



# 401 Caso Pareto II. Estudio de quejas y reclamaciones en restauración

## 4 Casos sectoriales. Analisis con Pareto.

Jose Ignacio González Gómez  
Departamento de Economía, Contabilidad y Finanzas - Universidad de La Laguna  
[www.jggomez.eu](http://www.jggomez.eu) **V.2.3**  
Ejercicio Basado en : ["SOLVED! Pareto Chart with TIES In Power BI | Step-By-Step Tutorial | Part 2"](#) - [Blog – ExcelFort](#)

Archivos fuentes: fichero ERP y Presupuesto  
Funciones DAX tratadas: ADDCOLUMNS, ALLSELECTED, ISBLANK, MAXX, RANKX, SELECTEDVALUE, SUMX  
Elementos visuales aplicados: Tabla, gráfico de columnas agrupadas y líneas., Tarjeta  
Otros. Creacion de parametros para establecer formatos condicionales en columnas.

## Contenido

1	Presentacion del caso.....	3
1.1	Información disponible .....	3
1.2	Objetivo .....	3
1.3	Justificación técnica de las medidas.....	4
1.3.1	Medida base .....	4
1.3.2	Sobre la medida Pareto % .....	5
1.3.3	Código base propuesto .....	5
2	Tabla de Pareto .....	6
2.1	Consideraciones .....	6
2.2	Codigo propuesto.....	6
2.2.1	Resultados parciales y resumen final.....	6
2.3	Tabla de Pareto y modelo final. ....	7
3	Gráfico de Pareto .....	8
3.1	Gráfico de columnas agrupadas y líneas.....	8
3.2	Creando un parámetro para el umbral de Pareto y medida de formato condicional ..	8
4	Perfeccionando el modelo. Clasificación Pareto ABC y análisis del resultado de la clasificación .....	10
4.1	Introducción .....	10
4.2	Parámetros de intervalos para el grupo A, B y C.....	10
4.2.1	Definición de los intervalos con parametros.....	10
4.2.2	Medidas de intervalos. Comprobacion de que los intervalos están correctamente definidos y advertencias.....	13
4.3	Creación de las medidas dinámicas de clasificación .....	14
4.3.1	Grupo de medidas “Medidas Clasifica Grupos”. Clasificación y Grupo .....	14
4.4	Análisis del resultado de la clasificación .....	16

---

4.4.1	Consideraciones .....	16
4.4.2	Medida de conversion. Grupo A1 B2 C3 .....	16
4.4.3	Medidas Grupo Total A, Total B y Total C .....	17
4.4.4	Medidas Grupo Total A%, Total B% y Total C%.....	18
4.4.5	Medidas para máximos y mínimos por categoría. ....	19
4.5	Tabla final del Análisis de la Clasificación.....	21

# 1 Presentacion del caso

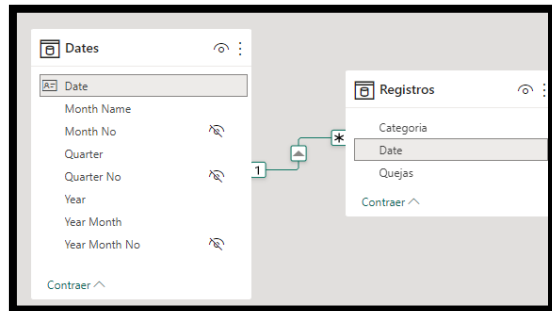
## 1.1 Información disponible

Disponemos de los registros de quejas de los clientes de nuestra cadena de cafeterías que presentamos en el fichero Excel Reclamaciones.xlsx y en concreto la única tabla de datos que contiene llamada "Registro" una tabla de hechos con los siguientes datos y estructura para el periodo 2023-2024.

	A	B	C	D
1		Date	Quejas	Categoría
2		04/01/2023	Confusión de pedidos	Servicio
3		04/01/2023	Confusión de pedidos	Servicio
4		04/01/2023	Confusión de pedidos	Servicio
5		04/01/2023	Confusión de pedidos	Servicio
6		01/06/2024	Confusión de pedidos	Servicio
7		07/06/2023	Confusión de pedidos	Servicio
8		12/06/2023	Confusión de pedidos	Servicio

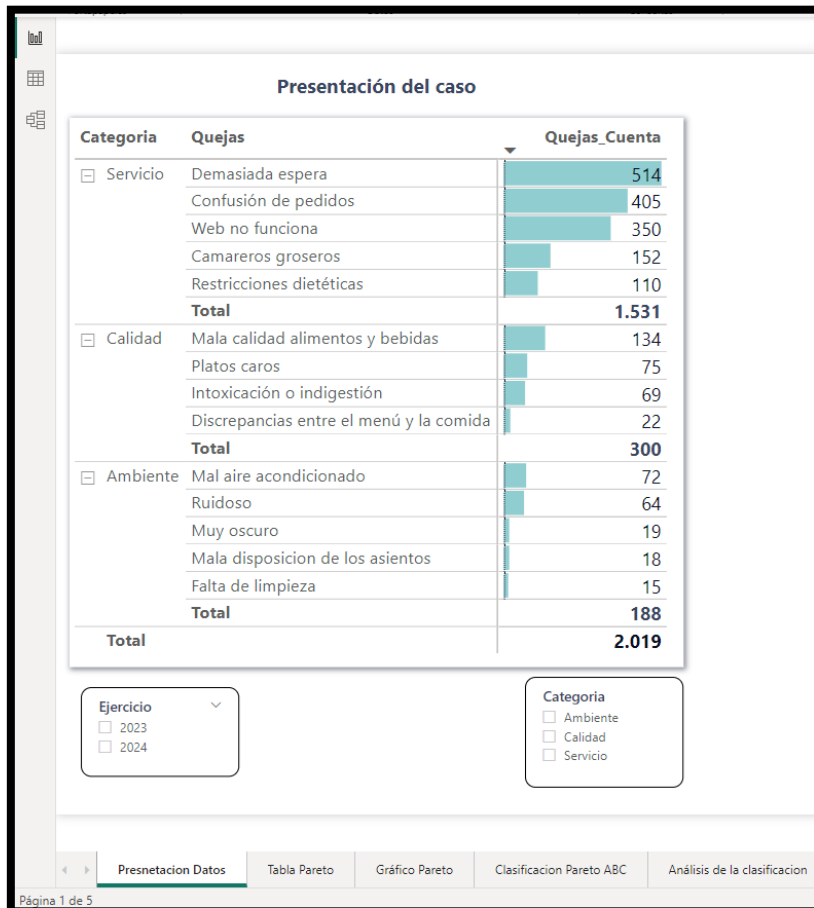
Esta tabla sencilla queremos analizar aplicando la técnica de Pareto al tipo de reclamaciones presentadas por nuestros clientes cuantificando cada tipo

Al disponer de un campo fecha deberemos crear una tabla calendario para relacionarla creando nuestro modelo de datos de forma sencilla tal y como mostramos a continuación.



## 1.2 Objetivo

Partimos de una primera visualización con una tabla resumen según el tipo de quejas y categoría, disponiendo de los segmentadores correspondientes.



El objetivo de este caso es aplicar la técnica de Pareto para el analisis de la relevancia de las quejas presentadas por nuestros clientes en formato tabla resumen como la siguiente y grafico

Quejas - motivo	Nº de Quejas	Pareto	Pareto %
Demasiada espera	514	514	25,46 %
Confusión de pedidos	405	919	45,52 %
Web no funciona	350	1269	62,85 %
Camareros groseros	152	1421	70,38 %
Mala calidad alimentos y bebidas	134	1555	77,02 %
Restricciones dietéticas	110	1665	82,47 %
Platos caros	75	1740	86,18 %
Mal aire acondicionado	72	1812	89,75 %
Intoxicación o indigestión	69	1881	93,16 %
Ruidoso	64	1945	96,33 %
Discrepancias entre el menú y la comida	22	1967	97,42 %
Muy oscuro	19	1986	98,37 %
Mala disposicion de los asientos	18	2004	99,26 %
Falta de limpieza	15	2019	100,00 %
<b>Total</b>	<b>2.019</b>		

**Categoría**

Ambiente

Calidad

Servicio

**Ejercicio**

2023

2024



### 1.3 Justificación técnica de las medidas

#### 1.3.1 Medida base

En nuestro caso y atendiendo al objetivo planteado de analisis del ABC, conocer el número de quejas de nuestros clientes según tipo, y para evaluar esta variable cualitativa, necesitamos una medida que nos cuente el número de sucesos que se obtiene y para ello creamos la siguiente medida:

$$\text{Quejas\_Cuenta} = \text{COUNTROWS}(\text{Registros})$$

En otros casos, por ejemplo, cuando la variable es cuantitativa como puede ser el importe de ventas u otra, la medida base estará fundamentada frecuentemente en una suma.

