

# T1. Taller encuestas. Descriptivos\*

VIII Jornadas R Albacete 2016\*\*

Álvaro Hernández Vicente, Elvira Ferre Jaén, Antonio José Perán Orcajada, Ana Belén  
Marín Valverde, Antonio Maurandi López\*\*\*

17 de noviembre de 2016

## Índice

<b>1. Lectura de datos</b>	<b>1</b>
<b>2. Descriptivos con paquete tables</b>	<b>1</b>
2.1. Variable sexo . . . . .	2
2.2. Variable actividadS . . . . .	2
2.3. Variable ingresos . . . . .	2
2.4. Variable origen . . . . .	3
2.5. Variable nivelIngles . . . . .	3
<b>3. Descriptivos con paquete likert</b>	<b>4</b>
3.1. Gráficos de barras . . . . .	4
3.2. Gráficos de densidad . . . . .	7
3.3. Gráficos de calor . . . . .	10
<b>4. Correlaciones</b>	<b>13</b>
4.1. Matriz de correlaciones . . . . .	13
4.2. Gráficos de correlaciones . . . . .	14
<b>Referencias y bibliografía</b>	<b>16</b>

## 1. Lectura de datos

Se leen los datos `saeraq.Rdata`, que provienen del fichero `raq.dat` utilizado en (A. Field, Miles, and Field 2012) (disponible en la web del libro).

Y

```
load( "saeraq.RData" )
```

## 2. Descriptivos con paquete tables

Se realizan descriptivos de cada variable con el paquete `tables` (Murdoch (2016)).

```
# library( tables )  
op <- booktabs()
```

---

\* *doc:T1\_descriptivos.Rmd*

\*\* <http://r-es.org/8jornadasR/>

\*\*\* Servicio de Apoyo Estadístico; alvarohv@um.es, elvira@um.es, antoniojose.peran@um.es, anabelen.marin4@um.es, amaurandi@um.es



```
# Función para imprimir tablas en latex
tablaLatex <- function( tabla, caption = NULL ){
  cat( '\\begin{table} \\centering\n' )
  if( !is.null( caption ) ) cat( paste0( '\\caption{', caption, '}\n' ) )
  latex( tt )
  cat( '\\end{table}' )
}
```

## 2.1. Variable sexo

```
tt <- tabular( ~ ( Sexo = sexo ) + ( Total = 1 ), data = df )

# html( tt, options = htmloptions( HTMLcaption = "Variable sexo", pad = TRUE ) )
tablaLatex( tt, caption = "Variable \\texttt{sexo}." )
```

Tabla 1: Variable `sexo`.

Sexo		
Mujer	Hombre	Total
1927	644	2571

## 2.2. Variable actividadS

```
tt <- tabular( ~ ( `Actividad S` = actividadS ) + ( Total = 1 ), data = df )

# html( tt, options = htmloptions( HTMLcaption = "Variable actividadS", pad = TRUE ) )
tablaLatex( tt, "Variable \\texttt{actividadS}." )
```

Tabla 2: Variable `actividadS`.

Actividad S				
Nada	Poco	Mucho	Muchísimo	Total
613	658	669	631	2571

## 2.3. Variable ingresos

```
ic1 <- function(x){
  mean( x ) - qt( 0.975, df = length( x ) - 1 ) * sd( x ) / sqrt( length( x ) )
}

ic2 <- function(x){
  mean( x ) + qt( 0.975, df = length( x ) - 1 ) * sd( x ) / sqrt( length( x ) )
}

tt <- tabular( ~ ( Ingresos = ingresos ) * ( ic1 + mean + sd + ic2 + median ) +
  ( Total = 1 ), data = df )
```



```
# html( tt, options = htmloptions( HTMLcaption = "Variable ingresos", pad = TRUE ) )
tablaLatex( tt, "Variable \\texttt{ingresos}." )
```

Tabla 3: Variable ingresos.

Ingresos					
ic1	mean	sd	ic2	median	Total
29320	29672	9095	30024	27771	2571

## 2.4. Variable origen

```
tt <- tabular( ~ ( Origen = origen ) + ( Total = 1 ), data = df )

# html( tt, options = htmloptions( HTMLcaption = "Variable origen", pad = TRUE ) )
tablaLatex( tt, "Variable \\texttt{origen}." )
```

Tabla 4: Variable origen.

Origen			
Albacete	Murcia	Helsinki	Total
1178	1112	281	2571

## 2.5. Variable nivelIngles

```
tt <- tabular( ( `Nivel de inglés` = nivelIngles ) + ( Total = 1 ) ~ ( Frecuencia = 1 ),
              data = df )

# html( tt, options = htmloptions( HTMLcaption = "Variable nivelIngles", pad = TRUE ) )
tablaLatex( tt, "Variable \\texttt{nivelIngles}." )
```

Tabla 5: Variable nivelIngles.

Nivel de inglés	Frecuencia
Nulo	297
CasiNulo	289
A1	451
A2	455
B1	473
B2	377
C1	145
C2	79
IsabelII	3
Shakespeare	2
Total	2571



### 3. Descriptivos con paquete `likert`

Se realizan gráficos descriptivos de los datos con el paquete `likert` (Bryer and Speersschneider (2015)) que, al estar basados en `ggplot2` (Wickham (2009)), permite modificarlos de forma sencilla.

```
# library( likert )
dfLikert <- df[ , grep( "^Q", colnames( df ) ) ]
colnames( dfLikert ) <- dicc[ grep( "^Q", dicc$item ), "spanish" ]

bloque1 <- 1:8
bloque2 <- 9:17
bloque3 <- 18:23

items1 <- likert( items = dfLikert[ , bloque1 ] )
items2 <- likert( items = dfLikert[ , bloque2 ] )
items3 <- likert( items = dfLikert[ , bloque3 ] )
```

