

Ventas y Marketing

Trabajo Práctico Final



**UNIVERSIDAD
TORCUATO DI TELLA**

Integrantes

González, Joaquín

Fecha de entrega: 27 de Septiembre de 2021

Introducción

En este trabajo se desarrolla e implementa un cuestionario con un diseño conjunto para determinar las preferencias de potenciales consumidores.

Diseño del Estudio

1) El producto a analizar es una notebook de gama media que tendrá distintas configuraciones de hardware, se espera obtener feedback respecto del análisis conjunto que nos permita tomar decisiones en base a las preferencias de nuestros clientes.

2) Atributos: Tamaño de la pantalla (asociado también al peso del equipo). Procesador. Memoria RAM (GB). Capacidad de almacenamiento (GB). Precio (USD)

3) Niveles de los atributos elegidos

Pantalla: 14", 15.6"

Procesador: Intel i3, Intel i5, Intel i7

Memoria RAM: 8GB, 16GB, 32GB

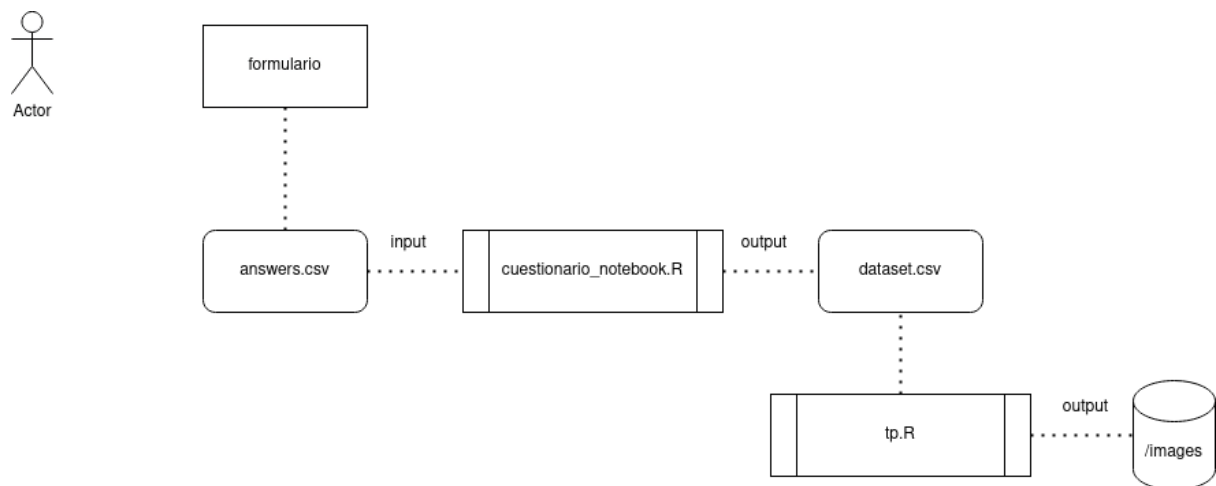
Almacenamiento (SSD): 250GB, 500GB, 1TB

Precios: 450USD, 700USD, 1300USD

4) Se genera un experimento ortogonal en base a los atributos de este trabajo utilizando el template de código R provisto por la cátedra. Luego, se genera un **Google Form** (<https://forms.gle/b2B7hAjPRW6o6pyK6>) para poder recolectar respuestas.

	Pantalla	Procesador	Memoria	Precio	Almacenamiento	Pantalla.1	Procesador.1	Memoria.1	Precio.1	Almacenamiento.1
6	15.6	i7	8GB	350USD	250GB	2	3	1	1	1
9	14	i5	16GB	350USD	250GB	1	2	2	1	1
23	14	i7	8GB	700USD	250GB	1	3	1	2	1
32	15.6	i3	32GB	700USD	250GB	2	1	3	2	1
44	15.6	i3	16GB	1300USD	250GB	2	1	2	3	1
67	14	i3	32GB	350USD	500GB	1	1	3	1	2
72	15.6	i7	32GB	350USD	500GB	2	3	3	1	2
73	14	i3	8GB	700USD	500GB	1	1	1	2	2
82	15.6	i5	16GB	700USD	500GB	2	2	2	2	2
94	15.6	i5	8GB	1300USD	500GB	2	2	1	3	2
101	14	i7	16GB	1300USD	500GB	1	3	2	3	2
110	15.6	i3	8GB	350USD	1TB	2	1	1	1	3
115	14	i3	16GB	350USD	1TB	1	1	2	1	3
138	15.6	i7	16GB	700USD	1TB	2	3	2	2	3
159	14	i5	32GB	1300USD	1TB	1	2	3	3	3

Se utiliza la librería de R **fastdummies** para generar un dataset con variables categóricas en base a los distintos niveles de cada atributo. Se genera un archivo que procesa los datos de la encuesta para generar el dataset que será input del script principal. Este flujo se muestra en el siguiente diagrama.



Formulario: <https://forms.gle/b2B7hAjPRW6o6pyK6>

Análisis de datos

1)

A continuación, se muestran las salidas de regresiones para los encuestados 15, 16 y 17. El resto de los resultados de las regresiones se pueden obtener ejecutando el script R adjuntado con el informe (tp.R).