### 1.3.2 Sobre la medida Pareto %

La segunda medida que vamos a crear es la principal que denominaremos Pareto% y que servirá de base para otras medidas derivadas de esta, por lo cual es considerada como base.

Esta tendrá como objetivo disponer de una tabla temporal con los registros seleccionados según contexto de filtro en la que se calcule los acumulados de la variable objeto de análisis, por ejemplo suma de ventas, número de partes, número de visitas, etc.. En este sentido, recordar que con la técnica de Pareto se pretende ir acumulando para cada registro o línea la columna la variable a evaluar.

Es decir, la propuesta es diseñar una medida que genere una tabla virtual según el contexto de filtro, deberá contar con una columna que nos permita ir de acumulado, ir sumando acumulativamente y por orden de descendente, es decir de mayor a menor, para construir esta columna de acumulado

Esto implica la necesidad de contar con un "Rankin" de valores correspondientes a la variable objeto de trabajo, por ejemplo, de volumen de ventas, número de fallos donde deberíamos evitar la existencia, pero este ranking debe estar truncado para evitar que tengan con valores duplicados que condicionen el establecer un ranking y para ello se propone un ajuste que pasaremos a justificar

### 1.3.3 Código base propuesto

Presentamos, a modo ilustrativo, el código de la medida fundamental Pareto%, con el objeto de llamar la atención de lo simple del mismo y que pasamos a explicar en el siguiente apartado.

```

1 Pareto % =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4   VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5   VAR _TablaVirtualTrabajo =
6     ADDCOLUMNS (
7       _SeleccionColumnaTrabajo,
8       "@NumDeQuejas", [Quejas_Cuenta],
9       "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10      | | | RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11    )
12 VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo) , [@Rank] )
13 VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [@NumDeQuejas] )
14 VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15 VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado)
16 RETURN
17 _ParetoPorcentaje
18 )

```

## 2 Tabla de Pareto

### 2.1 Consideraciones

Como hemos comentado anteriormente esta medida fundamental tiene como objetivo disponer de una tabla temporal o virtual con los registros seleccionados derivados del contexto de filtro y permita determinar los acumulados debiendo actualizarse automáticamente cada vez que se produce un cambio en el contexto de filtro. Por tanto, el eje central está en lograr los acumulados a nivel de registro o fila, lo que implicar el uso de funciones de iteración como SUMX, RANGX, etc...

Señalar que la clasificación de Pareto en este caso se basa A<= 80% - B>80% y C> 90%

### 2.2 Código propuesto

```

1 Pareto % =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4   VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5   VAR _TablaVirtualTrabajo =
6     ADDCOLUMNS (
7       _SeleccionColumnaTrabajo,
8       "@NumDeQuejas", [Quejas_Cuenta],
9       "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10      RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11     )
12   VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo) , [@Rank] )
13   VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [@NumDeQuejas] )
14   VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15   VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado)
16   VAR _Clasificacion = SWITCH(TRUE(),
17     _ValorAcumulado>0.9,"C",
18     _ValorAcumulado>0.8,"B",
19     _ValorAcumulado<=0.8,"A")
20   RETURN
21   _ParetoPorcentaje
22 )

```

#### 2.2.1 Resultados parciales y resumen final

De esta medida base creada (Pareto%) cuyo retorno final nos devuelve el resultado de la variable ParetoPorCiento (ver línea 17 del código), también podríamos modificarla y crear una nueva medida que llamamos Pareto y que devuelva el total en la línea 17 del código el valor acumulado.

Igualmente, para el Total del valor acumulado.

Así los resultados parciales obtenidos del diseño de las principales variables y medidas los presentamos en la siguiente tabla resumen.

```

16 RETURN
17 | _ValorAcumulado
18 )

```

```

16 RETURN
17 | _TotalValorAcumulado
18 )

```

**Esta sería la medida      Estas son algunas de las variables finales**

Quejas	Quejas_Cuenta	_ValorAcumulado	_TotalValorAcumulado	Pareto %
Demasiada espera	514	514	2019	25,46 %
Confusión de pedidos	405	919	2019	45,52 %
Web no funciona	350	1269	2019	62,85 %
Camareros groseros	152	1421	2019	70,38 %
Mala calidad alimentos y bebidas	134	1555	2019	77,02 %
Restricciones dietéticas	110	1665	2019	82,47 %
Platos caros	75	1740	2019	86,18 %
Mal aire acondicionado	72	1812	2019	89,75 %
Intoxicación o indigestión	69	1881	2019	93,16 %
Ruidoso	64	1945	2019	96,33 %
Discrepancias entre el menú y la comida	22	1967	2019	97,42 %
Muy oscuro	19	1986	2019	98,37 %
Mala disposición de los asientos	18	2004	2019	99,26 %
Falta de limpieza	15	2019	2019	100,00 %
<b>Total</b>	<b>2.019</b>			

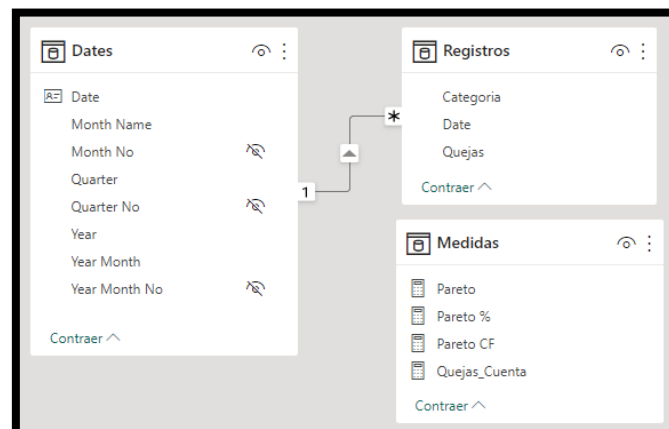
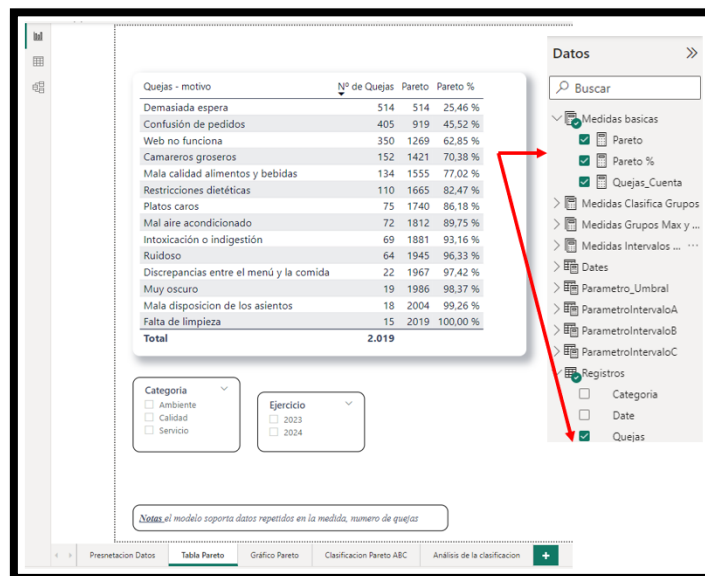
Quejas\_Cuenta = COUNTROWS(Registros)

