



# 220 Clientes sin pedidos con TD y formulas Dax

2 Casos TD y DAX Básico. Power Query, Power Pivot y medidas básicas.

Jose Ignacio González Gómez

Departamento de Economía, Contabilidad y Finanzas - Universidad de La Laguna

[www.jggomez.eu](http://www.jggomez.eu)

**V.2.2**

Ejercicio Basado: [David Asurmendi - Obtener los clientes que no me realizan pedidos usando DAX y Tablas Dinámicas. - YouTube](#)

Archivos fuentes: fichero Excel en carpeta 1 Datos 8 ficheros

Resumen: Comenzando a aprender Data Science y Business Intelligence con Excel aplicando Power Query, Power Pivot y medidas básicas con DAX.

Creando TD basado en un modelo de datos, nuestros primeros segmentadores y medidas basicas con DAX

Funciones DAX tratadas: Countrows, IF, ISBLANK

DAX, Power Query y Pivot, Tablas Dinámicas

## Contenido

- 1 Presentación..... 2
  - 1.1 Introducción ..... 2
  - 1.2 Sobre la información disponible ..... 2
- 2 Se pide..... 2
  - 2.1 Bases del modelo de datos para el analisis..... 2
  - 2.2 Cuadro de mando básico de analisis de pedidos de clientes..... 2
  - 2.3 Cuadro de mando de analisis de pedidos de clientes vs productos ..... 3
  - 2.4 Sobre los cálculos básicos necesarios ..... 4
- 3 Orientación, pasos sugeridos ..... 4
  - 3.1 Sobre el proceso ETL con Power Query ..... 4
  - 3.2 Creacion del modelo de datos (Power Pivot) y tabla calendario ..... 5
  - 3.3 Creacion de las medidas basicas con DAX para cuadro de mando 1 ..... 5
    - 3.3.1 TPedidos, COUNTROWS ..... 5
    - 3.3.2 Clientes0. IF, COUNTROWS, ISBLANK..... 6
    - 3.3.3 TPedidos v2. Combinando las dos medidas anteriores..... 6
  - 3.4 Creacion de las medidas basicas con DAX para cuadro de mando 2..... 6
- 4 Funciones DAX tratadas ..... 7
  - 4.1 COUNTROWS..... 7
  - 4.2 IF, ISBLANK ..... 8
    - 4.2.1 IF ..... 8
    - 4.2.2 ISBLANK ..... 9
- 5 Solución con captura de datos ..... 9
  - 5.1 Sobre el proceso ETL ..... 9
  - 5.2 Diseño del modelo de datos con Power Pivot ..... 10
- 6 Bibliografía ..... 13

## 1 Presentación

### 1.1 Introducción

FRUVE es una cooperativa agrícola dedicada a la comercialización de productos perecederos de frutas y verduras que opera en diversos países europeos.

Disponemos de la información comercial de los últimos ejercicios y nos interesa analizar los pedidos de los clientes y especialmente de los que no han realizado ninguno con el objeto de dirigir nuestras políticas de marketing hacia este segmento para mejorar las ventas.

### 1.2 Sobre la información disponible

Los datos están disponibles en ocho ficheros que deben examinarse previo a crear la conexión y posterior incorporación al modelo de datos y además sería conveniente identificar el tipo de tabla, es decir tabla de hechos o tabla de dimensión lo nos facilitara el diseño del modelo de datos.

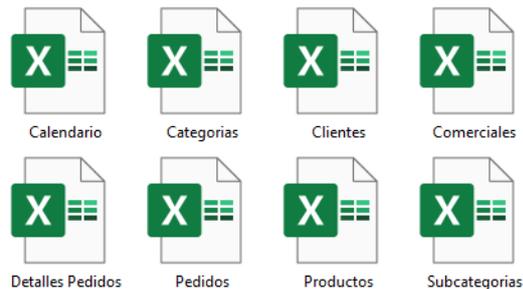


Ilustración 1

Respecto a las tablas disponibles, señalar simplemente lo siguiente:

1. Calendario. Se nos facilita una tabla calendario correspondiente a los periodos objeto de estudio, por tanto, no debemos crear ninguna
2. Categorías, tabla simple que contiene las categorías de los productos.
3. Clientes. Contiene información relacionada con el propio nombre de la tabla.
4. Comerciales, información relacionada.
5. Detalle de pedidos, contiene el detalle de cada uno de los pedidos (producto, cantidad, precio, descuento, etc)
6. Pedidos, información básica del pedido como es la fecha, cliente, comercial...
7. Productos, información relacionada.
8. Subcategorías, información relacionada.