Las mismas se obtienen realizando un loop sobre un dataset pre procesado que tiene una columna que permite identificar a qué usuario pertenecen las respuestas.

```
for (i in 1:n_encuestados){  
  tmp = lm(rankings ~ Pantalla.1_2 + Almacenamiento.1_2 + Almacenamiento.1_3 +  
  Procesador.1_2 + Procesador.1_3 + Memoria.1_2 + Memoria.1_3 + Precio.1_2 +  
  Precio.1_3,  
  data[data[, data$Encuestado == i]])  
  
  assign(paste("regresion", i , sep=""), tmp) # Guardo regresion en variable  
  print("-----")  
  print(sprintf("Regresion para encuestado %d", i))  
  print("-----")  
  print(summary(tmp))  
  ...  
}
```

El producto de referencia que determina los valores en el intercepto es:

- Pantalla 14", Almacenamiento 250GB, Procesador i3, Memoria RAM 8GB,
Precio 450usd

```
[1] "-----"
[1] "Regresion para encuestado 14"
[1] "-----"

Call:
lm(formula = rankings ~ Pantalla.1_2 + Almacenamiento.1_2 + Almacenamiento.1_3 +
  Procesador.1_2 + Procesador.1_3 + Memoria.1_2 + Memoria.1_3 +
  Precio.1_2 + Precio.1_3, data = data[data[, data$Encuestado ==
  i]])

Residuals:
    1     2     3     4     5     6     7     8     9    10    11    12
-0.45294  0.38868 -0.03106  0.25276 -0.15744 -0.30579  0.22385  0.06426 -0.31059  0.09274  0.23553  0.32700
    13    14    15
-0.18079  0.02462 -0.17082

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)      2.9826     0.3324   8.973 0.000287 ***
Pantalla.1_2     -0.2539     0.2257  -1.125 0.311712
Almacenamiento.1_2  4.6289     0.2669  17.344 1.17e-05 ***
Almacenamiento.1_3  5.9443     0.2945  20.185 5.52e-06 ***
Procesador.1_2      0.3748     0.2954   1.269 0.260372
Procesador.1_3     -0.2758     0.2660  -1.037 0.347382
Memoria.1_2        0.2539     0.2669   0.951 0.385098
Memoria.1_3       -0.3057     0.2945  -1.038 0.346810
Precio.1_2        -0.6758     0.2660  -2.540 0.051864 .
Precio.1_3       -1.8252     0.2954  -6.178 0.001619 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4298 on 5 degrees of freedom
Multiple R-squared:  0.9909,    Adjusted R-squared:  0.9745
F-statistic: 60.56 on 9 and 5 DF,  p-value: 0.0001434
```

```
[1] "-----"
[1] "Regresion para encuestado 15"
[1] "-----"

Call:
lm(formula = rankings ~ Pantalla.1_2 + Almacenamiento.1_2 + Almacenamiento.1_3 +
  Procesador.1_2 + Procesador.1_3 + Memoria.1_2 + Memoria.1_3 +
  Precio.1_2 + Precio.1_3, data = data[data[, data$Encuestado ==
  i]])

Residuals:
    1     2     3     4     5     6     7     8     9    10    11    12
 0.49412 -0.01265 -0.52612  0.14153 -0.09688 -0.32959  0.08171  0.51853 -0.26118  0.16747 -0.17694 -0.65400
    13    14    15
 0.42041  0.12724  0.10635

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)      3.91718     0.45019   8.701 0.000332 ***
Pantalla.1_2     -0.31576     0.30568  -1.033 0.348965
Almacenamiento.1_2 -0.14018     0.36148  -0.388 0.714132
Almacenamiento.1_3  0.05259     0.39887   0.132 0.900250
Procesador.1_2      4.48565     0.40012  11.211 9.86e-05 ***
Procesador.1_3      4.90447     0.36028  13.613 3.83e-05 ***
Memoria.1_2       -0.39018     0.36148  -1.079 0.329705
Memoria.1_3       -0.44741     0.39887  -1.122 0.312953
Precio.1_2       -0.29553     0.36028  -0.820 0.449375
Precio.1_3       -2.11435     0.40012  -5.284 0.003233 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5821 on 5 degrees of freedom
Multiple R-squared:  0.9797,    Adjusted R-squared:  0.9433
F-statistic: 26.86 on 9 and 5 DF,  p-value: 0.001035
```

```
[1] "-----"
[1] "Regresion para encuestado 16"
[1] "-----"

Call:
lm(formula = rankings ~ Pantalla.1_2 + Almacenamiento.1_2 + Almacenamiento.1_3 +
    Procesador.1_2 + Procesador.1_3 + Memoria.1_2 + Memoria.1_3 +
    Precio.1_2 + Precio.1_3, data = data[data[, data$Encuestado ==
    i]])

Residuals:
    1     2     3     4     5     6     7     8     9    10    11    12
-0.45882 -0.21147  0.60282 -0.07071  0.13818 -0.15288  0.40806 -0.32971  0.02824  0.36771 -0.32141 -0.18200
    13    14    15
 0.59712 -0.23065 -0.18447

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    7.01976    0.44529   15.765 1.87e-05 ***
Pantalla.1_2    0.45035    0.30235    1.489 0.196538
Almacenamiento.1_2 -0.07876    0.35754   -0.220 0.834353
Almacenamiento.1_3 -1.28812    0.39453   -3.265 0.022323 *
Procesador.1_2    0.52047    0.39576    1.315 0.245552
Procesador.1_3   -0.01129    0.35635   -0.032 0.975942
Memoria.1_2      0.67124    0.35754    1.877 0.119278
Memoria.1_3      1.21188    0.39453    3.072 0.027729 *
Precio.1_2      -2.61129    0.35635   -7.328 0.000742 ***
Precio.1_3      -6.27953    0.39576  -15.867 1.81e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5758 on 5 degrees of freedom
Multiple R-squared:  0.9829,    Adjusted R-squared:  0.9521
F-statistic: 31.93 on 9 and 5 DF,  p-value: 0.0006822
```

