto de vista del estudio de dos variables categóricas. Concretamente estudiaremos la relación entre la variable CLAS_TEMP y la variable CLAS_CODI. La variable CLAS_CODI es una variable categórica con las categorías "C_supercorto", "C_corto", "C_largo" y "C_superlargo", categorías que estarán delimitadas por el mínimo, Q1, Q2, Q3 y el máximo de la variable LONG_CODI (es decir, C_supercorto corresponde a aquellos códigos que tienen longitud entre el mínimo y Q1 de la variable LONG_CODI, con Q1 incluido, y así sucesivamente...; en el caso que algún cuartil sea no entero, lo truncaremos). Para definir la variable categórica CLAS_TEMP, distribuiremos los tiempos de ejecución en 3 categorías: "T_corto" (tiempo en el intervalo [47,81]), "T_medio" (tiempo en el intervalo (81,114]), "T_largo" (tiempo en el intervalo (114,149]).

a) Construid la tabla de contingencia que permita estudiar la relación entre las variables CLAS_TEMP y CLAS_CODI. En las filas pondremos el tiempo y en las columnas indicaremos la longitud, con lo que tendremos una tabla como ésta:

	C_supercorto	**C_corto**	**C_largo**	**C_superlargo**	**Total**
T_corto					
T_medio					
T_largo					
Total					

Para la construcción de la tabla se puede usar la función de R "table".
b) Utilizando la regla de Laplace, según la que "probabilidad = casos favorables/casos posibles" y sobre la base de los resultados de la tabla obtenida anteriormente, responded las siguientes preguntas:

1) Estimad la probabilidad de que un programa tenga un código superlargo
2) Estimad la probabilidad que un programa tenga un código superlargo y un tiempo de ejecución corto.
3) Estimad la probabilidad de que un programa superlargo tenga un tiempo de ejecución corto.
4) Estimad la probabilidad de que un programa tenga un tiempo de ejecución corto y sea largo o superlargo
5) Estimad la probabilidad de que un programa que tiene un tiempo de ejecución corto, sea largo o superlargo.

Solución

a) Primero introducimos los datos en R:

```
TEMP_CPU<-c(127,83,85,93,103,80,71,112,116,62,123,90,103,63,116,103,98,71,103,101,91,
            125,117,126,112,91,89,105,110,149,98,112,131,147,92,85,55,89,52,91,66,76,
            102,97,87,111,70,143,108,81,107,103,99,135,123,103,129,115,80,93,117,90,
            94,70,83,109,65,115,100,86,89,134,96,67,138,117,75,111,111,104,66,126,121,
            101,118,67,117,92,107,122,48,97,94,125,120,112,97,89,112,87)
```

```
LONG_CODI<-c(146,80,60,90,58,88,106,150,38,195,121,109,148,96,92,50,19,84,90,72,97,
             147,108,135,111,79,121,194,41,110,120,45,109,155,169,268,97,81,78,47,
             108,180,40,184,29,192,63,117,73,53,103,44,91,131,107,36,56,85,23,71,133,
             48,48,74,82,80,38,107,51,78,200,96,155,61,78,31,54,87,152,140,56,112,
             136,41,148,171,114,73,71,196,39,67,40,81,169,39,85,58,37,48)
```

A continuación creamos la variable CLAS_TEMP

```
CLAS_TEMP<-cut(TEMP_CPU,breaks=,c(47,81,114,150),
labels=c("T_corto","T_medio","T_largo"),right=FALSE)
```

y la CLAS_CODI usando los cuartiles de la variable; para ello usaremos la función “quantile” ya que “quantile(LONG_CODI, probs = **c**(0.25, 0.5, 0.75,1)” devuelve los cuantiles correspondientes a las probabilidades 0,25 (Q1), 0.5 (Q2=mediana), 0.75 (Q3) y 1 (máximo); de hecho estos cuartiles son Q1=56, Q2=86 y Q3=121. El mínimo es 19.

```
CLAS_CODI<-cut(LONG_CODI,breaks=c(19,56,86,121,268),
labels=c("C_supercorto","C_corto","C_largo","C_superlargo"),include.lowest=TRUE)
```