VAR \_ValorAcumulado = SUMX( FILTER( \_TablaVirtualTrabajo , [@Rank] <= \_MaxRankin ) , [@NumDeQuejas] )

VAR \_TotalValorAcumulado = CALCULATE( [Quejas\_Cuenta] , \_SeleccionColumnaTrabajo )

### 2.3 Tabla de Pareto y modelo final.

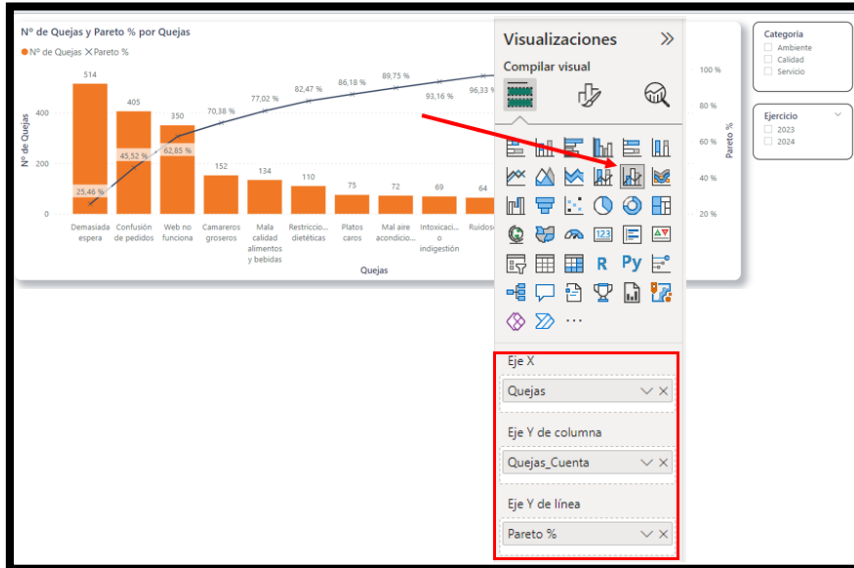
De esta forma y dando formato obtenemos el resultado final con las medidas empleadas y con los filtros o segmentadores básicos.



### 3 Gráfico de Pareto

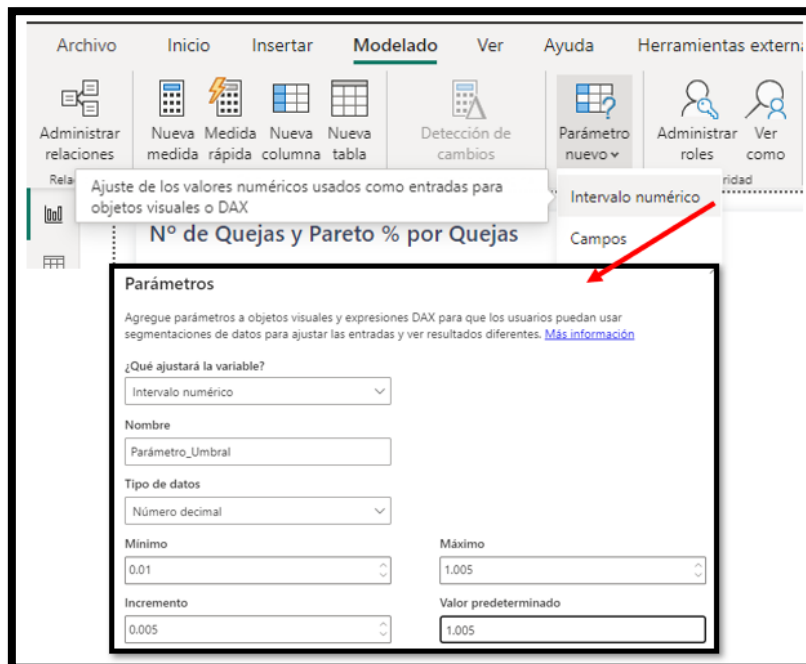
#### 3.1 Gráfico de columnas agrupadas y líneas

Basado en los cálculos y medidas basicas anteriores creamos un gráfico de columnas agrupadas y líneas con el siguiente diseño, donde la configuración basica es la siguiente.



#### 3.2 Creando un parámetro para el umbral de Pareto y medida de formato condicional

A continuación, vamos a crear un parámetro que llamaremos Umbral de Pareto y que tendrá como escala el intervalo de 0 al 100% y que nos permitirá realizar una analisis de selección de aquellos registros que superan un valor del umbral seleccionado.



Una vez creada el parámetro crearemos una medida que nos permita dar un formato condicional a nuestro gráfico, a esta la llamaremos Pareto Formato (Pareto formato condicional) y que tendrá esta sintaxis básica.

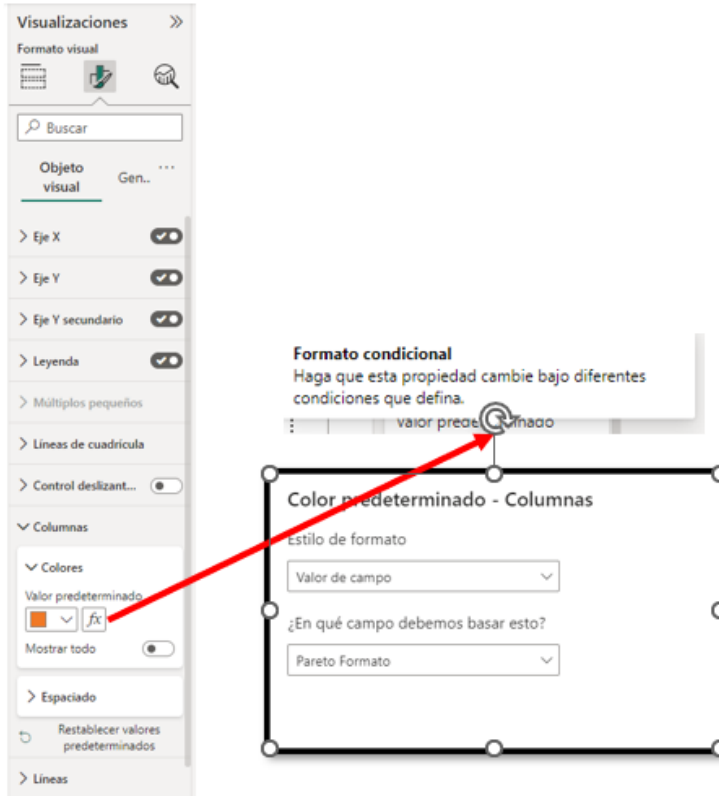


Pareto Formato =

IF( [Pareto %] <= [Valor Parametro\_Umbral] , "red" )

Lo que se pretende con esta medida es Pareto% es menor igual al valor del parámetro umbral se le asigna un color, en nuestro caso rojo.