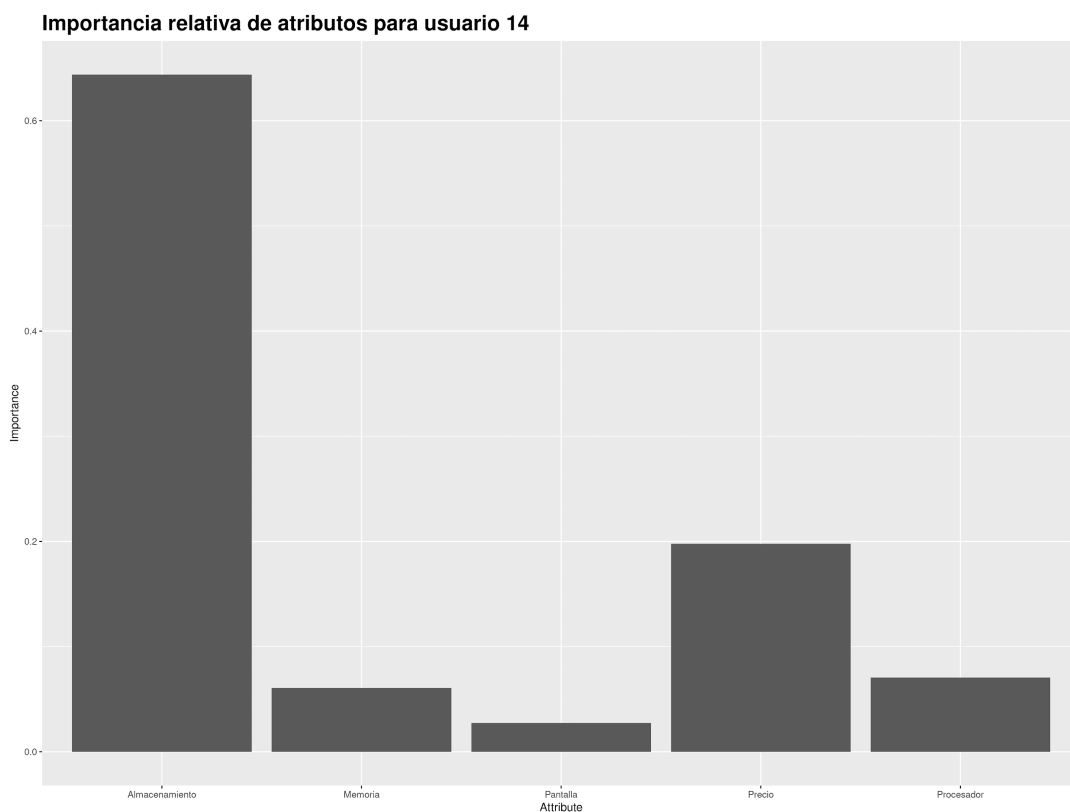
2) A continuación, se adjunta una tabla con la información de importancia relativa de cada atributo por encuestado.

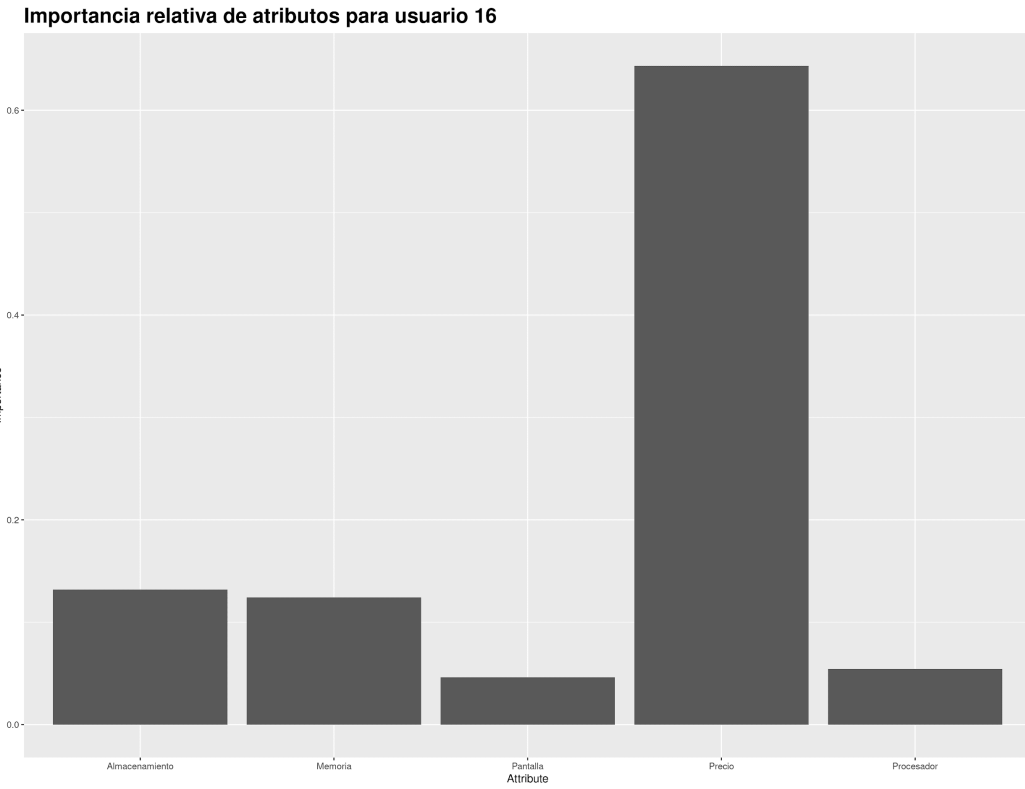
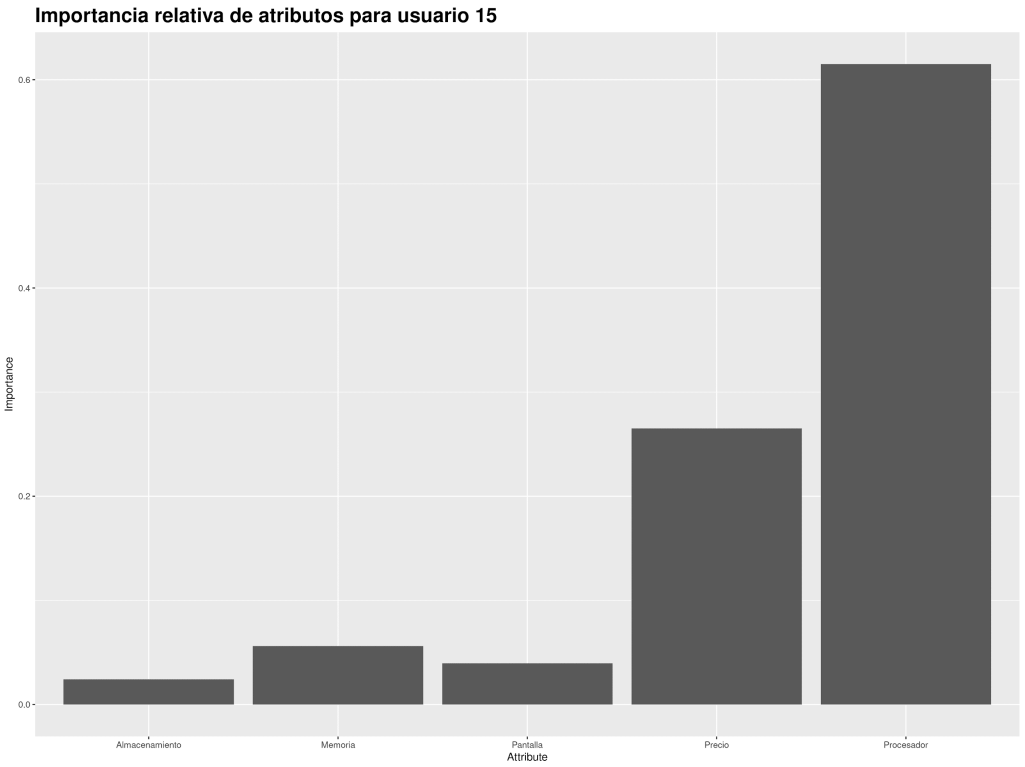
Usuario	Pantalla	Almacenamiento	Procesador	Memoria	Precio
1	0.058279	0.104899	0.229426	0.312748	0.294648
2	0.035070	0.367133	0.357213	0.023983	0.216602
3	0.045984	0.171720	0.345174	0.173185	0.263937
4	0.167331	0.195602	0.379405	0.130478	0.127184
5	0.013831	0.164830	0.230804	0.256378	0.334157
6	0.065698	0.253316	0.384081	0.074542	0.222363
7	0.074564	0.466538	0.157746	0.133047	0.168105
8	0.198133	0.084381	0.282729	0.167712	0.267044
9	0.025187	0.047886	0.150432	0.116052	0.660442
10	0.062620	0.150991	0.167627	0.237850	0.380912