#### 3.1. Gráficos de barras

```
plot( items1, centered = TRUE, group.order = colnames( items1$items ),
      legend.position = "right" ) +
  theme( axis.text.x = element_text( size = 10 ),
         axis.text.y = element_text( size = 13, hjust = 0 ),
         legend.text = element_text( size = 10 ),
         legend.title = element_text( size = 10 ) )
```

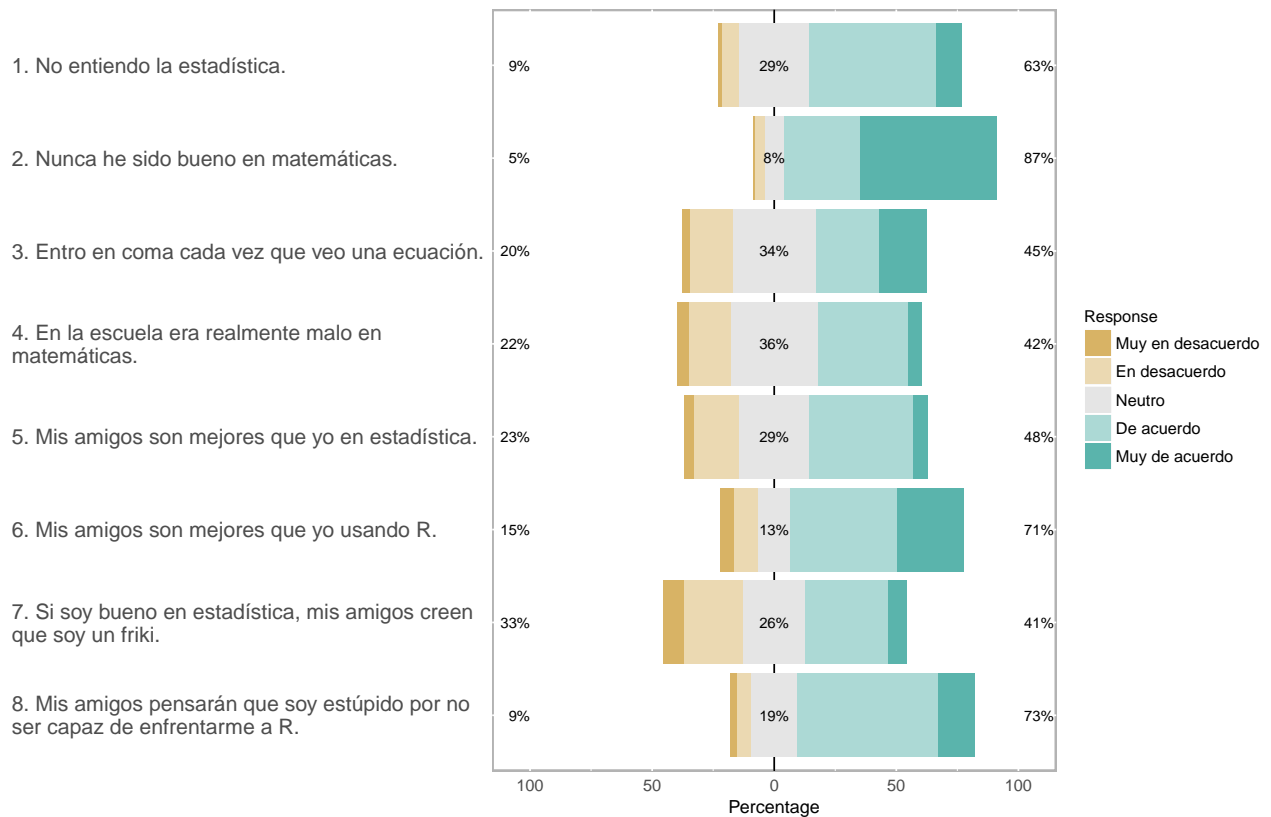


Figura 1: Gráfico de barras de los ítems 1 a 8.

```
plot( items2, centered = TRUE, group.order = colnames( items2$items ),
      legend.position = "right" ) +
  theme( axis.text.x = element_text( size = 10 ),
        axis.text.y = element_text( size = 13, hjust = 0 ),
        legend.text = element_text( size = 10 ),
        legend.title = element_text( size = 10 ) )
```