En forma de columnas

```
DADES4<-data.frame(TEMP_CPU,LONG_CODI,CLAS_TEMP,CLAS_CODI)
head(DADES4,10)

##    TEMP_CPU LONG_CODI CLAS_TEMP    CLAS_CODI
## 1       127       146   T_largo C_superlargo
## 2        83        80   T_medio      C_corto
## 3        85        60   T_medio      C_corto
## 4        93        90   T_medio      C_largo
## 5       103        58   T_medio      C_corto
## 6        80        88   T_corto      C_largo
## 7        71       106   T_corto      C_largo
## 8       112       150   T_medio C_superlargo
## 9       116        38   T_largo C_supercorto
## 10       62       195   T_corto C_superlargo
```

Hacemos ahora la tabla de contingencia

```
TAULA_TEMPS_CODI<-addmargins(table(CLAS_TEMP, CLAS_CODI))
TAULA_TEMPS_CODI

##          CLAS_CODI
## CLAS_TEMP C_supercorto C_corto C_largo C_superlargo Sum
##   T_corto            5       6       5            3  19
##   T_medio           17      16      10           11  54
##   T_largo            2       4      11           10  27
##   Sum               24      26      26           24 100
```

b)

1. La probabilidad pedida vale: $p(\text{C\_superlargo})=\frac{24}{100}=0,24.$
2. La probabilidad pedida vale: $p(\text{C\_superlargo}\cap\text{T\_corto})=\frac{3}{100}=0,03.$
3. La probabilidad pedida vale: $p(\text{T\_corto}\mid\text{C\_superlargo})=\frac{3}{24}=0,125.$
4. La probabilidad pedida vale:

$$p(\text{T\_corto} \cap (\text{C\_largo} \cup \text{C\_superlargo}))=$$
$$p(\text{T\_corto} \cap \text{C\_largo}) + p(\text{T\_corto} \cap \text{C\_superlargo})=$$
$$\frac{5}{100} + \frac{3}{100} = \frac{8}{100} = 0,08.$$

5. La probabilidad pedida vale:

$$p((\text{C\_largo} \cup \text{C\_superlargo}) \mid \text{T\_corto})=$$
$$p(\text{C\_largo} \mid \text{T\_corto}) + p(\text{C\_superlargo} \mid \text{T\_corto})=$$
$$\frac{5}{19} + \frac{3}{19} = \frac{8}{19} \simeq 0,421$$

Actividad 5: Testeo de un programa.

Concepto de probabilidad. Independencia.

Un programa de ordenador que contiene un error se testea en 3 tests diferentes. Si el test detecta el error, devuelve un 1 y en caso contrario, devuelve un 0. La tabla siguiente muestra el resultado del testeo del programa en 100 veces en los que se ejecutaron los tres tests.

TEST 1	TEST 2	TEST 3
0	0	1
0	0	0
0	0	0
1	0	1
0	0	1
1	1	1
0	0	0
0	1	0
0	1	0
0	0	0
0	0	1
1	0	0
0	0	1
0	0	0
0	1	1
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1	0
0	0	1
0	0	0
0	0	1
1	0	1
1	1	0
0	0	0
0	0	0
0	1	0
0	1	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	1
0	0	1
0	1	1
1	0	1
0	0	1
0	0	1
0	0	0
0	0	0
0	0	0
0	0	0
0	0	1
0	1	0