Ahora solo nos queda aplicarlo para lo cual volvemos al gráfico y en las propiedades de la columna identificamos la nueva propiedad



Obteniendo así el resultado final

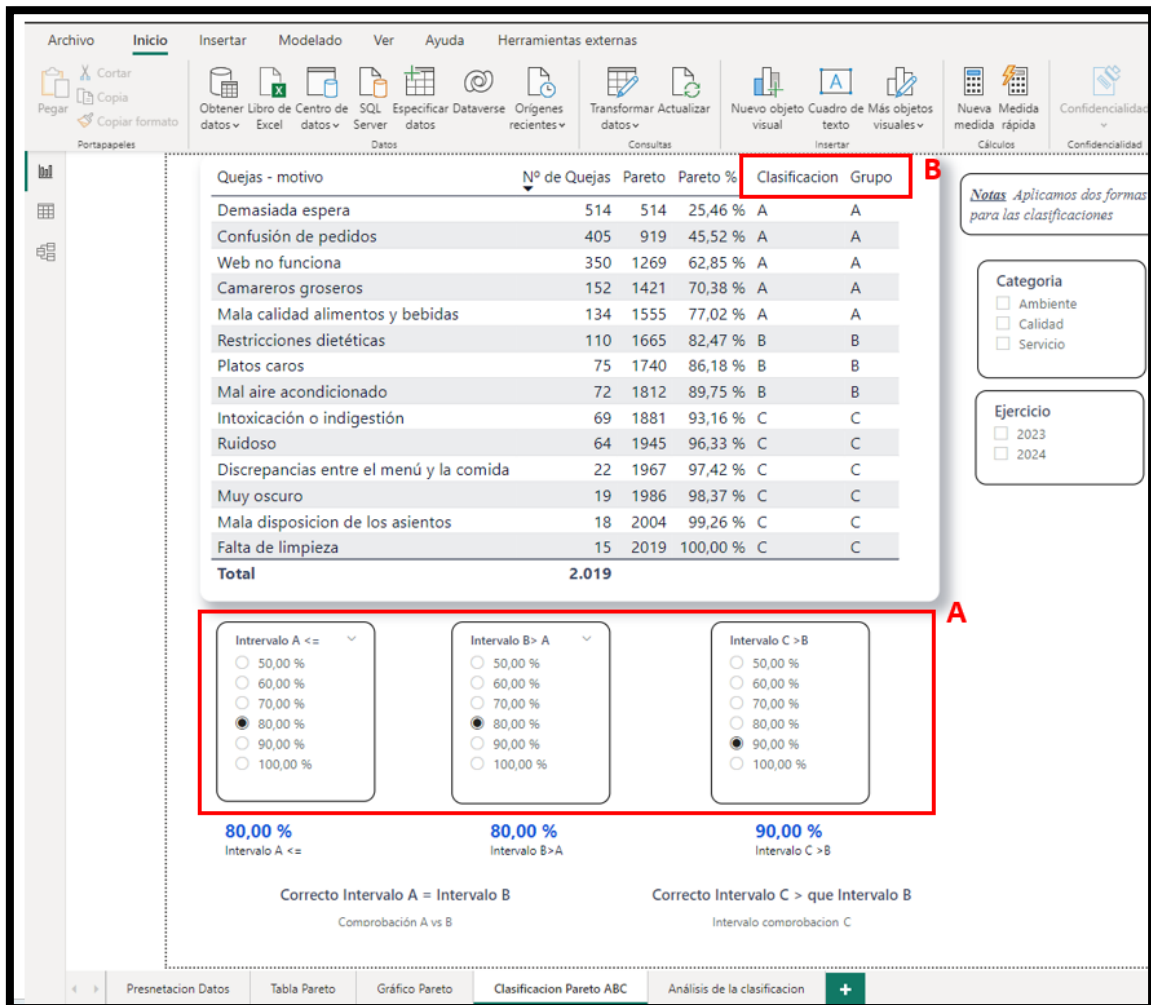


## 4 Perfeccionando el modelo. Clasificación Pareto ABC y análisis del resultado de la clasificación

### 4.1 Introducción

Con el fin de completar el análisis de Pareto incorporamos la clasificación ABC, que aunque por defecto se toma como referencia, como hemos señalado anteriormente,  $A \leq 80\%$  -  $B > 80\%$  y  $C > 90\%$  para los valores acumulados de la variable objeto de estudio, pero no necesariamente tiene que ser así, puede establecerse una clasificación o intervalos personalizados o grupos dinámicos adaptados por el usuario según criterio.

Así nuestro objetivo es cuadro de mando como el que aparece en la siguiente ilustración donde tendremos que definir las medidas que asignan a cada grupo o clasificación (A, B o C) y los filtros o criterios que definen los intervalos.



### 4.2 Parámetros de intervalos para el grupo A, B y C.

#### 4.2.1 Definición de los intervalos con parámetros

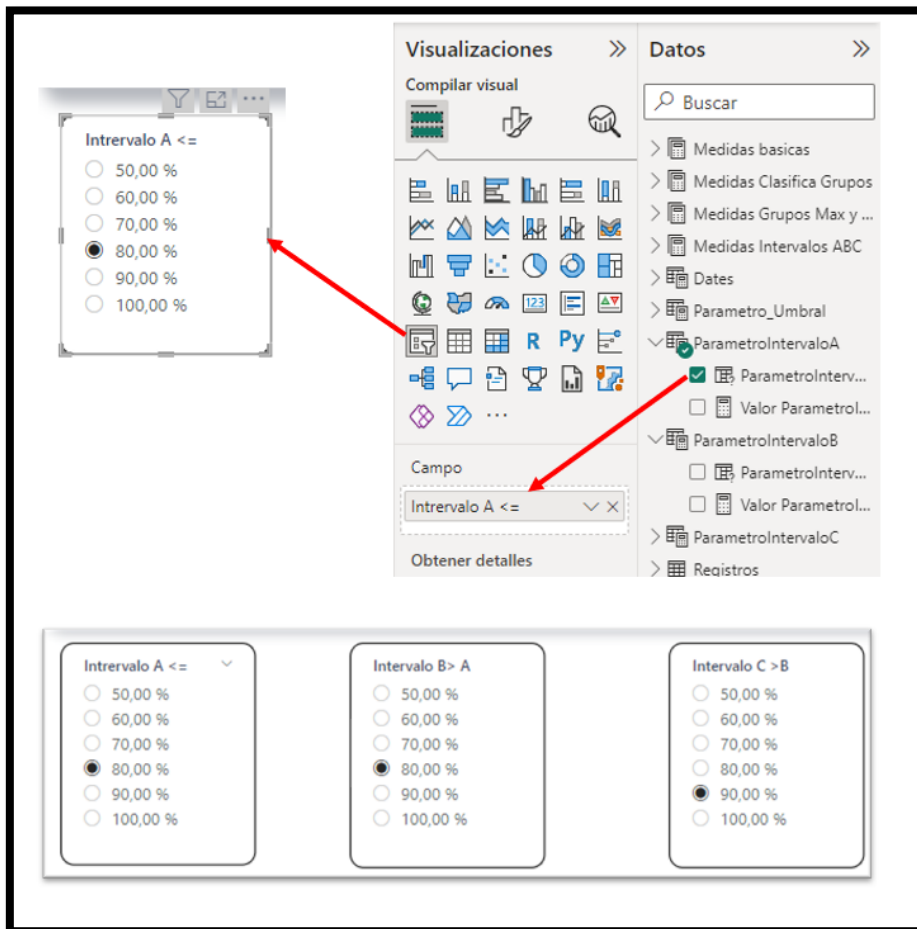
Comenzaremos en primer lugar por establecer los límites de los intervalos para la agrupación en las tres categorías y para que sea dinámico, es decir que el usuario los establezca, haremos uso de parámetros, definiendo uno para cada intervalo siendo su configuración la misma.

Así el parámetro que vamos a definir para el intervalo A tendrá la siguiente configuración.



La misma configuración tendrán los otros dos parámetros, ParametroIntervaloB y ParametroIntervaloC. Así tendremos creado los tres intervalos de grupo, A-B-C

Una vez creados los parámetros, asignamos cada uno de ellos a un segmentador de datos tal como se muestra en la siguiente ilustración y que permitirá al usuario seleccionar los valores correspondientes a cada intervalo para el grupo A, B y C.



Señalar una cuestión relevante para que la configuración de los intervalos sea correcta es necesario que se cumplan las siguientes reglas.

- El intervalo A  $\leq$  al valor seleccionado, por ejemplo el valor acumulado debe ser igual o inferior al 80%.
- El intervalo B  $>$  A debe ser superior al A, por tanto, se debe garantizar que el valor seleccionado en B sea igual al A
- El intervalo C  $>$  B

The screenshot shows a configuration interface with three columns of radio button options:

- Column 1:** "Intervalo A <=" with options 50,00 %, 60,00 %, 70,00 %, 80,00 % (selected), 90,00 %, 100,00 %. Below it, "80,00 %" is displayed in blue, followed by "Intervalo A <=".
- Column 2:** "Intervalo B > A" with options 50,00 %, 60,00 %, 70,00 %, 80,00 % (selected), 90,00 %, 100,00 %. Below it, "80,00 %" is displayed in blue, followed by "Intervalo B > A".
- Column 3:** "Intervalo C > B" with options 50,00 %, 60,00 %, 70,00 %, 80,00 %, 90,00 % (selected), 100,00 %. Below it, "90,00 %" is displayed in blue, followed by "Intervalo C > B".