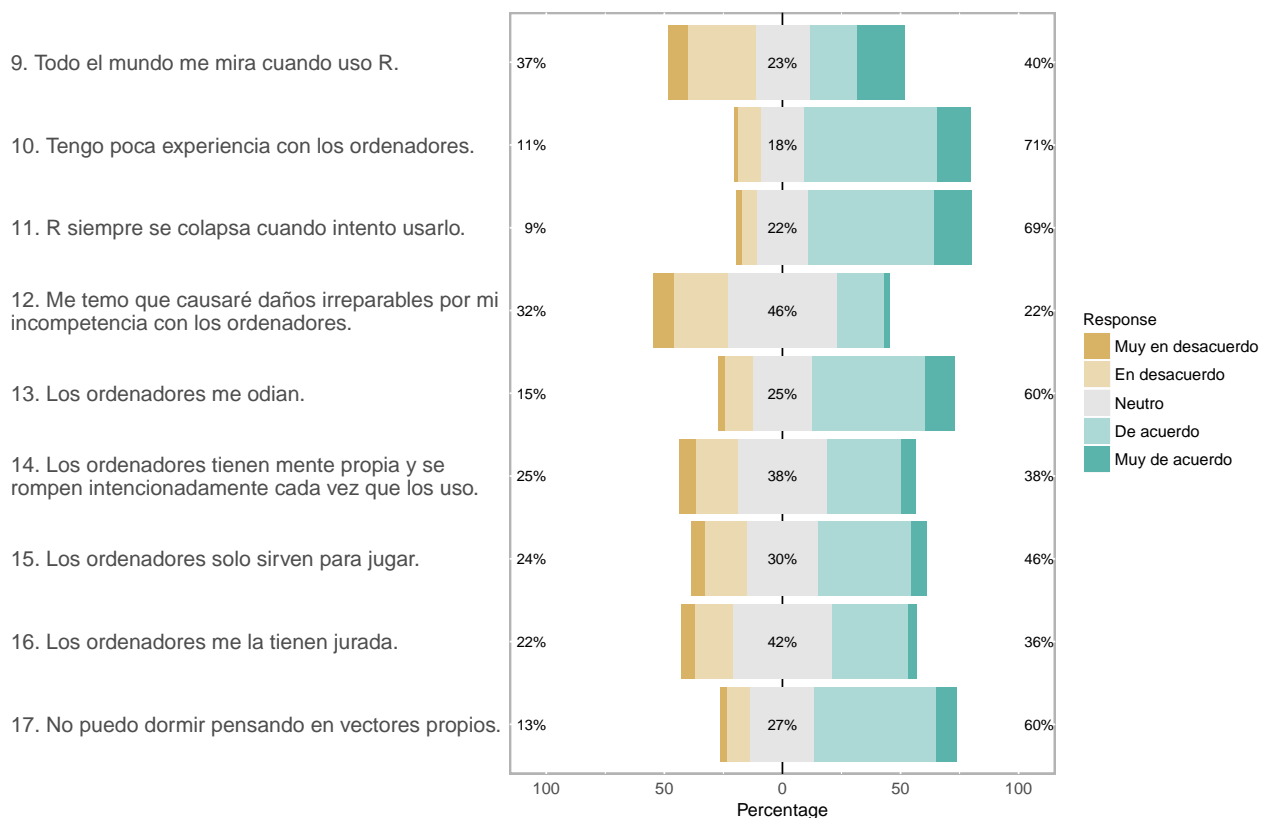


Figura 2: Gráfico de barras de los items 9 a 17.

```

plot( items3, centered = TRUE, group.order = colnames( items3$items ),
      legend.position = "right" ) +
  theme( axis.text.x = element_text( size = 10 ),
        axis.text.y = element_text( size = 13, hjust = 0 ),
        legend.text = element_text( size = 10 ),
        legend.title = element_text( size = 10 ) )
    
```

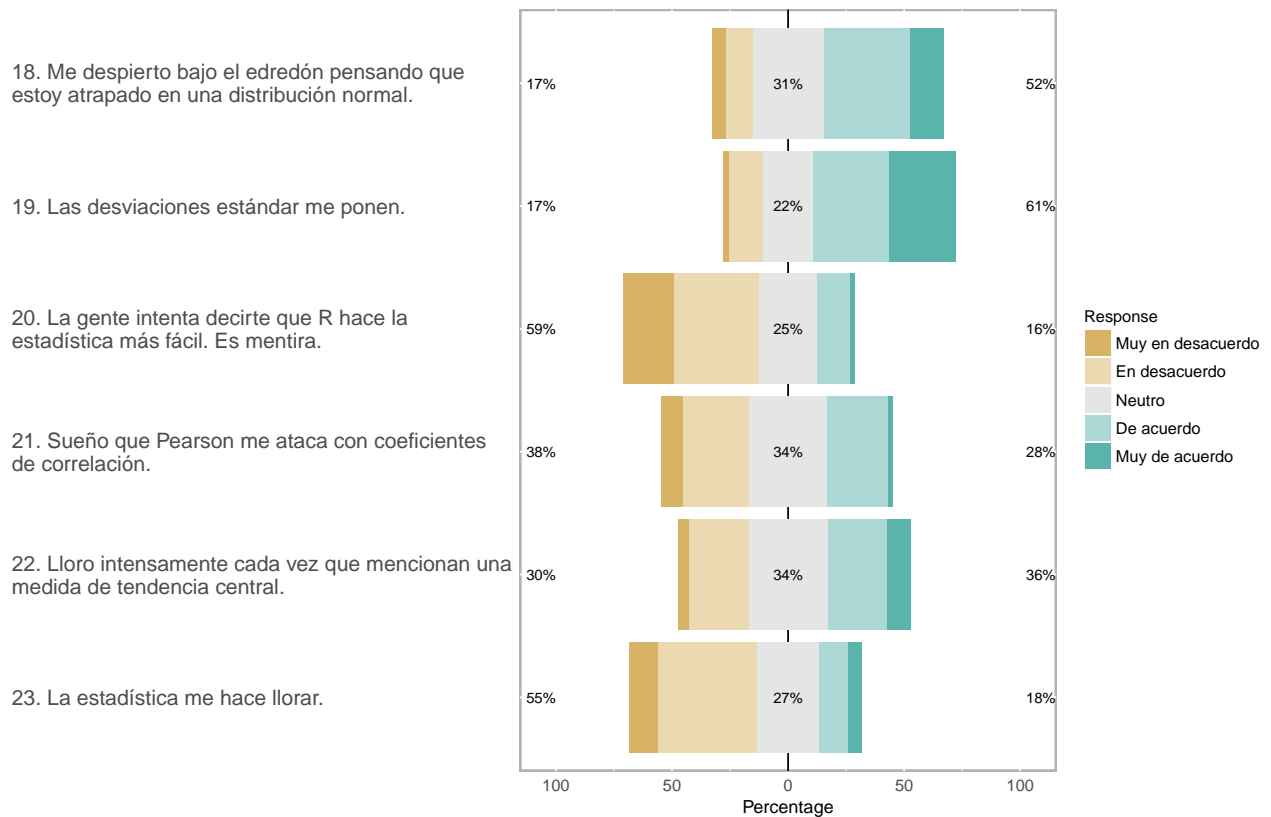


Figura 3: Gráfico de barras de los items 18 a 23.