## 2 Se pide

### 2.1 Bases del modelo de datos para el analisis.

Crear un libro Excel con el nombre Pedidos Fruve que contengan diferentes cuadros mando como los propuestos a continuación basado en tablas dinámicas, conectadas a las fuentes de datos facilitadas y agregadas al modelo de datos en Power Pivot, así como el crear los campos calculados en DAX necesarios

### 2.2 Cuadro de mando básico de analisis de pedidos de clientes.

Se requiere el diseño de un cuadro de mando para el analisis de los pedidos de clientes basado en una TD con dos segmentadores y conectado al modelo de datos tal y como se muestra en la siguiente ilustración. Para ello se recomienda crear una medida en DAX que llamaremos (TPedidos) que nos cuente el número de pedidos por cliente utilizando la formula COUNTROWS.

	A	B	C	D	E	F	G	H	I
1									
2									
3		Cientes	TPedidos						
4		Frutas Santiago S.L.	20						
5		Frutas Sol S.L.	11						
6		Frutas Sur S.L.	18						
7		Grupo Carrerf	20						
8		Grupo Hotelero Iber	29						
9		Obstladen Fruit	11						
10		Obstladen Park	3						
11		Restaurantes Asociados	26						
12		Via Fruta	6						
13		<b>Total general</b>	<b>144</b>						
14									

Ilustración 2

Esta medida se encuentra limitada en el sentido de que nos muestra el total de pedidos para aquellos clientes que han comprado, pero no muestra los clientes que **no han comprado**, información relevante que debemos tener en cuenta y para ello debemos crear una segunda medida que llamaremos Cliente0, como clientes que no han comprado y cuyo resultado de aplicar a la tabla dinámica será el siguiente:

	A	B	C	D	E	F	G	H	I
1									
2									
3		Cientes	TPedidos	Clientes0	T.Pedidos v2				
4		Fruits Fruits S.A.		0	0				
5		Frutas Santiago S.L.	20		20				
6		Frutas Sol S.L.	11		11				
7		Frutas Sur S.L.	18		18				
8		Fruterías Paco		0	0				
9		Grupo Carrerf	20		20				
10		Grupo Hotelero Iber	29		29				
11		Obstladen Fruit	11		11				
12		Obstladen Park	3		3				
13		Restaurantes Asociados	26		26				
14		Via Fruta	6		6				
15		<b>Total general</b>	<b>144</b>		<b>144</b>				

Ilustración 3

Esta opción permite crear una evolución de la medida TPedidos v2 como resultado de la combinación de ambas tal y como se muestra a continuación.

### 2.3 Cuadro de mando de analisis de pedidos de clientes vs productos

A continuación, queremos profundizar en el analisis anterior y conocer que compran y no compran nuestros clientes lo que nos permitirá un analisis más detallado de la demanda.

Para ello proponemos crear las mismas medidas que anteriormente, pero basada en la tabla Detalles\_Pedidos que contiene la información básica para el analisis (productos) y con los siguientes segmentadores, tal y como se muestra a continuación.

TPedidos_Productos	Productos	Albaricoque	Berenjena	Calabacín	Fresas	Mandarina	Melón	Total general
Fruits Fraits S.A.	0	0	0	0	0	0	0	0
Frutas Santiago S.L.	84	74	73	67	40	23	37	398
Frutas Sol S.L.	56	67	51	64	50	39	50	377
Frutas Sur S.L.	62	0	60	0	0	0	0	122
Fruterías Paco	0	0	0	0	0	0	0	0
Grupo Carrerf	56	47	55	81	20	23	22	304
Grupo Hotelero Iber	103	131	105	98	0	0	0	437
Obstladen Fruit	37	54	44	41	40	34	33	283
Obstladen Park	0	0	0	0	23	18	21	62
Restaurantes Asociados	70	87	82	85	34	29	30	417
Via Fruta	43	0	0	0	0	0	0	43
<b>Total general</b>	<b>511</b>	<b>460</b>	<b>470</b>	<b>436</b>	<b>207</b>	<b>166</b>	<b>193</b>	<b>2.443</b>

Ilustración 4

## 2.4 Sobre los cálculos básicos necesarios

Señalar que los cálculos básicos requeridos en las tablas dinámicas se pueden obtener de dos maneras diferentes.