0	0	0
0	0	0
0	0	0
0	0	0
0	0	1
0	0	0
0	0	1
0	0	0
0	0	1
0	0	0
0	0	0
0	0	1
0	0	0
0	0	0
0	1	0
0	0	0
0	0	1
0	0	0
0	0	1
1	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1	1
0	0	0
0	0	0
0	0	0
0	0	1
0	1	0
0	0	0
0	0	0
0	0	0
0	0	1
0	1	1
0	1	1
0	0	1
0	0	0
0	0	0
0	0	0
0	1	0
0	0	1
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Se pide:

a) Estimar la probabilidad de que cada uno de los tests detecte el error.
b) Estimar la probabilidad de que el TEST 1 y el TEST 2 detecten el error al mismo tiempo.
¿Podemos sospechar que los tests TEST1 y TEST2 son independientes?
c) Estimar la probabilidad de que los tres tests detecten el error al mismo tiempo.
¿Podemos sospechar que los tres tests son independientes?

Solución

a) Introducimos los datos en R:

```
TEST1<-c(0,0,0,1,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,
         0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,
         0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

Seguidamente vamos a calcular el número de veces que cada test detecta el error. Para ello, vamos a sumar el valor de cada variable, lo que nos indica cuantas veces el test correspondiente detecta el error. Podemos hacer

```
S1<-sum(TEST1)
S1

## [1] 7

S2<-sum(TEST2)
S2

## [1] 16

S3<-sum(TEST3)
S3

## [1] 30
```

con lo que las probabilidades pedidas son:

$$p(\text{TEST 1}) = \frac{7}{100} = 0,07,\ p(\text{TEST 2}) = \frac{16}{100} = 0,16,\ p(\text{TEST 3}) = \frac{30}{100} = 0,3.$$

b) Los tests TEST 1 y TEST 2 detectan el error al mismo tiempo si nos aparece un 1 en las columnas TEST 1 y TEST 2. Por tanto, tenemos que contar las veces en que aparece un 1 simultáneamente en las dos columnas. Una forma sencilla de hacer esta operación anterior es crear una nueva variable que sea el producto de las variables TEST 1 y TEST 2. De esta forma, esta tercera variable valdrá 1 sólo en el caso en que las columnas TEST 1 y TEST 2 sean 1 y en caso contrario valdrá 0. Sumando esta tercera variable, sabremos las veces en que los tests TEST 1 y TEST 2 han detectado el error. Podemos hacerlo de esta sencilla manera:

```
TEST12<-TEST1*TEST2
TEST12

##   [1] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
##  [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Para sumar la columna TEST12, hacemos la misma operación que hemos realizado en el apartado a) con las columnas TEST1, TEST2 y TEST3. El resultado de R es:

```
S12<-sum(TEST12)
S12

## [1] 2
```

La probabilidad de que los tests TEST 1 y TEST 2 detecten el error al mismo tiempo valdrá:

$$p(\text{TEST 1} \cap \text{TEST 2}) = \frac{2}{100} = 0,02.$$

Para que los tests TEST 1 y TEST 2 sean independientes, tiene que verificarse: $p(\text{TEST 1} \cap \text{TEST 2}) = p(\text{TEST 1}) \cdot p(\text{TEST 2}),$ pero $\frac{2}{100} \neq \frac{7}{100} \cdot \frac{16}{100} = \frac{7}{625}.$ Por tanto, no son independientes.

b) En general, tres sucesos A, B y C son independientes si se verifican todas las condiciones siguientes:

$$p(\text{A} \cap \text{B}) = p(\text{A}) \cdot p(\text{B}),\ p(\text{A} \cap \text{C}) = p(\text{A}) \cdot p(\text{C}),$$

$$p(\text{B} \cap \text{C}) = p(\text{B}) \cdot p(\text{C}),\ p(\text{A} \cap \text{B} \cap \text{C}) = p(\text{A}) \cdot p(\text{B}) \cdot p(\text{C}).$$

Entonces para que tres sucesos sean independientes, aparte de cumplirse que la probabilidad del producto sea el producto de las probabilidades, debe verificarse que sean independientes dos a dos. De aquí podemos deducir que los tres tests no son independientes al no serlo los tests TEST 1 y TEST 2.