### 3.2. Gráficos de densidad

```
plot( items1, type = "density" )
```

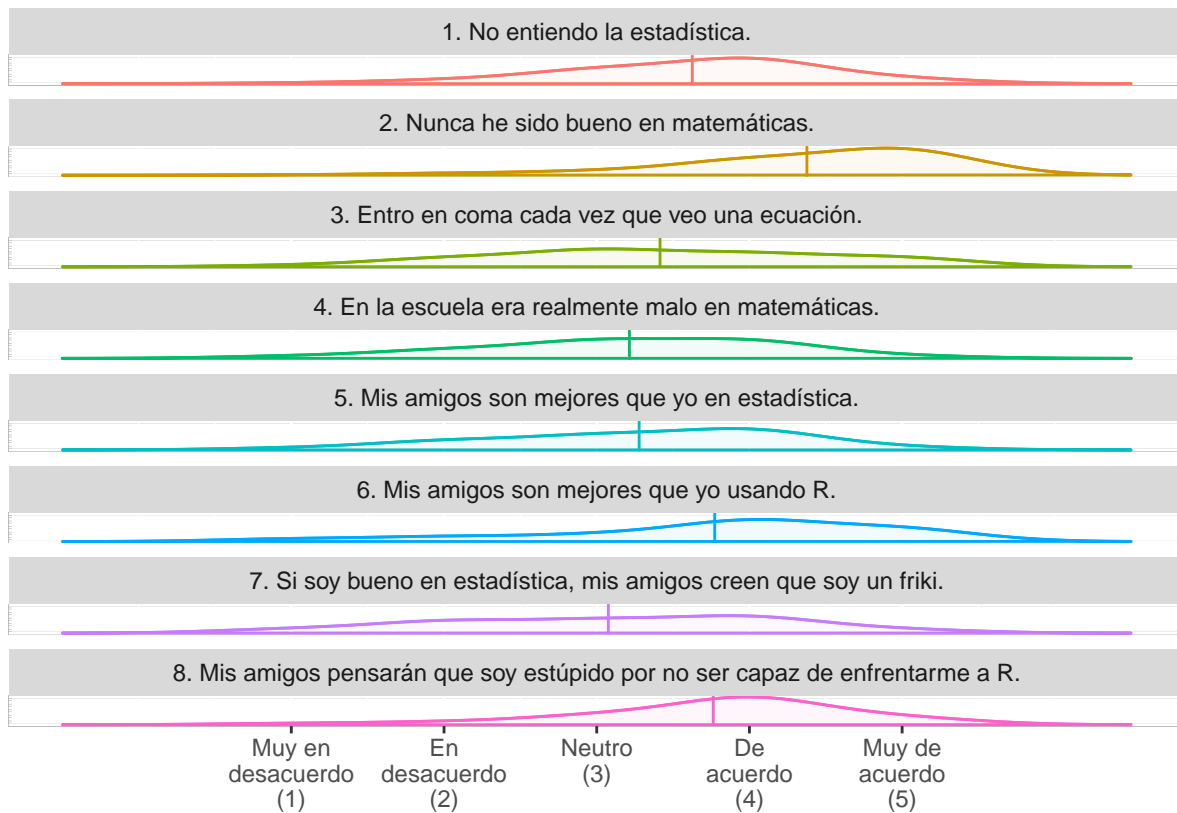


Figura 4: Gráfico de densidad de los ítems 1 a 8.

```
plot( items2, type = "density" )
```





Figura 5: Gráfico de densidad de los items 9 a 17.

```
plot( items3, type = "density" )
```