11	0.035925	0.177921	0.096553	0.280354	0.409248
12	0.225198	0.152445	0.159159	0.249423	0.213775
13	0.000503	0.080079	0.082938	0.721875	0.114605
14	0.027496	0.643771	0.070459	0.060607	0.197668
15	0.039595	0.024172	0.614999	0.056103	0.265130
16	0.046135	0.131957	0.054475	0.124147	0.643286
17	0.017306	0.142999	0.486163	0.199368	0.154164

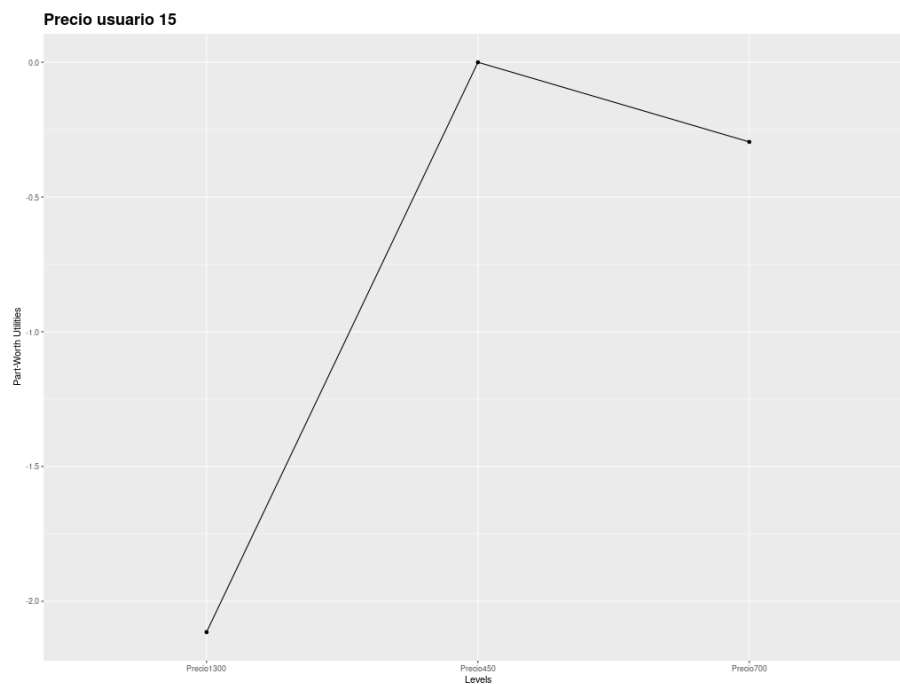
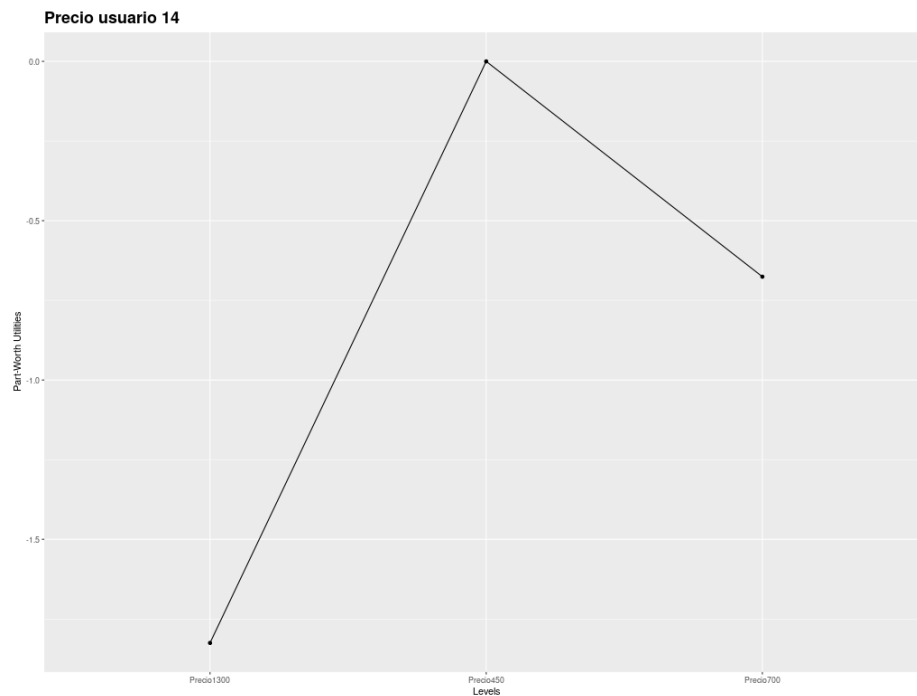
Verde: Atributo mas relevante, Amarillo: Segundo atributo mas relevante