Seguidamente, vamos a calcular las probabilidades de que los tests TEST 1 y TEST 3 detecten el error al mismo tiempo, de que los tests TEST 2 y TEST 3 detecten el error al mismo tiempo y de que los tres tests detecten el error al mismo tiempo usando el mismo método que en el

apartado anterior. Llamaremos a las variables TEST13, TEST23 y TEST123 y a sus sumas S13, S23 y S123 respectivamente.

```
TEST13<-TEST1*TEST3
TEST13

##   [1] 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
##  [36] 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

S13<-sum(TEST13)
S13

## [1] 4

TEST23<-TEST2*TEST3
TEST23

##   [1] 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [36] 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0

S23<-sum(TEST23)
S23

## [1] 6

TEST123<-TEST1*TEST2*TEST3
TEST123

##   [1] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

S123<-sum(TEST123)
S123

## [1] 1
```

La probabilidad de que los tres tests detecten el error al mismo tiempo vale: $p(\text{TEST } 1 \cap \text{TEST } 2 \cap \text{TEST } 3) = \frac{1}{100} = 0,01.$

Actividad 6: Conocimiento de los lenguajes de programación actuales por parte de los estudiantes de ciencias de la computación.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

La siguiente tabla nos indica el conocimiento de los lenguajes de programación Java y qué tipo de especialidad han elegido estudiar por parte de 25 estudiantes de ciencias de la computación de 0 (ningún conocimiento) a 100 (máximo dominio del lenguaje):

Java	Especialidad
50.27	Gestión
50.51	Sistemas
49.58	Gestión
50.09	Sistemas
50.27	Gestión
50.72	Sistemas
49.22	Sistemas
50.09	Sistemas
51.37	Gestión
49.93	Gestión
50.58	Gestión
48.92	Gestión

50.21	Gestión
49.81	Gestión
50.69	Gestión
48.50	Gestión
48.77	Gestión
48.56	Gestión
49.39	Gestión
50.79	Gestión
48.67	Sistemas
51.03	Sistemas
49.73	Sistemas
52.20	Sistemas
50.32	Sistemas

Diremos que un estudiante tiene un nivel alto de java si su conocimiento supera el valor 52, un estudiante tiene un valor medio de java si su conocimiento está entre 50 y 52 y en caso contrario, diremos que el estudiante tiene un nivel bajo en java. Se pide:

a) Construid la tabla de contingencia que permita estudiar la relación entre las variables "nivel del estudiante en java" y "especialidad elegida".
b) Utilizando la regla de Laplace, según la que "probabilidad = casos favorables/casos posibles" y sobre la base de los resultados de la tabla obtenida anteriormente, responded las siguientes preguntas:
    a. Estimad la probabilidad de que un estudiante tenga un nivel medio en java.
    b. Estimad la probabilidad que un estudiante tenga un nivel alto en java y estudie la especialidad de Gestión.
    c. Estimad la probabilidad de que un estudiante de Sistemas tenga un nivel bajo en java.
    d. Los sucesos "tener un nivel medio en java" y estudiar "informática de gestión", ¿son independientes?

Solución

a) En primer lugar, introducimos los datos en R:

```
Java<-c(50.27,50.51,49.58,50.09,50.27,50.72,49.22,50.09,51.37,49.93,50.58,48.92,
        50.21,49.81,50.69,48.50,48.77,48.56,49.39,50.79,48.67,51.03,49.73,52.20,50.32)
Especialidad<-c("Gestión","Sistemas","Gestión","Sistemas","Gestión","Sistemas",
                "Sistemas","Sistemas","Gestión","Gestión","Gestión","Gestión",
                "Gestión","Gestión","Gestión","Gestión","Gestión","Gestión",
                "Gestión","Gestión","Sistemas","Sistemas","Sistemas","Sistemas",
                "Sistemas")
```

Antes de crear la variable categórica que nos dará el nivel de Java de los estudiantes, hallemos el máximo de la variable Java. Para ello, usaremos "summary" que también nos da otros resúmenes numéricos

```
SJ<-summary(Java)
SJ

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   48.50   49.39   50.09   50.01   50.58   52.20
```

El máximo, como vemos, vale 52,2.