At the bottom, two validation messages are shown:

- "Correcto Intervalo A = Intervalo B" with "Comprobación A vs B" below it.
- "Correcto Intervalo C > que Intervalo B" with "Intervalo comprobacion C" below it.

Así en el ejemplo quedaría que Grupo A hasta el 80% del valor acumulado, Grupo B del 80% hasta el 90% y del Grupo C será a partir del 90%.

Otro ejemplo correcto sería.

The screenshot shows a configuration interface with three columns of radio button options:

- Column 1:** "Intervalo A <=" with options 50,00 %, 60,00 % (selected), 70,00 %, 80,00 %, 90,00 %, 100,00 %. Below it, "60,00 %" is displayed in blue, followed by "Intervalo A <=".
- Column 2:** "Intervalo B > A" with options 50,00 %, 60,00 % (selected), 70,00 %, 80,00 %, 90,00 %, 100,00 %. Below it, "60,00 %" is displayed in blue, followed by "Intervalo B > A".
- Column 3:** "Intervalo C > B" with options 50,00 %, 60,00 %, 70,00 %, 80,00 % (selected), 90,00 %, 100,00 %. Below it, "80,00 %" is displayed in blue, followed by "Intervalo C > B".

At the bottom, two validation messages are shown:

- "Correcto Intervalo A = Intervalo B" with "Comprobación A vs B" below it.
- "Correcto Intervalo C > que Intervalo B" with "Intervalo comprobacion C" below it.

Sin embargo no sería correcto en los siguientes casos...

Intervalo A <=

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**60,00 %**  
Intervalo A <=

Intervalo B > A

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**70,00 %**  
Intervalo B > A

Intervalo C > B

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**80,00 %**  
Intervalo C > B

**ERROR Intervalo A tiene que ser igual a Intervalo B**

Comprobación A vs B

**Correcto Intervalo C > que Intervalo B**

Intervalo comprobacion C

Intervalo A <=

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**90,00 %**  
Intervalo A <=

Intervalo B > A

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**90,00 %**  
Intervalo B > A

Intervalo C > B

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**90,00 %**  
Intervalo C > B

**Correcto Intervalo A = Intervalo B**

Comprobación A vs B

**ERROR Intervalo C tiene que ser mayor a Intervalo B**

Intervalo comprobacion C

#### 4.2.2 Medidas de intervalos. Comprobacion de que los intervalos están correctamente definidos y advertencias

En estos casos para verificar y advertir al usuario que estos intervalos están correctamente definidos hemos creado unas medidas para su comprobación y que nos permita activar la advertencia o conformidad. Asi tenemos

Intervalo A-B = [Valor ParametroIntervaloA]-[Valor ParametroIntervaloB]

Nos indica si el resultado es 0 o distinto de 0, si es 0 es correcto en caso contrario error.

Intervalo B-C = [Valor ParametroIntervaloB]-[Valor ParametroIntervaloC]

En este caso debe ser mayor que 0 para que sea correcto, es decir garantizar que el valor seleccionado de C sea superior al B.

Estas serán las reglas basicas, de valoracion y ahora vamos a ver las dos medidas que nos faltan de comprobación.

Intervalo comprobacion\_B =

```
IF([Valor ParametroIntervaloA]<>[Valor ParametroIntervaloB], "ERROR Intervalo A
tiene que ser igual a Intervalo B",
"Correcto Intervalo A = Intervalo B"
)
```

Datos

Buscar

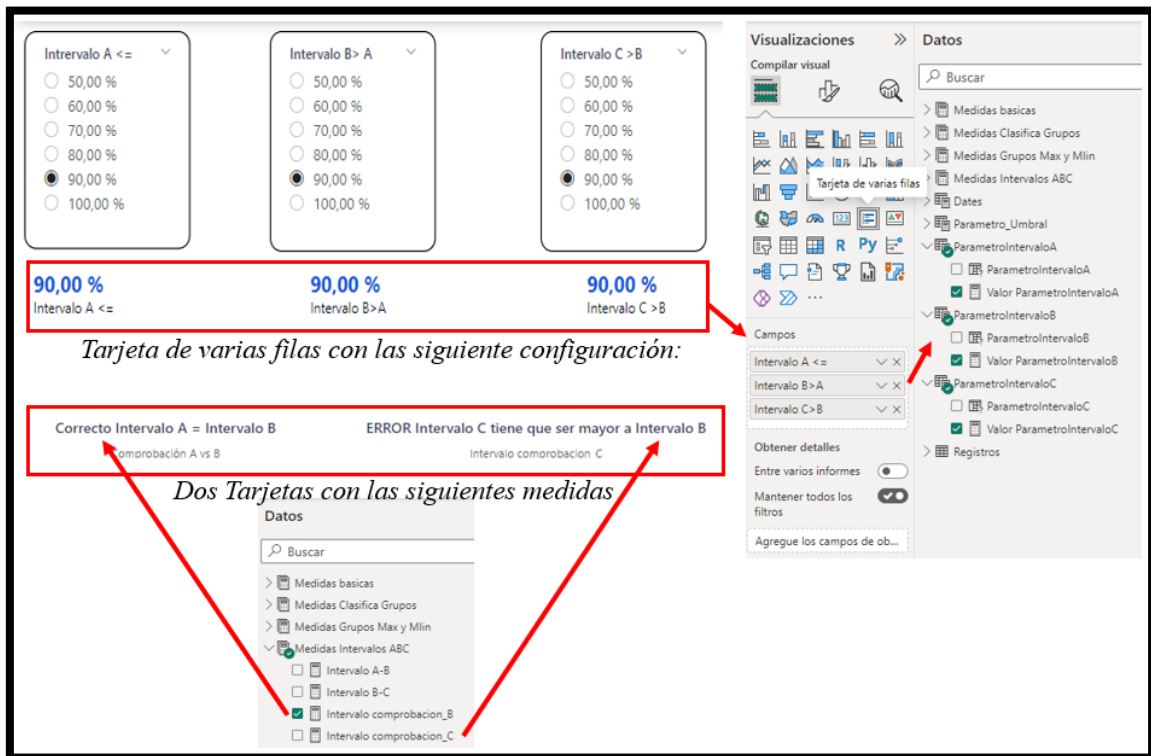
- Medidas basicas
- Medidas Clasifica Grupos
- Medidas Grupos Max y Mlin
- Medidas Intervalos ABC
  - Intervalo A-B
  - Intervalo B-C
  - Intervalo comprobacion\_B
  - Intervalo comprobacion\_C
- Datos
- Parametro\_Umbra
- ParametroIntervaloA
- ParametroIntervaloB
- ParametroIntervaloC
- Registros

Esta medida comprueba y garantiza que B sea igual a A y en ese caso advierte del error. Lo mismo sería para la otra medida.

**Intervalo comprobacion\_C =**

```
IF([Valor ParametroIntervaloC]<=[Valor ParametroIntervaloB],"ERROR Intervalo C
tiene que ser mayor a Intervalo B",
"Correcto Intervalo C > que Intervalo B"
)
```

Hemos diseñado una tarjeta de varias filas con el fin de mostrar los elementos o valores seleccionados (Valor del Parámetro de Intervalo Seleccionado)



Con la creación de las medidas Intervalo de Comprobación B y C se pretende advertir del posible error en la configuración de los intervalos, estas estarían asignadas a dos tarjetas.