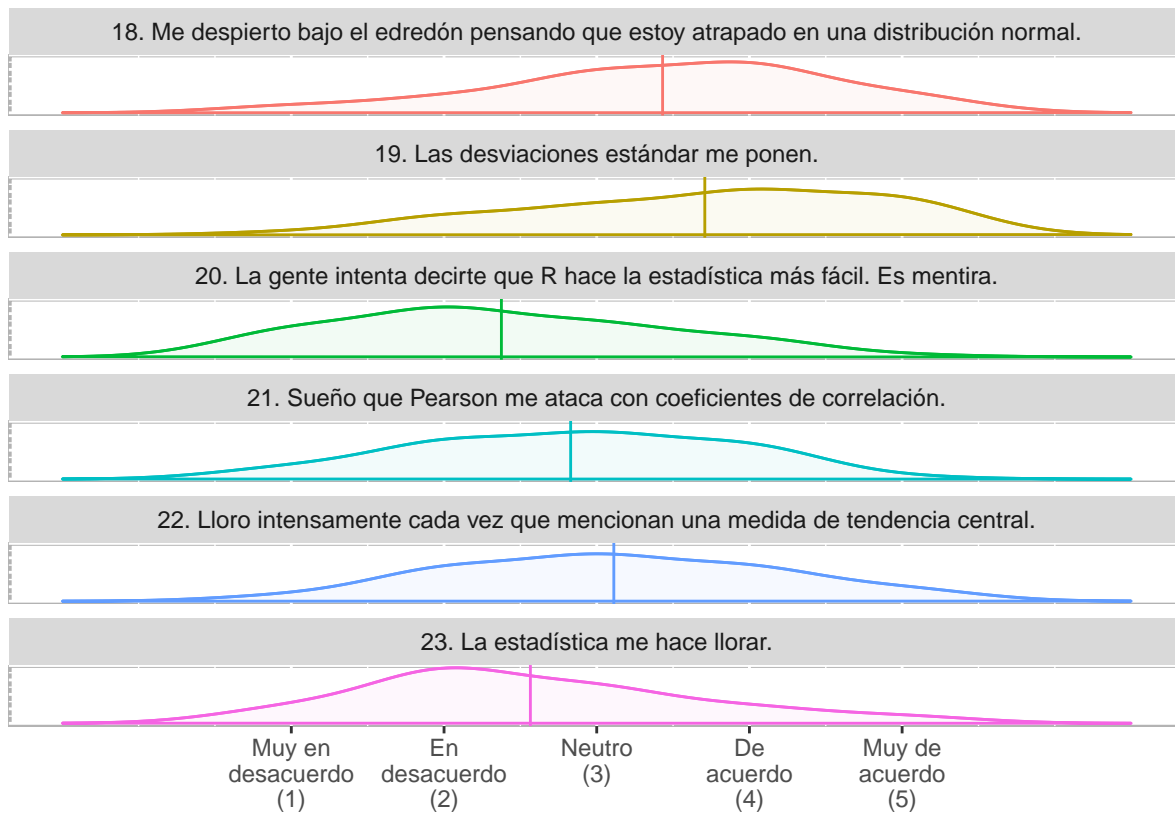


Figura 6: Gráfico de densidad de los items 18 a 23.

### 3.3. Gráficos de calor

```
plot( items1, type = "heat", group.order = colnames( items1$items ) ) +
  theme( axis.text.x = element_text( size = 8 ),
        axis.text.y = element_text( size = 13, hjust = 0 ),
        legend.text = element_text( size = 10 ),
        legend.title = element_text( size = 10 ) )
```

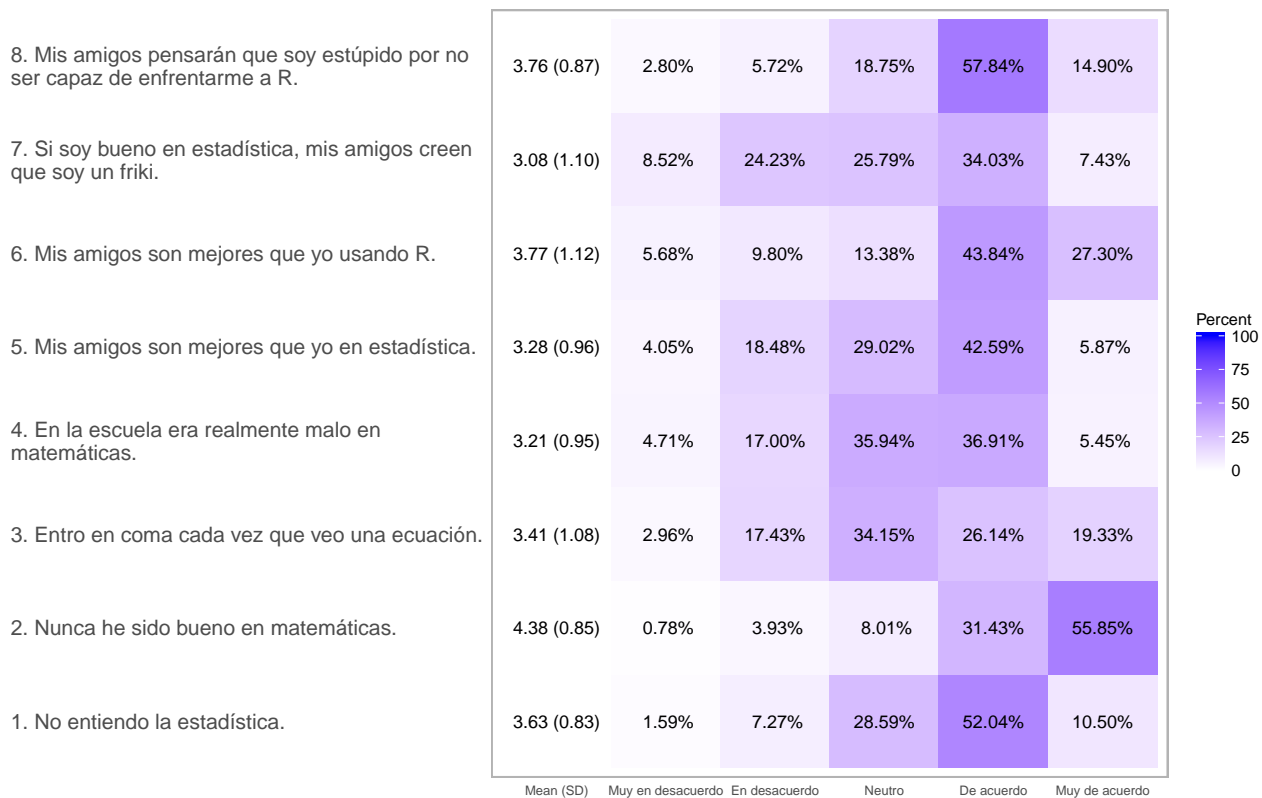


Figura 7: Gráfico de calor de los items 1 a 8.

```
plot( items2, type = "heat", group.order = colnames( items2$items ) ) +
  theme( axis.text.x = element_text( size = 8 ),
         axis.text.y = element_text( size = 13, hjust = 0 ),
         legend.text = element_text( size = 10 ),
         legend.title = element_text( size = 10 ) )
```