Construimos ahora la variable categórica perdida con el nombre NJ

```
NJ<-cut(Java,breaks=,c(0,50,52,55), labels=c("bajo","medio","alto"),right=FALSE)
NJ

##  [1] medio medio bajo  medio medio medio bajo  medio medio bajo  medio
## [12] bajo  medio bajo  medio bajo  bajo  bajo  bajo  medio bajo  medio
## [23] bajo  alto  medio
## Levels: bajo medio alto
```

```
TEJ<-addmargins(table(Especialidad,NJ))
TEJ

##             NJ
## Especialidad bajo medio alto Sum
##     Gestión     8     7    0  15
##     Sistemas    3     6    1  10
##     Sum        11    13    1  25
```

.

b) a. La probabilidad pedida vale: $p(\text{medio}) = \frac{13}{25} = 0,52.$

b. La probabilidad pedida vale: $p(\text{alto} \cap \text{Gestión}) = \frac{0}{25} = 0.$

c. La probabilidad pedida vale: $p(\text{bajo} \mid \text{Sistemas}) = \frac{3}{10} = 0,3.$

d. Para ver si los sucesos "tener nivel medio en Java" y "estudiar Informática de Gestión" son independientes hemos de ver si $p(\text{medio} \cap \text{Gestión}) = p(\text{medio}) \cdot p(\text{Gestión}).$ Hallemos las probabilidades anteriores:

$p(\text{medio} \cap \text{Gestión}) = \frac{7}{25},\ p(\text{medio}) = \frac{13}{25},\ p(\text{Gestión}) = \frac{15}{25},$ pero

$0,28 = \frac{7}{25} \neq \frac{13}{25} \cdot \frac{15}{25} = 0,312.$ Por tanto, no son independientes.

Actividad 7: Número de cortes en la red de una empresa de servicios de Internet.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

Una pequeña empresa que se dedica a dar servicio de Internet tiene durante 50 días el número de cortes siguientes en la red: 2, 1, 0, 0, 1, 1, 1, 2, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 1, 2, 0, 0, 0, 2, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 2, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0. Los 30 primeros cortes se produjeron por la noche, entre las 0.00 h. y las 8.00 h. de la mañana, los 10 siguientes se produjeron por la mañana, entre las 8.00 h. y las 14.00 h. y los 10 últimos se produjeron por la tarde, entre las 14.00 h. y las 0.00 h. Se pide:

a) Construid la tabla de contingencia que permita estudiar la relación entre las variables "número de cortes de la red" y "franja horaria".
b) Utilizando la regla de Laplace, según la que "probabilidad = casos favorables/casos posibles" y sobre la base de los resultados de la tabla obtenida anteriormente, responded las siguientes preguntas:
    a. Estimad la probabilidad de que se produzcan 2 cortes en un día.
    b. Estimad la probabilidad que se produzca un corte por la noche.
    c. Estimad la probabilidad de que no se produzcan cortes durante el día.
    d. Si sabemos que se han producido 2 cortes, estimad la probabilidad de que éstos se produzcan por la tarde.
    e. Los sucesos "no tener cortes" y "estar por la noche", ¿son independientes?