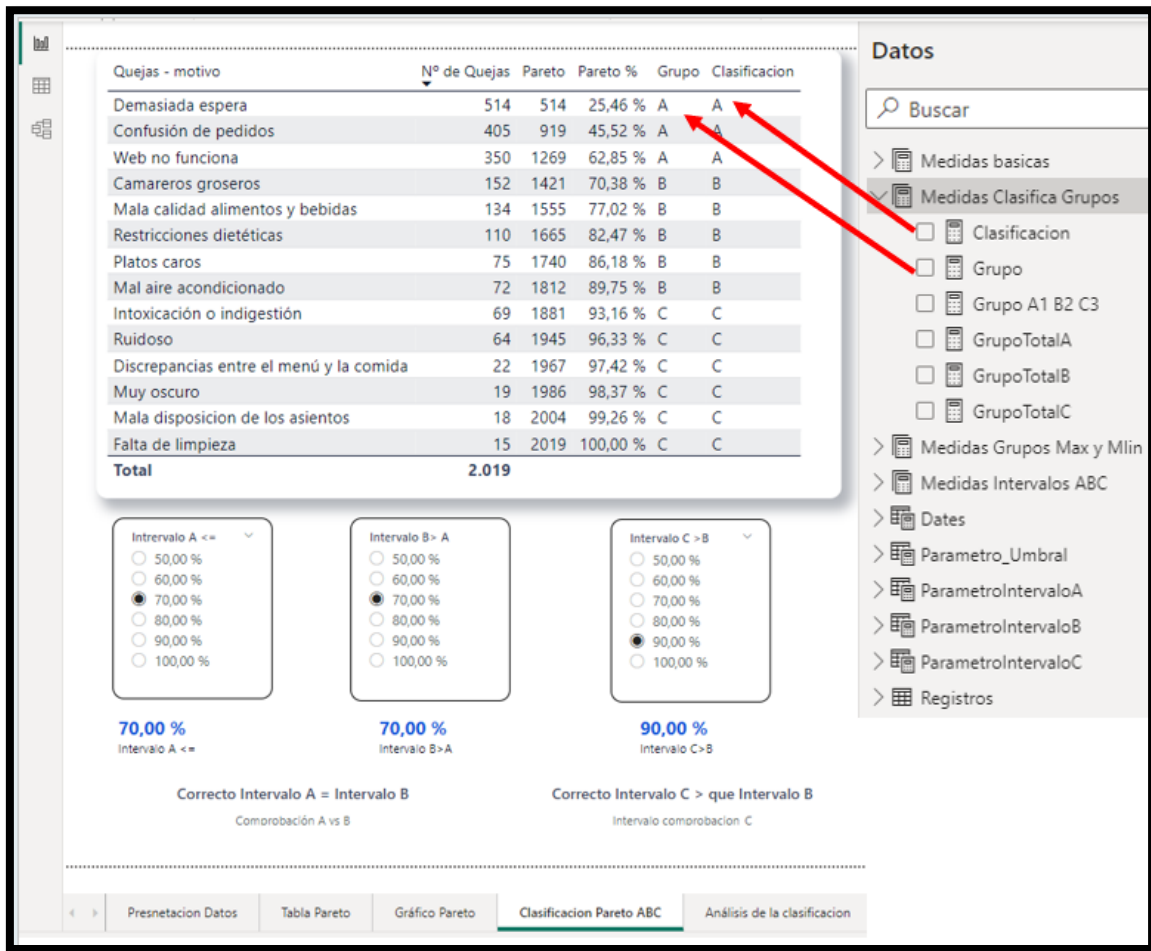
### 4.3 Creación de las medidas dinámicas de clasificación

#### 4.3.1 Grupo de medidas “Medidas Clasifica Grupos”. Clasificación y Grupo

Con este grupo de medidas pretendemos clasificar cada uno de las filas o registros de la tabla, en nuestro caso el tipo de queja, en función el valor acumulado y el intervalo correspondiente (A,B,C) ajustado a la configuración según los parámetros establecidos.

Así para la medida **Clasificación** se propone el siguiente código:

```
Clasificacion = If(HASONEVALUE(Registros[Quejas])=TRUE(),
SWITCH(TRUE(),
[Pareto %]>ParametroIntervaloC[Valor ParametroIntervaloC],"C",
[Pareto %]>ParametroIntervaloB[Valor ParametroIntervaloB],"B",
[Pareto %]<=ParametroIntervaloA[Valor ParametroIntervaloA],"A",
BLANK()))
```



Esta mismo resultado lo podemos obtener con la medida **Grupo** integrado en el código inicial del apartado 2.2 Código propuesto, tal y como presentamos a continuación:

```

1 Grupo =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4   VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5   VAR _TablaVirtualTrabajo =
6     ADDCOLUMNS (
7       _SeleccionColumnaTrabajo,
8       "@NumDeQuejas", [Quejas_Cuenta],
9       "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10      | RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11    )
12   VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo) , [@Rank] )
13   VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [@NumDeQuejas] )
14   VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15   VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado )
16   VAR _Grupo = SWITCH(TRUE(),
17     _ParetoPorcentaje>ParametroIntervaloC[Valor ParametroIntervaloC], "C",
18     _ParetoPorcentaje>ParametroIntervaloB[Valor ParametroIntervaloB], "B",
19     _ParetoPorcentaje<=ParametroIntervaloA[Valor ParametroIntervaloA], "A" )
20   RETURN
21   _Grupo
22 )

```

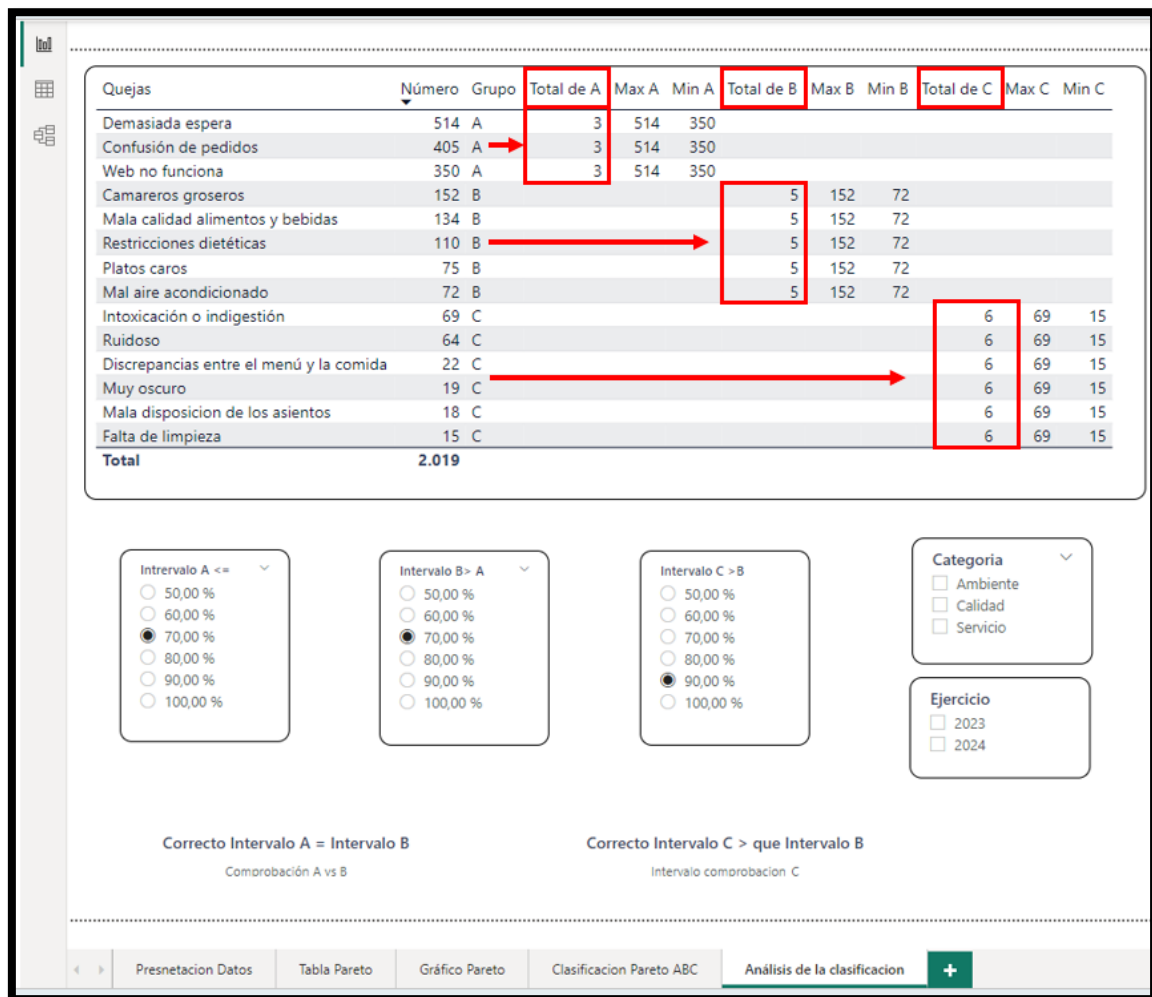
Donde destaca la adaptación realizada para determinar el grupo de pertenencia para lo cual hemos creado la medida **\_Grupo** que le asigna el valor a cada registro o fila tal y como se muestra a partir de la línea 16.