- **La manera mas sencilla**, los podemos obtener cambiando la configuración por defecto de las tablas dinámicas que evita que se puedan mostrar resultados, relacionados con datos que no están en nuestro origen de datos. En este caso se trataría de las ventas que no se producen a determinados clientes. Los clientes sí que estarían en el origen de datos, pero las ventas no, porque no se han producido. Esta sería quizás la manera más sencilla de obtener este tipo de resultados.
- **Utilizando campos calculados con DAX (medidas DAX)**, que es la más idónea por su posible reutilización en otras tablas dinámicas. Es decir, en este caso, que será el que nos ocupe utilizaremos las tablas dinámicas con el complemento Power Query, Power Pivot y calculo básicos con DAX.

## 3 Orientación, pasos sugeridos

### 3.1 Sobre el proceso ETL con Power Query

Del analisis previo de los datos originales podemos concluir que las fuentes se encuentran correctamente depuradas y por tanto el proceso ETL se limitará a crear las conexiones a los ficheros lo que facilitará que sea fácilmente actualizable.

Asi el primer paso es acceder desde nuestro libro de trabajo (Pedidos Fruve) al editor de datos o Power Query (Datos-Obtener Datos- Iniciar Editor de Power Query) y conectar con los ficheros disponibles.

Una vez creada la conexión estas consultas deben ser agregada al modelo de datos para su posterior tratamiento y definición de relaciones creando asi el modelo de datos en Power Pivot.

## 3.2 Creacion del modelo de datos (Power Pivot) y tabla calendario

En este caso **no es necesario crear una tabla calendario** porque ya disponemos de ella, solo debemos garantizar que la tenemos activada o marcada por defecto como tabla calendario.

Destacar dentro de las relaciones entre las tablas en el modelo de datos la tabla calendario se relaciona con la tabla pedidos a traves del campo Fecha de la tabla Calendario con FechaApertura de la tabla Pedidos

## 3.3 Creacion de las medidas basicas con DAX para cuadro de mando 1

### 3.3.1 TPedidos, COUNTROWS

En primer lugar, vamos a crear una medida para nuestra TD que nos arroje el total de pedidos de cada cliente y que llamaremos "TPedidos".

Esta medida a crear con DAX es muy sencilla, lo que hace es contar de la tabla de pedidos el total de clientes que han realizado pedidos y para ello vamos a crearla con la función COUNTROWS(Pedidos)

Medida	
Nombre de la tabla:	Pedidos
Nombre de la medida:	TPedidos
Descripción del valor:	Cuenta el total de pedidos por clientes en el contexto
Fórmula:	<input type="text" value="=COUNTROWS(Pedidos)"/> <input type="button" value="Comprobar fórmula DAX"/>

Esto lo podríamos hacer directamente con las funciones recuento, pero nos interesa trabajar a traves de una medida, ya que nos va a permitir mayor flexibilidad en el trabajo, como veremos posteriormente.

A	B	C	D	E	F	G	H	I	J
1									
2	Todos los clientes		Cientes	TPedidos					
3	Fruits Fraits S.A.		Frutas Santiago S.L.		20				
4	Frutas Santiago S.L.		Frutas Sol S.L.		11				
5	Frutas Sol S.L.		Frutas Sur S.L.		18				
6	Frutas Sur S.L.		Grupo Carrerf		20				
7	Fruterías Paco		Grupo Hotelero Iber		29				
8	Grupo Carrerf		Obstladen Fruit		11				
9	Grupo Hotelero Iber		Obstladen Park		3				
10	Obstladen Fruit		Restaurantes Asociados		26				
11	Obstladen Park		Via Fruta		6				
12	Restaurantes Asociados		<b>Total general</b>		<b>144</b>				
13	Via Fruta								
14	<b>Total general</b>								

Ilustración 5

El resultado lo podemos ver en la ilustración anterior en la que de los 11 clientes que disponemos solo muestra 9, es decir aquellos clientes que han realizado pedidos, es decir, no muestra los clientes que no han realizado pedidos, por tanto, la función COUNTROWS tiene como limitación que los clientes que no tienen ventas no se muestran por que la función COUNTROWS devuelve vacío o en blanco y ese es un comportamiento que por defecto tienen las TD en el sentido que si no tienen un elementos con valores para mostrar como es vacío o blanco, no lo muestra en la TD.

En circunstancias normales podría ser correcto este comportamiento pero en otras ocasiones como las que nos ocupa si interesa mostrar estos valores cero correspondiente a los clientes sin pedidos.