Solución

a) Primeramente introducimos los datos en R:

Para el número de cortes:

```
NCOR<-c(2, 1, 0, 0, 1, 1, 1, 2, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 1, 2, 0, 0, 0, 2,
        0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 2, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0)
```

Para la franja horaria (TIPUS) tenemos que crear una variable que empiece por 30 veces “Noche”, luego 10 “Mañana” y 10 “Tarde”, cosa que se puede hacer fácilmente con la función “rep”

```
TIPUS<-rep(c("Noche","Mañana","Tarde"),c(30,10,10))
TIPUS

##  [1] "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"
##  [8] "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"
## [15] "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"
## [22] "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"  "Noche"
## [29] "Noche"  "Noche"  "Mañana" "Mañana" "Mañana" "Mañana" "Mañana"
## [36] "Mañana" "Mañana" "Mañana" "Mañana" "Mañana" "Tarde"  "Tarde"
## [43] "Tarde"  "Tarde"  "Tarde"  "Tarde"  "Tarde"  "Tarde"  "Tarde"
## [50] "Tarde"
```

Para construir la tabla de contingencia hacemos

```
TCD<-addmargins(table(NCOR,TIPUS))
TCD

##      TIPUS
## NCOR  Mañana Noche Tarde Sum
##   0        8    15     9  32
##   1        0    10     1  11
##   2        2     5     0   7
##   Sum     10    30    10  50
```

b) a. La probabilidad pedida vale: $p(2 \text{ cortes}) = \frac{7}{50} = 0,14.$

b. La probabilidad pedida vale: $p(1 \text{ corte} \cap \text{Noche}) = \frac{10}{50} = 0,2.$

c. La probabilidad pedida vale: $p(0 \text{ cortes}) = \frac{32}{50} = 0,64.$

d. La probabilidad pedida vale: $p(\text{Tarde} \mid 2 \text{ cortes}) = \frac{0}{7} = 0.$

e. Para ver si los sucesos “no tener cortes” y “estar por la noche” son independientes, hemos de ver si se verifica $p(0 \text{ cortes} \cap \text{Noche}) = p(0 \text{ cortes}) \cdot p(\text{Noche}).$ Hallemos las probabilidades anteriores: $p(0 \text{ cortes} \cap \text{Noche}) = \frac{15}{50}$, $p(0 \text{ cortes}) = \frac{32}{50}$, $p(\text{Noche}) = \frac{30}{50}$, pero $0,3 = \frac{15}{50} \neq \frac{32}{50} \cdot \frac{30}{50} = 0,384.$ Por tanto, no son independientes.

Actividad 8: Filtro de detección de mensajes no deseados.

Concepto de probabilidad. Probabilidad condicionada. Independencia. Teorema de la Probabilidad Total. Teorema de Bayes.

La tabla siguiente nos muestra el resultado de detección de correo no deseado (“SPAM”) por un filtro instalado en el servidor de correo de una empresa. La primera columna nos indica qué tipo de correo entró por el servidor (“SPAM”, no deseado o “HAM”, correo correcto) y la segunda columna nos indica lo que detectó el filtro.

CORREO ENTRANTE	CORREO DETECTADO COMO…
SPAM	SPAM

SPAM	HAM
SPAM	SPAM
HAM	HAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
HAM	HAM
SPAM	SPAM
SPAM	SPAM
SPAM	HAM
SPAM	HAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	HAM
SPAM	HAM
HAM	SPAM
SPAM	HAM
SPAM	SPAM
SPAM	SPAM
HAM	SPAM
HAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	HAM
HAM	SPAM
HAM	SPAM
SPAM	SPAM
SPAM	HAM
SPAM	HAM
SPAM	SPAM
SPAM	HAM
SPAM	HAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	SPAM
SPAM	HAM
SPAM	HAM
HAM	HAM
HAM	SPAM
SPAM	HAM
SPAM	SPAM
SPAM	HAM
SPAM	SPAM
SPAM	HAM

Se pide:

a) Construid la tabla de contingencia que permita estudiar la relación entre las variables "correo entrante" y "correo detectado".
b) Estimad la probabilidad de que haya entrado un correo "SPAM" en el servidor.
c) Estimad la probabilidad de que el filtro haya detectado un correo como "HAM".
d) Estimad la probabilidad de que el filtro haya actuado correctamente.
e) Los sucesos "ser SPAM" y "el filtro ha detectado HAM", ¿son independientes?
f) Hallar la probabilidad de un falso positivo.