## 4.4 Análisis del resultado de la clasificación

### 4.4.1 Consideraciones

Una vez realizada la clasificación y realizada la tabla anterior “Clasificación Pareto ABC” es relevante realizar un análisis del resultado de la clasificación según el diseño de intervalos establecidos tal y como presentamos en la siguiente ilustración, donde mostremos el total del elementos (tipos de queja) clasificadas en el grupo A (Total A) así como en el resto de grupos además de los valores máximos y mínimos por categoría como resultado de la configuración de los criterios de clasificación (intervalos).

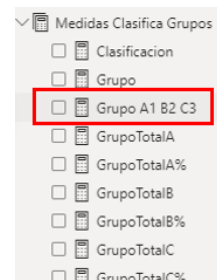
Es decir, presentamos como cuadro de mando para el “Análisis de clasificación” el propuesto en la siguiente ilustración.



### 4.4.2 Medida de conversión. Grupo A1 B2 C3

Una vez tenemos agrupado los elementos (tipos de quejas) según la clasificación ABC, interesa analizar estos resultados en términos agregados por categoría, es decir obtener el total de elementos que pertenecen a cada categoría (ABC) para ver la tasa o nivel de agrupamiento, esto supone usar la función DAX de COUNTROWS para contar el número de filas concretas condicionadas a la clasificación obtenida en el intervalo A,B o C.

La propuesta será crear una medida que llamaremos Grupo A1 B2 C3 y que simplemente convierte el A en 1, el B en 2 y el C en 3, tal como mostramos a continuación.





```

1 Grupo A1 B2 C3 = If(HASONEVALUE(Registros[Quejas])=TRUE(),
2 SWITCH(TRUE(),
3     [Grupo]="A",1,
4     [Grupo]="B",2,
5     [Grupo]="C",3,
6     BLANK()))

```

#### 4.4.3 Medidas Grupo Total A, Total B y Total C

De esta forma podremos contar el número de registros que están clasificados por cada categoría y para ello creamos las siguientes medidas.

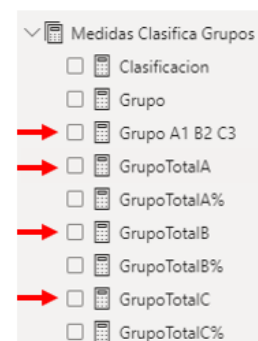
Hemos adaptado el código base a partir de la línea 20 que hemos creado la clasificación

```

1 GrupoTotalA =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4     VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5     VAR _TablaVirtualTrabajo =
6         ADDCOLUMNS (
7             _SeleccionColumnaTrabajo,
8             "@NumDeQuejas", [Quejas_Cuenta],
9             "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10            RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11        )
12     VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo) , [@Rank] )
13     VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [@NumDeQuejas] )
14     VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15     VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado)
16     VAR _Grupo = SWITCH(TRUE(),
17         _ParetoPorcentaje>ParametroIntervaloC[Valor ParametroIntervaloC],"C",
18         _ParetoPorcentaje>ParametroIntervaloB[Valor ParametroIntervaloB],"B",
19         _ParetoPorcentaje<=ParametroIntervaloA[Valor ParametroIntervaloA],"A")
20     // creo una tabla temporal con las clasificacion ABC
21     Var _TablaGruposABC=
22         ADDCOLUMNS (
23             _SeleccionColumnaTrabajo,
24             "@Clasifica", _Grupo,
25             "@QuejaTipo",Registros[Quejas],
26             "@123",[Grupo A1 B2 C3])
27     // Grupo A
28     Var _GrupoA=
29     DISTINCT(FILTER(_TablaGruposABC,[@Clasifica]="A"))
30     //creamos para cada una que cuente por tipo A,B y C
31     Var _TCategoriaA=
32     COUNTROWS(FILTER(_GrupoA,[@123]=1))
33     Return
34     _TCategoriaA
35 )

```

Igualmente hacemos para el resto de las medidas relacionadas, es decir, adaptamos las líneas de la 28 a la 35 para crear las otras medidas de totales, GrupoTotalB y GrupoTotalC.



#### 4.4.4 Medidas Grupo Total A%, Total B% y Total C%

Además de disponer del valor total de cada grupo nos interesa conocer el valor relativo que representa sobre el total y para ellos crearemos las siguientes medidas GrupoTotalA%, GrupoTotalB% y GrupoTotalC%, que es una simple modificación o ajuste de las anteriores, es decir de la de GrupoTotalA, GrupoTotalB y GrupoTotalC.

Así para el caso de la medida GrupoTotalA%, recuperamos la medida anterior GrupoTotalA y realizamos el siguiente ajuste.

```

1 GrupoTotalA% =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4   VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5   VAR _TablaVirtualTrabajo =
6     ADDCOLUMNS (
7       _SeleccionColumnaTrabajo,
8       "@NumDeQuejas", [Quejas_Cuenta],
9       "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10      | | | RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11    )
12   VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo) , [@Rank] )
13   VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [NumDeQuejas] )
14   VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15   VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado)
16   VAR _Grupo = SWITCH(TRUE(),
17     _ParetoPorcentaje>ParametroIntervaloC[Valor ParametroIntervaloC],"C",
18     _ParetoPorcentaje>ParametroIntervaloB[Valor ParametroIntervaloB],"B",
19     _ParetoPorcentaje<=ParametroIntervaloA[Valor ParametroIntervaloA],"A")
20   // creo una tabla temporal con las clasificaciones ABC
21   Var _TablaGruposABC=
22     ADDCOLUMNS (
23       _SeleccionColumnaTrabajo,
24       "@Clasifica", _Grupo,
25       "@QuejaTipo",Registros[Quejas],
26       "@123",[Grupo A1 B2 C3])
27   /* Creamos una variable para determinar el total de elementos únicos clasificados y no repetidos */
28   Var _TotalElementosUnicos =
29     COUNTROWS(_TablaGruposABC)
30   // Grupo A
31   Var _GrupoA=
32     DISTINCT(FILTER(_TablaGruposABC,[@Clasifica]="A"))
33   //creamos para cada una que cuente por tipo A,B y C
34   Var _TCategoriaA=
35     COUNTROWS(FILTER(_GrupoA,[@123]=1))
36   // creamos la tasa para cada grupo
37   Var _GrupoATasa=
38     DIVIDE(_TCategoriaA,_TotalElementosUnicos)
39   Return
40   | _GrupoATasa
41 )

```

En concreto lo nuevo es la línea 27 a la 29 donde calculamos el total de elementos o tipos de quejas y el otro ajuste el comprendido entre las líneas 36 a la 40 donde calculamos la tasa de significación sobre el total o valor relativo.