### 3.3.2 Clientes0. IF, COUNTROWS, ISBLANK

Para solventar este inconveniente crearemos una segunda medida que llamaremos Clientes0 y que mostrara los clientes que están vacíos o en blanco y sustituyéndolo por 0,

$$=IF(ISBLANK([TPedidos]);0;BLANK())$$

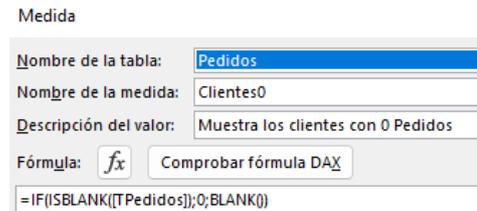


Ilustración 6

Por tanto, la fórmula es crear un condicional que cuando cuente columnas de clientes con valores vacíos o en blanco lo sustituya por 0 y de esta forma serán mostrados.

### 3.3.3 TPedidos v2. Combinando las dos medidas anteriores.

Finalmente podemos combinar las dos medidas anteriores creando una nueva que llamaremos TPedidos v2 y que puesta en nuestra TD obtendremos el siguiente resultado.

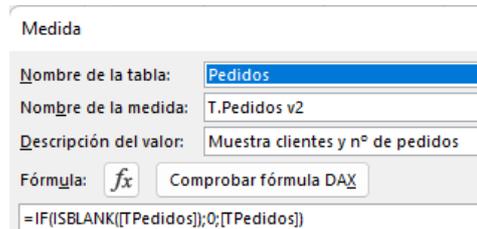


Ilustración 7

$$TPedidos v2 =IF(ISBLANK([TPedidos]);0;[TPedidos])$$

La fórmula anterior podría simplificarse empleando una sola medida:

$$TPedidos v3 =IF(ISBLANK(COUNTROWS(Pedidos));0;COUNTROWS(Pedidos))$$

	A	B	C	D	E	F	G	H	I	J
1										
2										
3		Cientes	TPedidos	Clientes0	T.Pedidos v2	TPedidos v3				
4		Fruits Fraits S.A.		0	0	0				
5		Frutas Santiago S.L.	20		20	20				
6		Frutas Sol S.L.	11		11	11				
7		Frutas Sur S.L.	18		18	18				
8		Fruterías Paco		0	0	0				
9		Grupo Carref	20		20	20				
10		Grupo Hotelero Iber	29		29	29				
11		Obstladen Fruit	11		11	11				
12		Obstladen Park	3		3	3				
13		Restaurantes Asociados	26		26	26				
14		Via Fruta	6		6	6				
15		<b>Total general</b>	<b>144</b>		<b>144</b>	<b>144</b>				

Ilustración 8

## 3.4 Creacion de las medidas basicas con DAX para cuadro de mando 2.

En este segundo caso y basado en la medida T.Pedidos v3 crearemos una nueva medida (que llamaremos **TPedidos\_Productos**) que actúe igual, es decir que nos cuente el total filas pero en este caso de la tabla Detalles\_Pedidos ya que nos interesa evaluar el total de pedidos de cada producto y cuyo resultado será el cuadro de mando propuesta en la Ilustración 4.

Medida		?
Nombre de la tabla:	<input type="text" value="Detalles_Pedidos"/>	
Nombre de la medida:	<input type="text" value="TPedidos_Productos"/>	
Descripción:	<input type="text" value="Contara el total de pedidos segun producto de la tabla Detalle de Pedidos"/>	
Fórmula:	<input type="text" value="fx"/>	<input type="button" value="Comprobar fórmula"/>
<pre>=IF(ISBLANK(COUNTROWS(Detalles_Pedidos));0;COUNTROWS(Detalles_Pedidos))</pre>		

Ilustración 9

## 4 Funciones DAX tratadas

### 4.1 COUNTROWS

#### [COUNTROWS | Interactive Chaos](#)

La función COUNTROWS cuenta el número de filas de una tabla.

Si la tabla no contiene ninguna fila, se devuelve un BLANK. Debe tenerse en cuenta que, para el cálculo, la función es contextualizada. Es decir, no calcula, en el sentido estricto del término, el número de filas de la tabla, sino el número de filas de la tabla una vez se aplica a ésta el contexto que corresponda.

#### **Ejemplo 1**

En este ejemplo se desea conocer el número de filas contenidas en la tabla FactSales, para lo que se define la siguiente medida: `NumVentas = COUNTROWS (FactSales)`

Si se muestra el resultado en una visualización tipo "card", obtenemos:

2.282.482  
NumVentas  
Ilustración 10

### Ejemplo 2

Si tenemos en FactSales un listado de ventas (correspondiendo cada fila a una venta) y definimos la siguiente medida: `Nº de ventas = COUNTROWS (FactSales)`

..podemos crear una matriz a la que llevar los años y la medida recién creada. Tal y como se ve en la siguiente imagen, la medida se contextualiza para cada uno de los diferentes años:

CalendarYear	Nº de ventas
2007	980.642
2008	697.026
2009	604.814
<b>Total</b>	<b>2.282.482</b>

Ilustración 11

## 4.2 IF, ISBLANK

### 4.2.1 IF

#### IF | Interactive Chaos

La función IF comprueba si una condición se cumple y devuelve un valor en caso positivo (si la evaluación de la condición devuelve el valor lógico True) y otro en caso negativo (si la evaluación de la condición devuelve el valor lógico False).  
Sintaxis:

```
IF(
    logical_test,
    value_if_true
    [, value_if_false]
)
```

### Ejemplo 1

En este ejemplo se evalúa cada venta como "Gran venta" o "Pequeña venta" en función de que el importe de la misma sea o no mayor que 100.000€:

Fecha	Precio	Tipo de venta
lunes, 15 de febrero de 2016	153.200 €	Gran venta
martes, 22 de marzo de 2016	90.500 €	Pequeña venta
viernes, 1 de abril de 2016	115.200 €	Gran venta
domingo, 3 de abril de 2016	80.420 €	Pequeña venta
jueves, 5 de mayo de 2016	32.480 €	Pequeña venta
viernes, 3 de junio de 2016	224.000 €	Gran venta

Ilustración 12

### Ejemplo 2

En este ejemplo no se incluye el argumento value\_if\_false. Para confirmar el tipo del resultado que la función IF devuelve en el caso de que logical\_test se evalúe como False, se añade una nueva columna personalizada con la fórmula: `Es blank = ISBLANK (Tabla1 [Tipo de venta])`

Fecha	Precio	Tipo de venta	Es blank
lunes, 15 de febrero de 2016	153.200 €		False
martes, 22 de marzo de 2016	90.500 €		True
viernes, 1 de abril de 2016	115.200 €		False
domingo, 3 de abril de 2016	80.420 €		True
jueves, 5 de mayo de 2016	32.480 €		True
viernes, 3 de junio de 2016	224.000 €		False

Ilustración 13

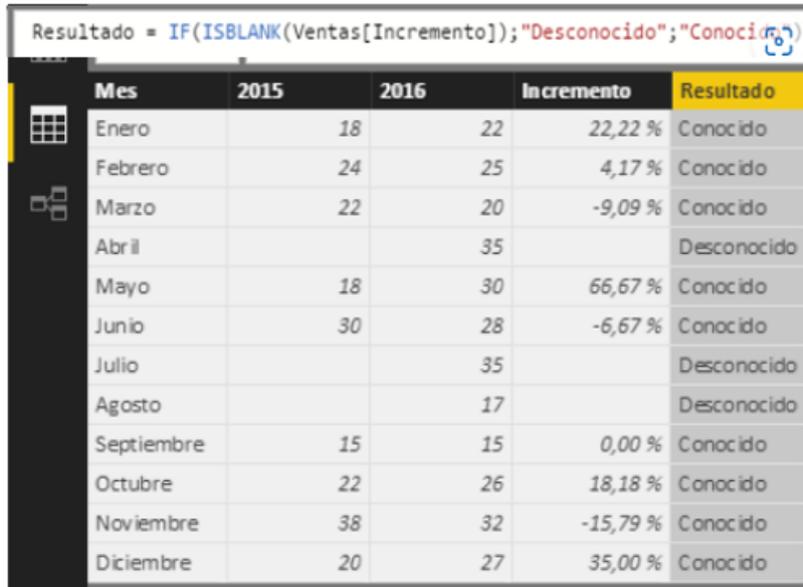
## 4.2.2 ISBLANK

### [ISBLANK | Interactive Chaos](#)

La función **ISBLANK** evalúa si un valor toma o no el valor *Blank* y devuelve *True* en caso positivo o *False* en caso negativo.

#### **Ejemplo**

En el siguiente ejemplo se calcula el incremento de ventas entre 2015 y 2016 y, en la última columna, se devuelve el texto "*Conocido*" o "*Desconocido*" en función de que el valor situado en la columna "*Incremento*" sea o no un *Blank*:



Mes	2015	2016	Incremento	Resultado
Enero	18	22	22,22 %	Conocido
Febrero	24	25	4,17 %	Conocido
Marzo	22	20	-9,09 %	Conocido
Abril		35		Desconocido
Mayo	18	30	66,67 %	Conocido
Junio	30	28	-6,67 %	Conocido
Julio		35		Desconocido
Agosto		17		Desconocido
Septiembre	15	15	0,00 %	Conocido
Octubre	22	26	18,18 %	Conocido
Noviembre	38	32	-15,79 %	Conocido
Diciembre	20	27	35,00 %	Conocido

Ilustración 14

## 5 Solución con captura de datos

### 5.1 Sobre el proceso ETL

En este caso procedemos a crear la conexión individualmente a cada fichero Excel disponible en la carpeta de datos.