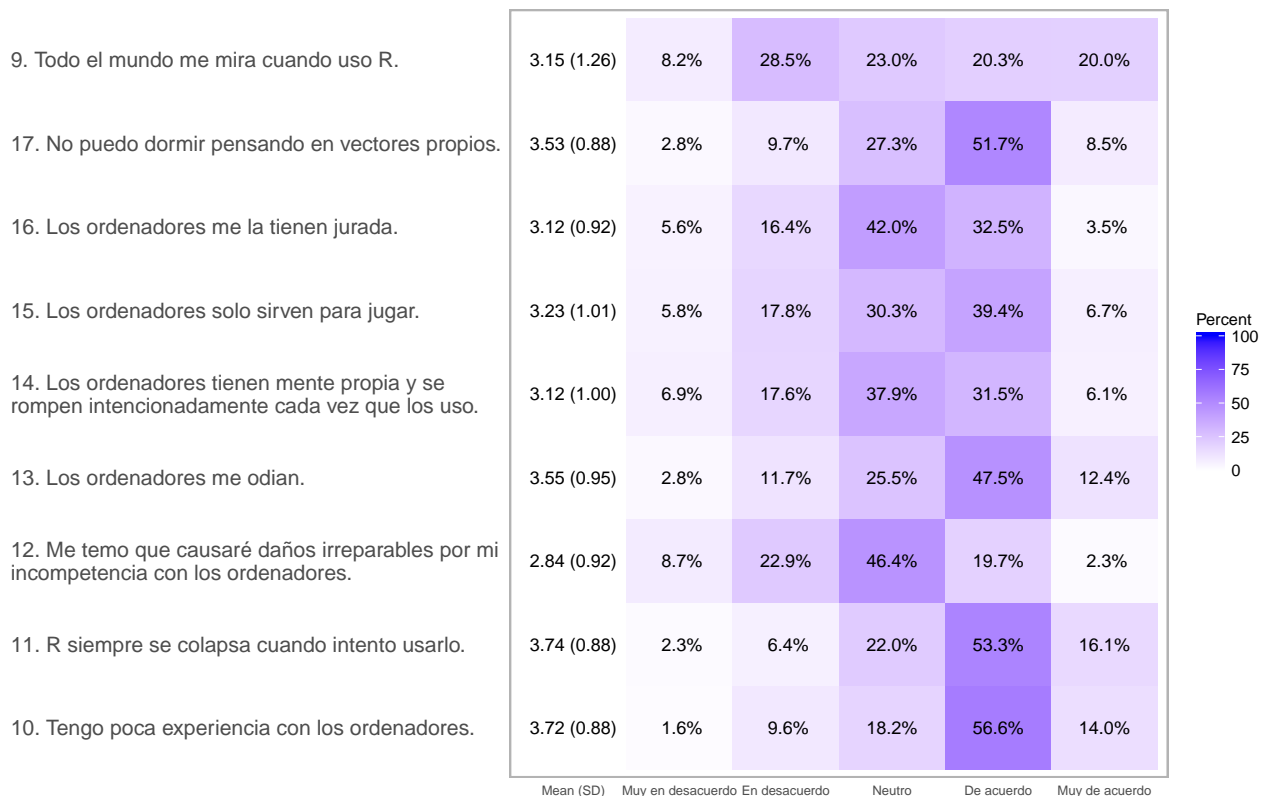


Figura 8: Gráfico de calor de los items 9 a 17.

```
plot( items3, type = "heat", group.order = colnames( items3$items ) ) +
  theme( axis.text.x = element_text( size = 8 ),
         axis.text.y = element_text( size = 13, hjust = 0 ),
         legend.text = element_text( size = 10 ),
         legend.title = element_text( size = 10 ) )
```