Así tenemos como resultado no solo los valores absolutos para cada categoría sino el valor relativo respecto al total o significación.

Quejas	Número	Grupo	Total de A	GrupoTotalA%	Total de B	GrupoTotalB%	Total de C	GrupoTotalC%
Demasiada espera	514	A	2	14,3 %				
Confusión de pedidos	405	A	2	14,3 %				
Web no funciona	350	B			3	21,4 %		
Camareros groseros	152	B			3	21,4 %		
Mala calidad alimentos y bebidas	134	B			3	21,4 %		
Restricciones dietéticas	110	C					9	64,3 %
Platos caros	75	C					9	64,3 %
Mal aire acondicionado	72	C					9	64,3 %
Intoxicación o indigestión	69	C					9	64,3 %
Ruidoso	64	C					9	64,3 %
Discrepancias entre el menú y la comida	22	C					9	64,3 %
Muy oscuro	19	C					9	64,3 %
Mala disposición de los asientos	18	C					9	64,3 %
Falta de limpieza	15	C					9	64,3 %
<b>Total</b>	<b>2.019</b>							

#### 4.4.5 Medidas para máximos y mínimos por categoría.

Para completar el análisis del resultado de la clasificación vamos a incorporar unas nuevas medidas con los valores extremos para cada categoría. Así para el caso de la clasificación crearemos las medidas GrupoAMax y GrupoAMin que nos devolverá el valor máximo y mínimo de la variable objeto de estudio.

Igualmente procederemos para los elementos del grupo B y C.

El código de la medida del GrupoAMax está basado en la medida GrupoTotalA el cual podemos copiar y adaptar a partir de la línea 32 a la 38 donde definimos la variable que devuelve el valor máximo.

Datos >>

Buscar

- > Medidas basicas
- > Medidas Clasifica Grupos
- ▼ Medidas Grupos Max y Min
  - GrupoAMax
  - GrupoAMin
  - GrupoBMax
  - GrupoBMin
  - GrupoCMax
  - GrupoCMin

```

1 GrupoAMax =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4   VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5   VAR _TablaVirtualTrabajo =
6     ADDCOLUMNS (
7       _SeleccionColumnaTrabajo,
8       "@NumDeQuejas", [Quejas_Cuenta],
9       "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10      | | | RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11    )
12   VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo ) , [@Rank] )
13   VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [@NumDeQuejas] )
14   VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15   VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado )
16   VAR _Grupo = SWITCH(TRUE(),
17     _ParetoPorcentaje>ParametroIntervaloC[Valor ParametroIntervaloC],"C",
18     _ParetoPorcentaje>ParametroIntervaloB[Valor ParametroIntervaloB],"B",
19     _ParetoPorcentaje<=ParametroIntervaloA[Valor ParametroIntervaloA],"A")
20   // creo una tabla temporal con las clasificacion ABC
21   Var _TablaGruposABC=
22     ADDCOLUMNS (
23       _SeleccionColumnaTrabajo,
24       "@Clasifica", _Grupo,
25       "@QuejaTipo",Registros[Quejas],
26       "@123",[Grupo A1 B2 C3],
27       "@TotalQuejas",[Quejas_Cuenta]
28     )
29   // Grupo A
30   Var _GrupoA=
31   DISTINCT(FILTER(_TablaGruposABC,[@Clasifica]="A"))
32   //creamos para cada uno Maximo y minimo
33   // Max del Grupo A
34   Var _GrupoAMax=
35   MAXX(FILTER(_GrupoA,[@123]=1),[@TotalQuejas])
36 Return
37 | _GrupoAMax
38 )

```

Cambiando solo estas mismas líneas obtenemos el mínimo tal y como se muestra a continuación.

```

1 GrupoAMin =
2 VAR _ColumnaTrabajo= SELECTEDVALUE( Registros[Quejas]) RETURN
3 IF( NOT ISBLANK( _ColumnaTrabajo),
4   VAR _SeleccionColumnaTrabajo = ALLSELECTED( Registros[Quejas] )
5   VAR _TablaVirtualTrabajo =
6     ADDCOLUMNS (
7       _SeleccionColumnaTrabajo,
8       "@NumDeQuejas", [Quejas_Cuenta],
9       "@Rank", RANKX ( _SeleccionColumnaTrabajo, [Quejas_Cuenta] )+
10      | | | RANKX ( _SeleccionColumnaTrabajo, Registros[Quejas])/1000000
11    )
12   VAR _MaxRankin = MAXX( FILTER( _TablaVirtualTrabajo , Registros[Quejas] = _ColumnaTrabajo) , [@Rank] )
13   VAR _ValorAcumulado = SUMX( FILTER( _TablaVirtualTrabajo , [@Rank] <= _MaxRankin ) , [@NumDeQuejas] )
14   VAR _TotalValorAcumulado = CALCULATE( [Quejas_Cuenta] , _SeleccionColumnaTrabajo )
15   VAR _ParetoPorcentaje = DIVIDE( _ValorAcumulado , _TotalValorAcumulado)
16   VAR _Grupo = SWITCH(TRUE(),
17     _ParetoPorcentaje>ParametroIntervaloC[Valor ParametroIntervaloC],"C",
18     _ParetoPorcentaje>ParametroIntervaloB[Valor ParametroIntervaloB],"B",
19     _ParetoPorcentaje<=ParametroIntervaloA[Valor ParametroIntervaloA],"A")
20   // creo una tabla temporal con las clasificacion ABC
21   Var _TablaGruposABC=
22     ADDCOLUMNS (
23       _SeleccionColumnaTrabajo,
24       "@Clasifica", _Grupo,
25       "@QuejaTipo",Registros[Quejas],
26       "@123",[Grupo A1 B2 C3],
27       "@TotalQuejas",[Quejas_Cuenta]
28     )
29   // Grupo A
30   Var _GrupoA=
31   DISTINCT(FILTER(_TablaGruposABC,[@Clasifica]="A"))
32   //creamos para cada uno Maximo y minimo
33   // Minimo del Grupo A
34   Var _GrupoAMin=
35   MINX(FILTER(_GrupoA,[@123]=1),[@TotalQuejas])
36 Return
37 _GrupoAMin
38 )

```

### 4.5 Tabla final del Análisis de la Clasificación

Por tanto, el resultado final del analisis de la clasificación queda agrupada en la siguiente propuesta.

Quejas	Número	Grupo	Total de A	Total de A%	Max A	Min A	Total de B	Total de B%	Max B	Min B	Total de C	Total de C%	Max C	Min C
Confusión de pedidos	405	A	2	14,3 %	514	405								
Demasiada espera	514	A	2	14,3 %	514	405								
Camareros groseros	152	B					3	21,4 %	350	134				
Discrepancias entre el menú y la comida	22	C									9	64,3 %	110	15
Falta de limpieza	15	C									9	64,3 %	110	15
Intoxicación o indigestión	69	C									9	64,3 %	110	15
Mal aire acondicionado	72	C									9	64,3 %	110	15
Mala calidad alimentos y bebidas	134	B					3	21,4 %	350	134				
Mala disposición de los asientos	18	C									9	64,3 %	110	15
Muy oscuro	19	C									9	64,3 %	110	15
Platos caros	75	C									9	64,3 %	110	15
Restricciones dietéticas	110	C									9	64,3 %	110	15
Ruidoso	64	C									9	64,3 %	110	15
Web no funciona	350	B					3	21,4 %	350	134				
<b>Total</b>	<b>2.019</b>													

**Intervalo A <=**

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**Intervalo B > A**

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**Intervalo C > B**

50,00 %

60,00 %

70,00 %

80,00 %

90,00 %

100,00 %

**Categoría**

Ambiente

Calidad

Servicio

Correcto Intervalo A = Intervalo B

Comorobación A vs B

Correcto Intervalo C > que Intervalo B

Intervalo comorobacion C

Presnetacion Datos
Tabla Pareto
Gráfico Pareto
Clasificacion Pareto ABC
Análisis de la clasificacion +