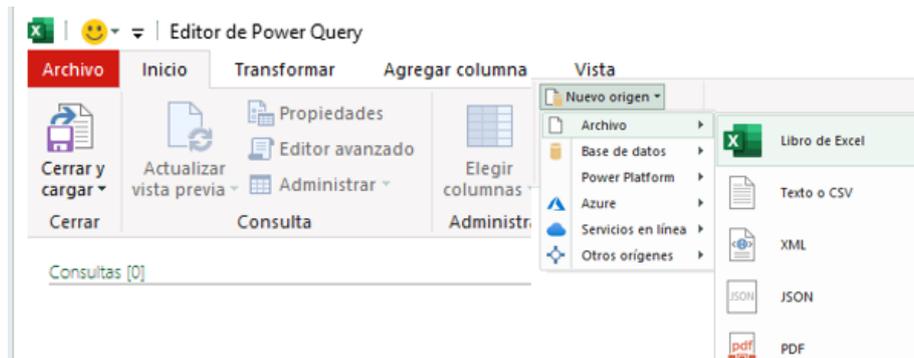


Ilustración 15

Así el resultado final debería ser similar a de la siguiente ilustración que contiene todas las conexiones a las fuentes, verificando simplemente que los campos, su nombre y tipo son los correctos.

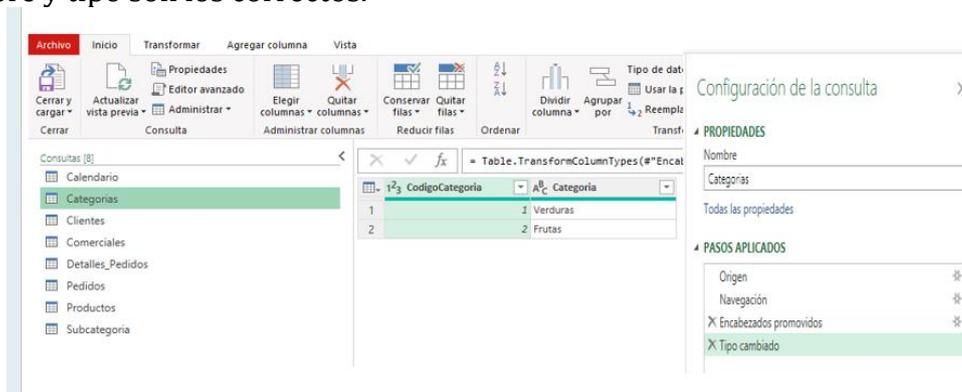


Ilustración 16

El paso final será acceder a la opción anterior de “Cerrar y Cargar” de tal forma que los datos se encuentran ya vinculados en nuestro libro de trabajo Pedidos Fruve.

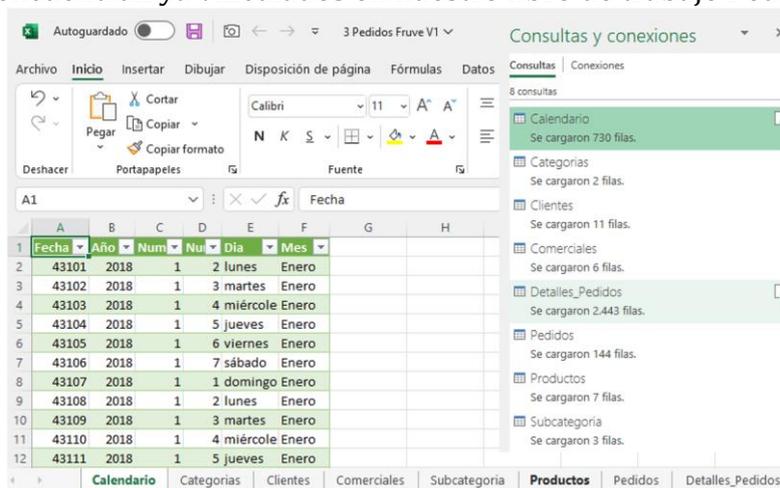


Ilustración 17

## 5.2 Diseño del modelo de datos con Power Pivot

Partiendo de las conexiones disponibles anteriores comenzaremos creando el modelo de datos agregando las mismas a través de la pestaña Power Pivot – Agregar a Modelo de Datos.