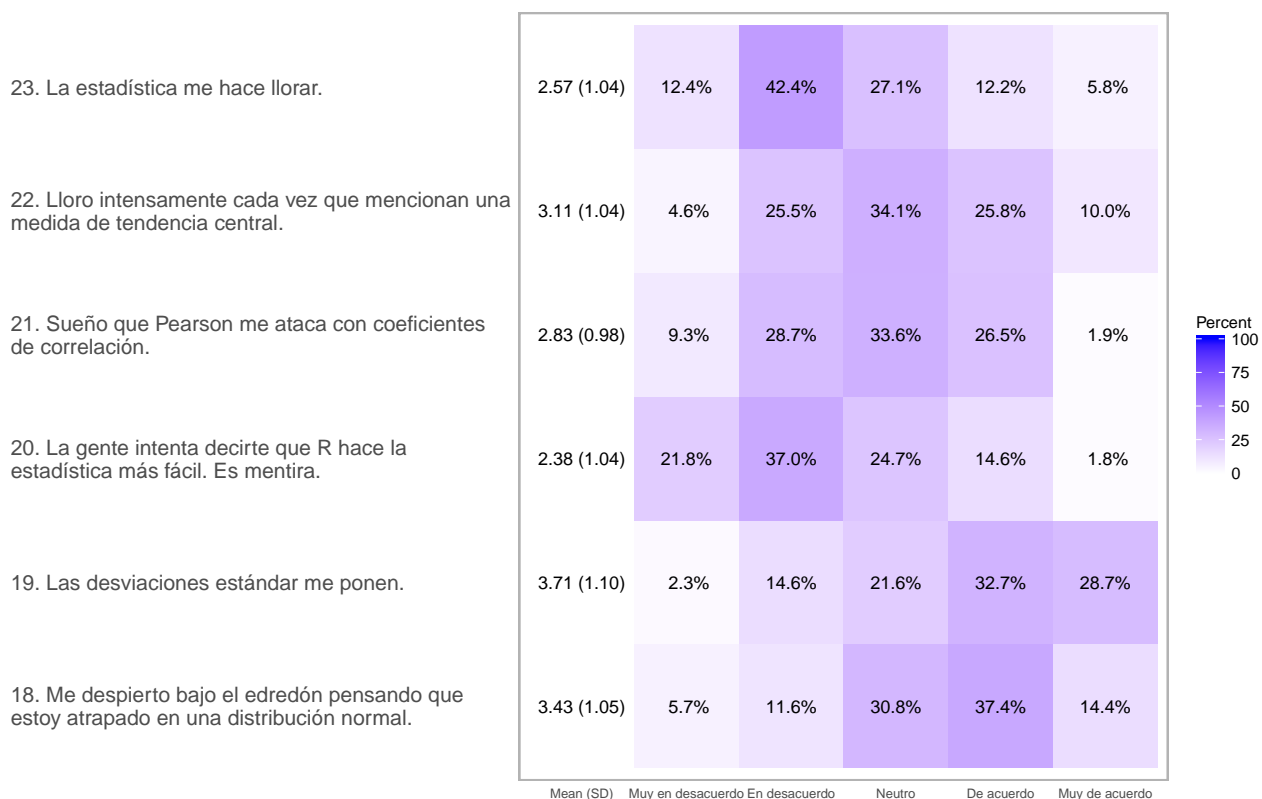


Figura 9: Gráfico de calor de los items 18 a 23.

## 4. Correlaciones

Se estudian correlaciones con los paquetes `corr` (Jackson (2016)) y `corrplot` (Wei and Simko (2016)).

```
# library( corr )
# library( corrplot )

dfCor <- df[ , grep( "^Q", colnames( df ) ) ]
```

### 4.1. Matriz de correlaciones

Se calcula la correlación con la función `cor` de la librería `stats`.

```
dfCor <- lapply( dfCor, as.numeric ) # devuelve una lista
dfCor <- as.data.frame( dfCor )     # coarcionamos a data.frame

corr <- cor( dfCor )
kable( corr[ 1:8, 1:8 ],
       caption = "Tabla de correlaciones (solo se muestran las ocho primeras)" )
```



Tabla 6: Tabla de correlaciones (solo se muestran las ocho primeras)

	Q01	Q02	Q03	Q04	Q05	Q06	Q07	Q08
Q01	1.0000000	-0.0987240	-0.3366489	0.4358602	0.4024399	0.2167340	0.3053651	0.3307376
Q02	-0.0987240	1.0000000	0.3183902	-0.1118597	-0.1193466	-0.0742097	-0.1591745	-0.0496226
Q03	-0.3366489	0.3183902	1.0000000	-0.3804602	-0.3103088	-0.2267405	-0.3819533	-0.2586342
Q04	0.4358602	-0.1118597	-0.3804602	1.0000000	0.4006722	0.2782015	0.4086150	0.3494294
Q05	0.4024399	-0.1193466	-0.3103088	0.4006722	1.0000000	0.2574601	0.3393918	0.2686270
Q06	0.2167340	-0.0742097	-0.2267405	0.2782015	0.2574601	1.0000000	0.5135805	0.2228318
Q07	0.3053651	-0.1591745	-0.3819533	0.4086150	0.3393918	0.5135805	1.0000000	0.2974970
Q08	0.3307376	-0.0496226	-0.2586342	0.3494294	0.2686270	0.2228318	0.2974970	1.0000000

Otra opción es utilizar las funciones `correlate` y `fashion` de librería `corr` (Jackson 2016).

```
corr2 <- correlate( dfCor )
kable( fashion( corr2[ 1:8, 1:9 ] ),
       caption = "Tabla de correlaciones (solo se muestran las ocho primeras)" )
```

Tabla 7: Tabla de correlaciones (solo se muestran las ocho primeras)

rowname	Q01	Q02	Q03	Q04	Q05	Q06	Q07	Q08
Q01		-.10	-.34	.44	.40	.22	.31	.33
Q02	-.10		.32	-.11	-.12	-.07	-.16	-.05
Q03	-.34	.32		-.38	-.31	-.23	-.38	-.26
Q04	.44	-.11	-.38		.40	.28	.41	.35
Q05	.40	-.12	-.31	.40		.26	.34	.27
Q06	.22	-.07	-.23	.28	.26		.51	.22
Q07	.31	-.16	-.38	.41	.34	.51		.30
Q08	.33	-.05	-.26	.35	.27	.22	.30	