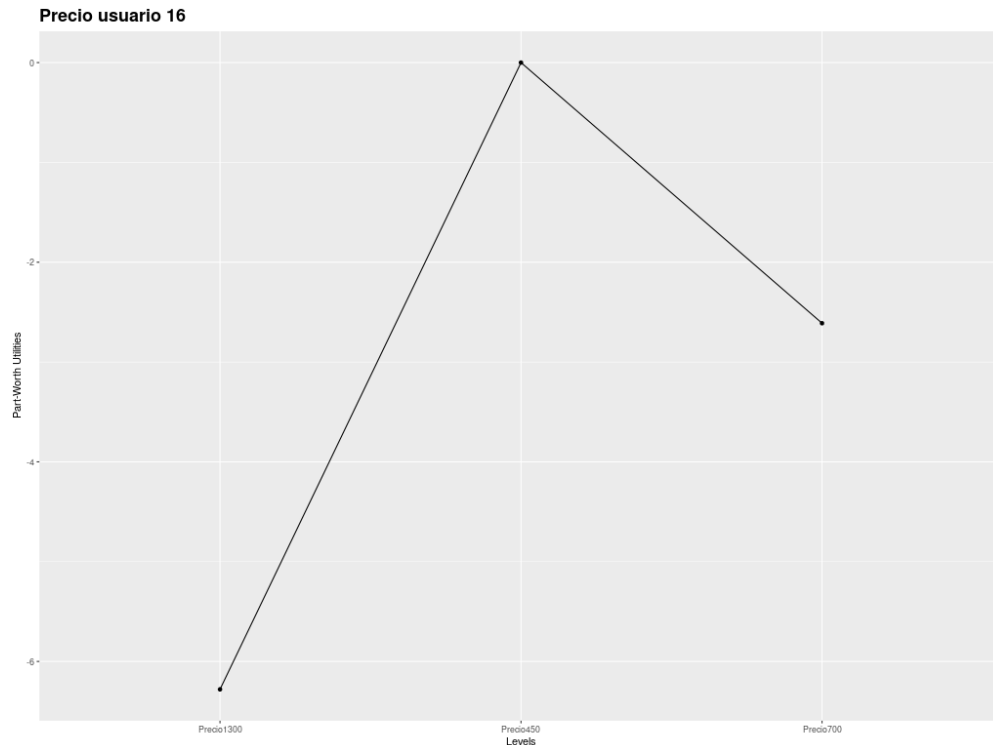
El script que se adjunta, además, exporta histogramas con la importancia de los atributos por usuario, se adjuntan los gráficos para los usuarios 14, 15 y 16 respectivamente.