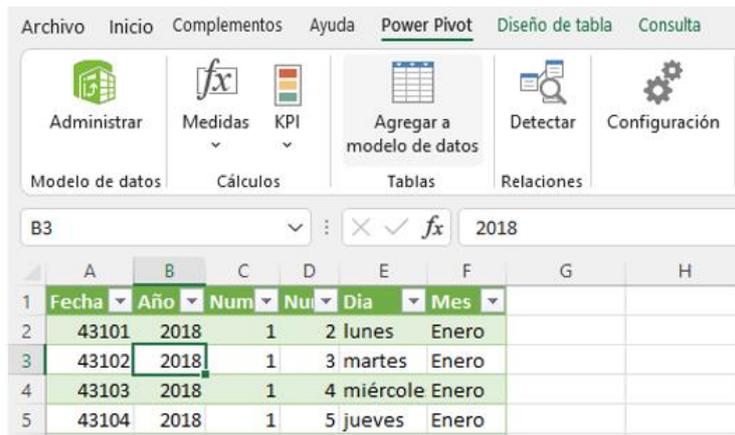


Ilustración 18

El resultado será la disposición de todas las tablas de información en el modelo de datos pero sin relacionar.

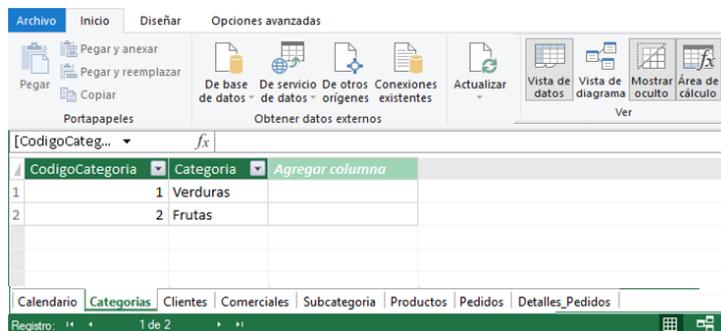


Ilustración 19

Un primer aspecto a considerar en el diseño del modelo de datos es el de disponer de una tabla calendario, en este caso vamos a seleccionar Calendario y marcarla como tabla calendario si no está establecida por defecto.

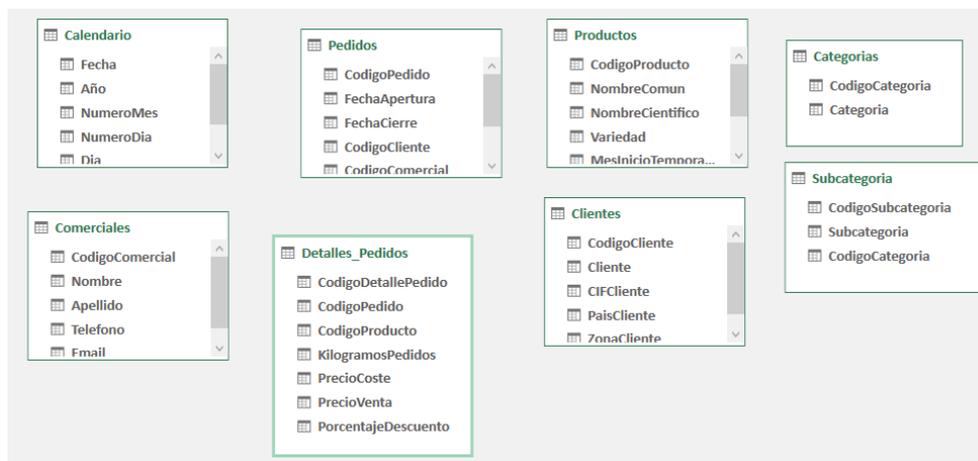


Ilustración 20

Vamos a crear las distintas conexiones entre las tablas, siempre desde las tablas de dimensión a las de hechos con un tipo de relación 1 a varios o varios a uno.

Comencemos creando las conexiones básicas comenzando con la relación central entre la tabla Calendario- Pedidos-Detalle\_Pedidos.