## 4.2. Gráficos de correlaciones

```
network_plot( corr2, min_cor = 0.4 )
```

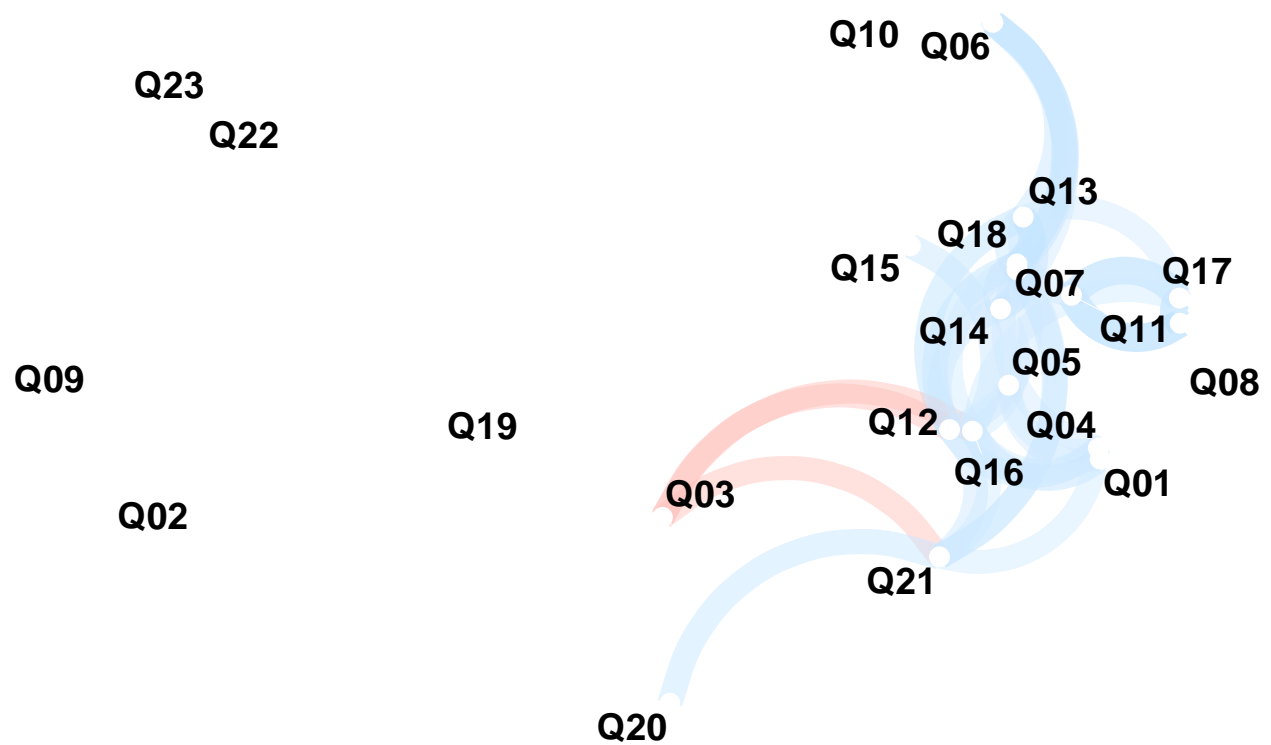


Figura 10: Red de correlaciones. Valor correlación mínima

```
corrplot.mixed( corr, tl.pos = "lt", diag = 'n', upper = "ellipse",
                number.cex = 0.4, tl.cex = 0.8,
                order = "hclust" )
```

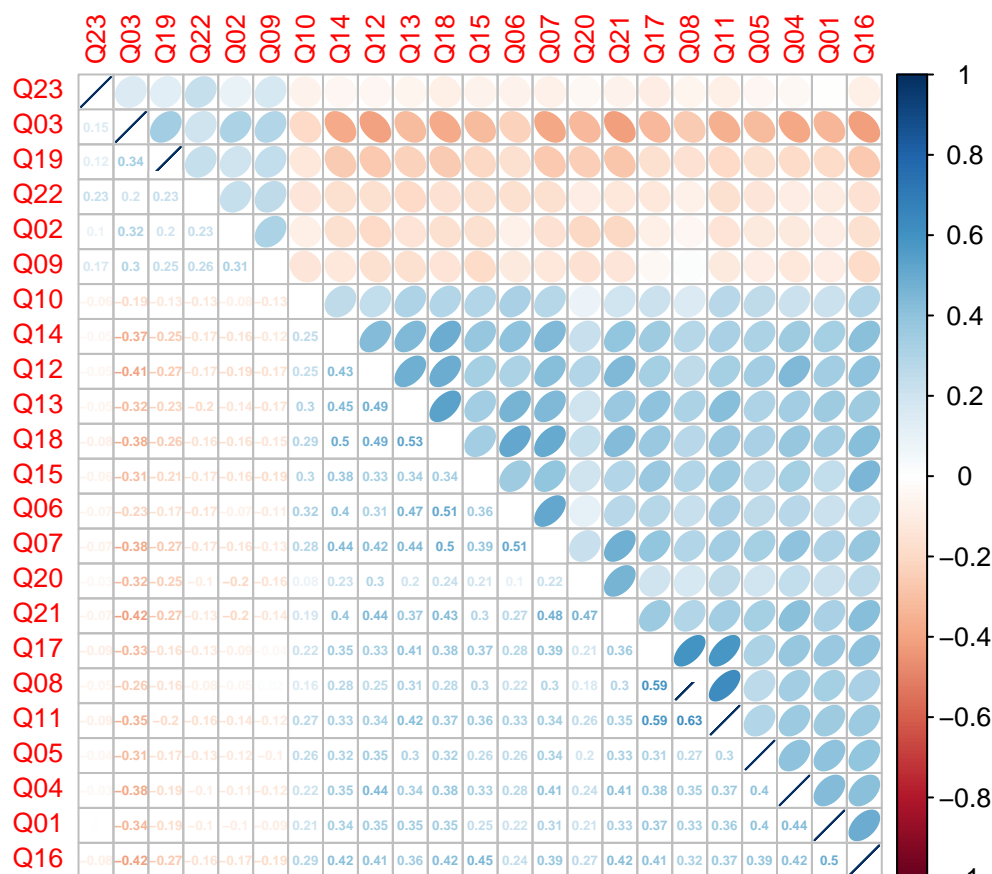


Figura 11: Matriz de correlaciones mixta

## Referencias y bibliografía

Bryer, Jason, and Kimberly Speerscheider. 2015. *Likert: Functions to Analyze and Visualize Likert Type Items*. <http://CRAN.R-project.org/package=likert>.

Field, Andy, Jeremy Miles, and Zoe Field. 2012. *Discovering Statistics Using R*. 1st edition. Sage Publications Ltd.

Jackson, Simon. 2016. *Corrr: Correlations in R*. <https://CRAN.R-project.org/package=corrr>.

Murdoch, Duncan. 2016. *Tables: Formula-Driven Table Generation*. <http://CRAN.R-project.org/package=tables>.

Wei, Taiyun, and Viliam Simko. 2016. *Corrplot: Visualization of a Correlation Matrix*. <http://CRAN.R-project.org/package=corrplot>.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. <http://ggplot2.org>.

Xie, Yihui. 2015. *Dynamic Documents with R and Knitr*. Vol. 29. CRC Press.