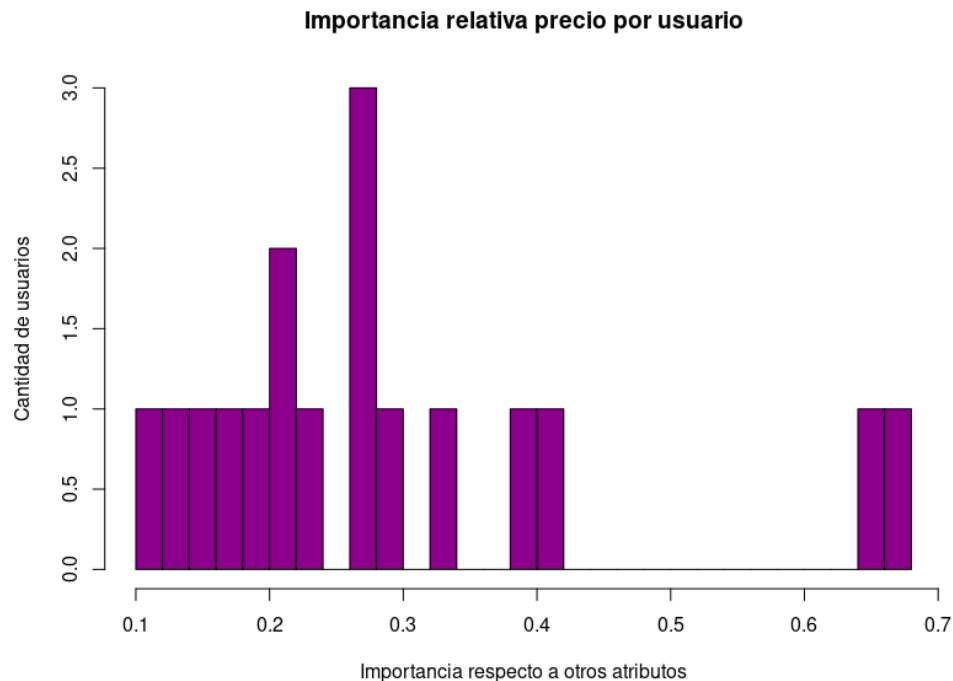


3) A continuación, se adjuntan gráficos de utilidad para el precio para los encuestados 14, 15 y 16 respectivamente. El resto de los gráficos donde se analiza la utilidad por variable se adjuntan en la carpeta **images** junto con este informe y pueden ser generadas nuevamente corriendo el script principal.





De analizar tanto los gráficos de los usuarios 14, 15 y 16 adjuntados en el informe como el de la totalidad de los encuestados, se puede observar una esperable sensibilidad a los precios altos, teniendo en estos la menor utilidad. A priori, se puede pensar que el precio no se está correlacionando con otras características de los productos ofrecidos como podría ser la calidad, que haga que un usuario tenga más utilidad al pagar más por el producto. Por otro lado, en términos de importancia relativa de los atributos (ver histograma a continuación), el precio es el más importante solo en 29.4% de los encuestados, por lo que hay una valoración importante respecto de los otros atributos elegidos para lanzar la encuesta del producto lo que nos indica que tenemos una buena selección de atributos para luego armar ofertas para lanzar en el mercado.



- 4) A continuación, se adjuntan los valores de voluntad a pagar por cambio en el atributo más importante por usuario. Este output se obtiene, también, al ejecutar el script principal **tp.R** que se adjunta con este informe.

El siguiente fragmento utilizado se encuentra dentro del bucle for principal y se realiza por usuario.

```
# Se quita el precio de relative_importance_user ya que no debemos realizar el
# analisis con ese atributo
relative_importance_user = c(cellPantalla_importance,
cellAlmacenamiento_importance, cellProcesador_importance,
cellMemoria_importance)
# Capturamos el atributo mas importante del current user del loop y el valor
most_important_attribute_value =
relative_importance_user[which.max(relative_importance_user)]
most_important_attrubute = attributes[which.max(relative_importance_user)]

cell <- subset(pw_ut_df_tmp, pw_ut_df_tmp$Variable ==
most_important_attrubute)
cell_max_ut <- max(cell$pw_ut_tmp)
cell_min_ut <- min(cell$pw_ut_tmp)

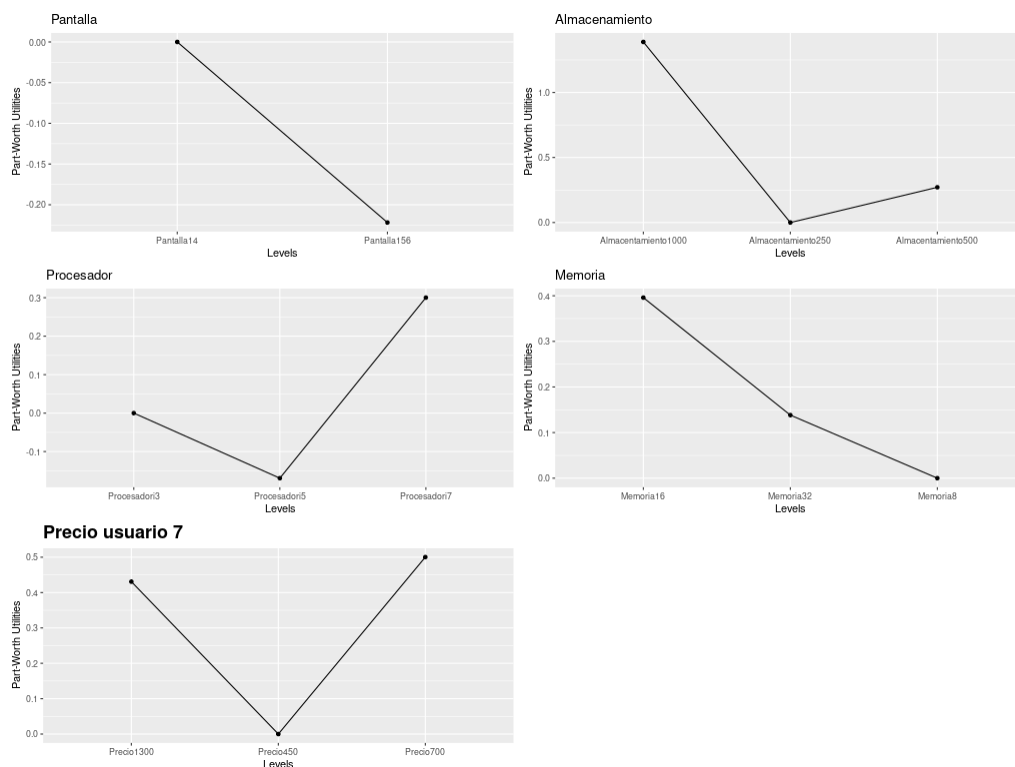
# Para comparar, siempre tomamos la diferencia entre el maximo y el minimo
# como rango para la categoria mas importante para este usuario
ut_range <- cell_max_ut - cell_min_ut
```

```
# Vemos el rango de precio entre alto y bajo y la diferencia de utilidad entre
esos precios para este usuario
price_range <- highPrice - lowPrice
price_ut_range <- cellPrecio$pw_ut_tmp[cellPrecio$Levels == "Precio450"] -
cellPrecio$pw_ut_tmp[cellPrecio$Levels == "Precio1300"]

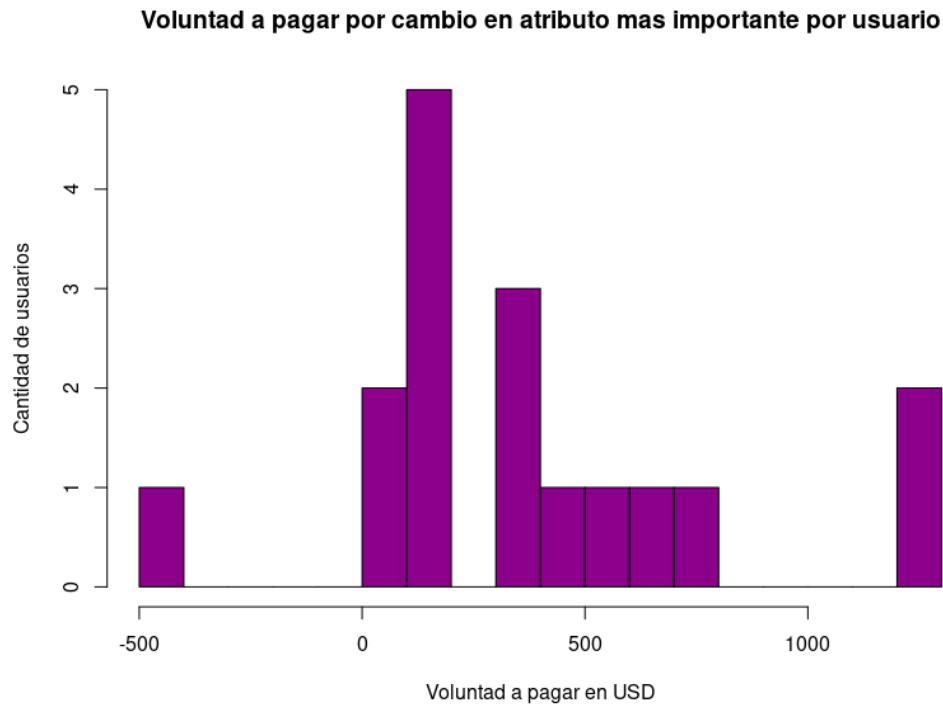
# The monetary value of one unit of utility is
mv <- price_range / price_ut_range
# Por lo tanto, el WTP para modificar el atributo favorito en base al rango de
precio es:
mv*ut_range
```