Solución

a) Primeramente Introducimos los datos en R:

```
ENTRA<-c("SPAM","SPAM","SPAM","HAM","SPAM","SPAM","SPAM","HAM","SPAM","SPAM",
         "SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","HAM","SPAM",
         "SPAM","SPAM","HAM","HAM","SPAM","SPAM","SPAM","SPAM","SPAM","HAM",
         "HAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM",
         "SPAM","SPAM","SPAM","HAM","HAM","SPAM","SPAM","SPAM","SPAM","SPAM")
DETECTA<-c("SPAM","HAM" ,"SPAM","HAM" ,"SPAM","SPAM","SPAM","HAM" ,"SPAM",
```

```
            "SPAM","HAM" ,"HAM" ,"SPAM","SPAM","SPAM","SPAM","HAM" ,"HAM" ,
            "SPAM","HAM" ,"SPAM","SPAM","SPAM","SPAM","SPAM","SPAM","SPAM",
            "SPAM","HAM" ,"SPAM","SPAM","SPAM","HAM" ,"HAM" ,"SPAM","HAM" ,
            "HAM" ,"SPAM","SPAM","SPAM","SPAM","HAM" ,"HAM" ,"HAM" ,"SPAM",
            "HAM" ,"SPAM","HAM" ,"SPAM","HAM")
```

Para construir la tabla de contingencia hacemos

```
TED<-addmargins(table(ENTRA,DETECTA))
TED

##       DETECTA
## ENTRA  HAM SPAM Sum
##   HAM    3    6   9
##   SPAM  16   25  41
##   Sum   19   31  50
```

b) La probabilidad pedida vale: $p(\text{SPAM(ENTRANTE)}) = \frac{41}{50} = 0,82.$

c) La probabilidad pedida vale: $p(\text{HAM(FILTRO)}) = \frac{19}{50} = 0,38.$

d) La probabilidad pedida vale:

$$p(\text{SPAM(ENTRANTE)} \cap \text{SPAM(FILTRO)}) + p(\text{HAM(ENTRANTE)} \cap \text{HAM(FILTRO)}) = \frac{25}{50} + \frac{3}{50} = \frac{28}{50} = 0,56.$$

e) Para estudiar la independencia de los sucesos hay que ver si se verifica: $p(\text{SPAM(ENTRANTE)} \cap \text{HAM(FILTRO)}) = p(\text{SPAM(ENTRANTE)}) \cdot p(\text{HAM(FILTRO)}).$

f) Hallemos las probabilidades anteriores:

$$p(\text{SPAM(ENTRANTE)} \cap \text{HAM(FILTRO)}) = \frac{16}{50},$$

$$p(\text{SPAM(ENTRANTE)}) = \frac{41}{50},\ p(\text{HAM(FILTRO)}) = \frac{19}{50},$$

pero $0,32 = \frac{16}{50} \neq \frac{41}{50} \cdot \frac{19}{50} = 0,3116.$ Por tanto, no son independientes.

g) La probabilidad pedida vale:

$$p(\text{HAM(ENTRANTE)} \cap \text{SPAM(FILTRO)}) = \frac{6}{50} = 0,12.$$

Direcciones de interés

http://en.wikipedia.org/wiki/Probability
Definición, tratamiento matemático, aplicaciones e historia de la probabilidad en la Wikipedia.

http://wiki.stat.ucla.edu/socr/index.php/EBook#Chapter_III:_Probability
Libro on line sobre probabilidades y estadística. Muy recomendable.

http://www.math.csusb.edu/faculty/stanton/m262/index.html
Applets de java sobre probabilidades y estadística.

http://math.dartmouth.edu/~prob/prob/prob.pdf
Libro on line sobre probabilidades con un enfoque muy práctico y con muchos ejemplos.