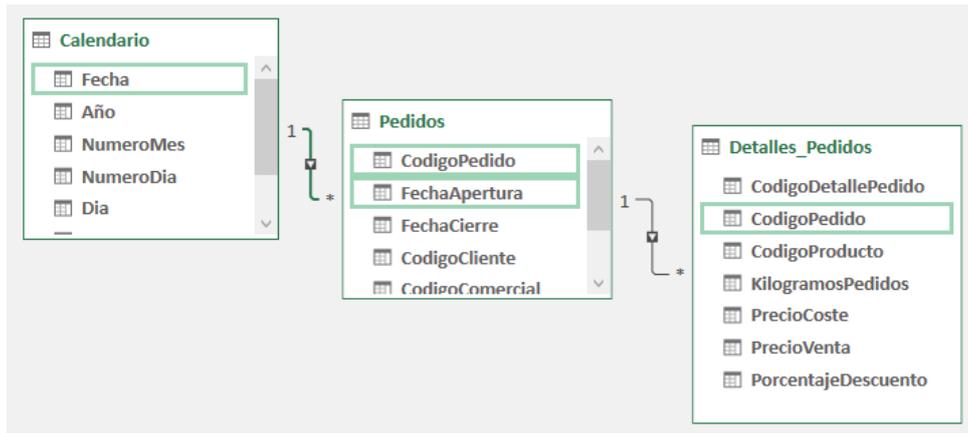


Ilustración 21

Finalmente, y siguiendo la lógica de las relaciones entre las tablas y los campos, el resultado final debe ser similar al siguiente:

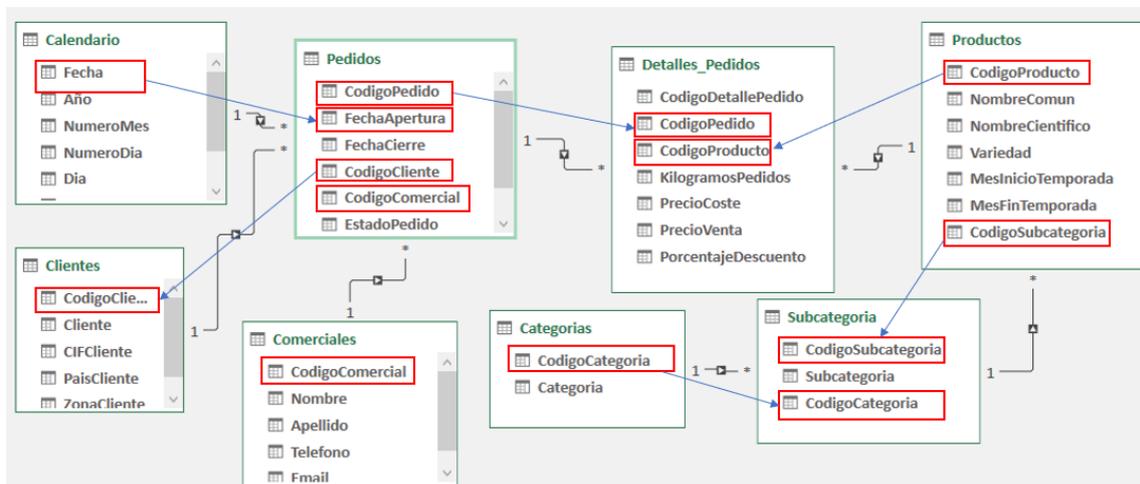


Ilustración 22

Y desde ahí agregar esta conexión al modelo de datos.

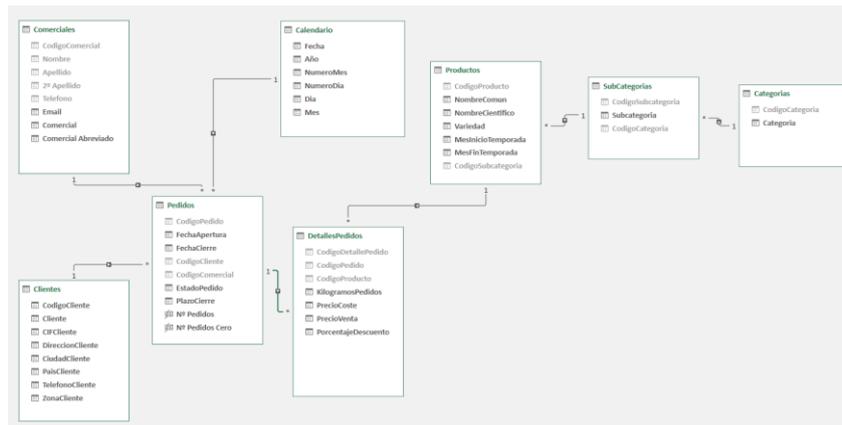


Ilustración 23

## 6 Bibliografía

[Funciones DAX | Interactive Chaos](#)

[Escenarios DAX | Interactive Chaos](#)

[Obtener los clientes que no me realizan pedidos usando DAX y Tablas Dinámicas. - YouTube](#)

[Obtener los clientes sin pedidos usando DAX y Tablas Dinámicas. – David Asurmendi](#)