Usuario Encuestado	Voluntad de pago
1	302.612933
2	137.621816
3	148.088321
4	1231.034483
5	419.279734
6	1227.777778
7	-437.766248
8	630.655551
9	32.416670
10	336.935293
11	369.537774
12	710.478017
13	593.928180
14	118.235142
15	126.941910
16	60.959982
17	118.130990

El caso de voluntad de pago negativa del encuestado 7 lo asociamos a un outlier dentro de nuestro universo de usuarios encuestados. En ese caso puntual, el atributo más valorado por el usuario es el almacenamiento (mayor utilidad con 1TB), por lo que cambiar a otra opción baja drásticamente su voluntad a pagar ya que el precio no modifica tanto la utilidad como se puede ver en la siguiente figura.



Se adjunta un histograma para ver en qué valores se centran las voluntades de pago para esta encuesta.



- 5) De la tabla elaborada en el punto dos, se pueden identificar grupos de usuarios con preferencias marcadas en atributos específicos para **precio**, **procesador/memoria**, **procesador/almacenamiento** observando que columnas tuvieron alta densidad de atributos más importantes (marcados en verde y amarillo respectivamente). El atributo **pantalla** termina siendo poco valorado para ser considerado como criterio para elaborar un producto específico.

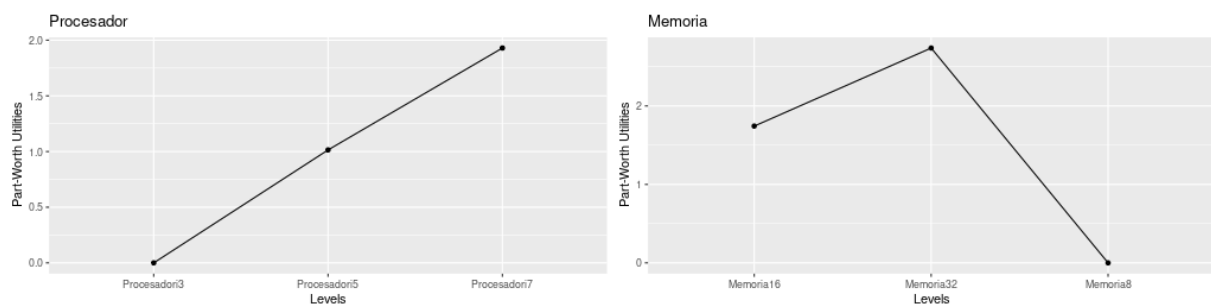
Para el grupo **precio** (usuarios 5, 9, 10, 11, 16) vemos que el precio está muy por encima de la valoración de cualquier otro atributo. Para este grupo de usuarios podríamos armar productos con configuraciones de hardware económicas con el

objetivo de ofrecer precios muy bajos sin tener en cuenta procesadores, memorias y almacenamiento de gran calidad/capacidad. Por ejemplo:

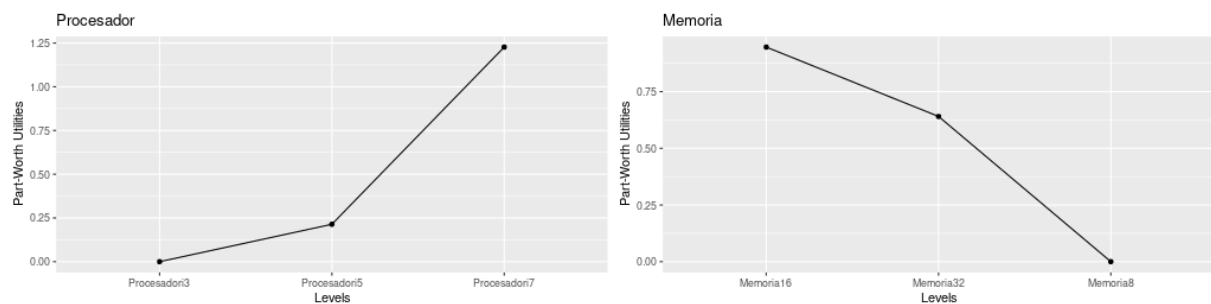
- Procesador i3, 8GB RAM, 250GB Almacenamiento, 14"/15.6", 450USD
precio.

Además, se observa que el precio es el atributo que más aparece como **segundo más relevante**.

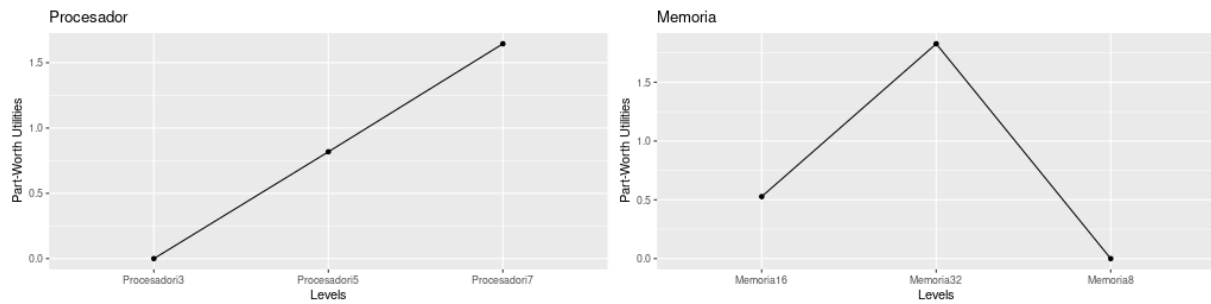
Para el grupo **procesador/memoria**, para los casos que tiene alta valoración en este atributo, se observa que los modelos preferidos son el i7 e i5 (solo un usuario presenta utilidad invertida en este atributo). En estos casos, se observa una gran cantidad de usuarios que valoran capacidad de memoria entre 16GB y 32GB



Usuario 10



Usuario 9



Usuario 5

Por lo que se pueden armar productos que contemplen estos dos modelos de procesador combinados con capacidades de memoria de 16GB y 32GB respectivamente utilizando el resto de los atributos de menor costo para poder optimizar el margen de ganancia.

- i7, 16GB, 250GB, 14"
- i7, 16GB, 250GB, 15.6"
- i5, 16GB, 250GB, 14"
- i5, 16GB, 250GB, 15.6"

Para el grupo **procesador/almacenamiento** se puede replicar la estrategia de **procesador/memoria** para elegir los atributos del producto.

- i7, 8GB, 1TB, 14"
- i7, 8GB, 1TB, 15.6"
- i5, 8GB, 1TB, 14"
- i5, 8GB, 1TB, 15.6"

6) Los productos elegidos que no se encuentran entre los diseñados para la encuesta son:

- i7, 32GB, 1TB, 14", 1300USD (A)
- i5, 16GB, 1TB, 14", 700USD (B)

En cada iteración del loop for principal, se calcula la utilidad de cada producto para cada usuario. El producto que tiene mejor utilidad será el elegido por el usuario.

Esto se guarda en el vector booleano **elijo_a**.

```
# La utilidad de un producto es la suma de las utilidades de sus atributos
para el usuario de esta iteracion
# Producto A: i7, 32GB RAM, 1TB, 14", 1300USD
# Producto B: i5, 16GB, RAM, 1TB, 14", 700USD

utility_a <- sum(pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Intercept"] +
pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Procesadori7"]
+ pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Memoria32"]
+ pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Almacenamiento1000"]
+ pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels ==
"Pantalla14"] + pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Precio1300"]))

utility_b <- sum(pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Intercept"] +
pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Procesadori5"]
+ pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Memoria16"]
+ pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Almacenamiento1000"]
+ pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels ==
"Pantalla14"] + pw_ut_df_tmp$pw_ut_tmp[pw_ut_df_tmp$Levels == "Precio700"]))

elijo_a = c(elijo_a, utility_a>utility_b)
```

Luego, se calcula el market share en base a cuantos usuarios eligieron cada producto con el siguiente fragmento.

```
usuarios_a = length(elijo_a[elijo_a == TRUE])
usuarios_b = length(elijo_a[elijo_a == FALSE])
usuarios = 17
market_share_a = usuarios_a / usuarios
market_share_b = usuarios_b / usuarios

print(sprintf("Market Share A: %f", market_share_a ))
print(sprintf("Market Share B: %f", market_share_b ))
```

Los resultados obtenidos son:

```
> print(sprintf("Market Share A: %f", market_share_a ))  
[1] "Market Share A: 0.470588"  
> print(sprintf("Market Share B: %f", market_share_b ))  
[1] "Market Share B: 0.529412"
```

Con esta información, sería una buena idea desarrollar el producto B.