# Filtrado y Agregación de un Catálogo Online

Práctica de el módulo de Bases de Datos No SQL

Laura Rodríguez Ropero



# Filtrado y Agregación de un Catálogo Online

Práctica de el módulo de Bases de Datos No SQL

por

# Laura Rodríguez Ropero

#### 07/11/2024

#### Índice

1.	Introducción y contexto	. 1
2.	Desarrollo de ejercicios	. 2
	2.1 Inserción y Actualización de Datos	. 2
	2.2 Proyección y Filtrado	
	2.3 Pipeline de Agregación	. 5
3	Conclusión	12

Profesor: Álvaro Bravo

Facultad: Facultad de Estudios Estadísticos
Universidad: Universidad Complutense de Madrid

Máster: Big Data, Data Science e Inteligencia Artificial



## Introducción y contexto

Actualmente, el análisis de datos provenientes de productos de moda es un área de interés creciente tanto para investigadores como para profesionales de la industria. Con el auge de las plataformas de comercio electrónico, las marcas y los minoristas recopilan grandes volúmenes de información sobre sus productos, que pueden incluir desde características físicas (como el tipo de prenda o el color) hasta descripciones detalladas y datos temporales (año de lanzamiento, temporada, entre otros). Estos datos se utilizan no solo para la gestión de inventarios y la optimización de ventas, sino también para identificar patrones de consumo, detectar tendencias de moda y mejorar la experiencia del cliente. Mi motivación para elegir este conjunto de datos es mi gran interés por el mundo de la moda y el diseño. Como soy una chica que acaba de llegar de Extremadura, me inspira la idea de que algún día pueda crear y patentar mi propia marca de moda.

Este trabajo se centra en el análisis de un **dataset de productos de moda**, que consta de **44,446 registros** de productos, cada uno con una serie de atributos clave. Entre estos atributos se incluyen:

- ID del producto: Identificador único para cada artículo.
- **Género**: Clasificación del producto por género (por ejemplo, "Men", "Women").
- Categoría principal (masterCategory): Tipo general del producto (como "Apparel" o "Accessories").
- Subcategoría (subCategory): Subdivisión más específica del tipo de producto (por ejemplo, "Shirts", "Watches", "Shoes").
- Color base (baseColour): Color principal del producto.
- Temporada (season): Época del año en la que el producto está destinado a ser usado.
- Año (year): Año en el que el producto fue lanzado o añadido al catálogo.
- Uso (usage): Contexto en el que el producto se utiliza, como "Casual" o "Formal".
- Nombre descriptivo del producto (productDisplayName): Nombre que describe el producto para su presentación al cliente.
- URL de la imagen (image\_url): Enlace a la imagen del producto, que permite visualizarlo.

El objetivo de este trabajo es realizar un análisis exhaustivo de este dataset, se utilizará **MongoDB** como base de datos y **MongoDB Compass** junto con **NoSQLBooster** como herramientas de cliente para la ejecución de consultas y agregaciones. Además, se emplearán herramientas como **Python** para la generación de visualizaciones.

MongoDB es una base de datos NoSQL que permite manejar grandes volúmenes de datos estructurados y no estructurados de manera eficiente, lo que la convierte en una opción ideal para datasets heterogéneos como el que se maneja en este proyecto.

## Desarrollo de ejercicios

A continuación, presento el desarrollo de los ejercicios de inserción, actualización, proyección, filtrado y agregación de datos que se han realizado sobre el dataset de productos elegido. Este análisis incluye varios scripts que permiten modificar y analizar la base de datos.

### 2.1. Inserción y Actualización de Datos

El dataset original se ha modificado para incluir las URLs de las imágenes asociadas a cada producto. Esta modificación es clave para que la información de las imágenes esté correctamente integrada en el análisis, y podamos realizar visualizaciones y consultas posteriores.

El siguiente script (Actualizar\_dataset.js) actualiza los productos existentes añadiendo el campo image\_url a partir de una colección separada de imágenes. Se conecta a MongoDB y itera sobre cada imagen para asociarla al producto correspondiente.

```
const { MongoClient } = require('mongodb');
3 async function updateProductsWithImages() {
    const uri = "mongodb://localhost:27017";
    const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: true });
    trv {
      await client.connect();
     const db = client.db('fashion_products');
     const productosCollection = db.collection('styles');
10
      const imagenesCollection = db.collection('images');
11
     const imagenes = await imagenesCollection.find({}).toArray();
13
     for (const imagen of imagenes) {
15
       const productId = parseInt(imagen.filename.split('.')[0]);
16
17
        await productosCollection.updateOne(
18
19
          { id: productId },
          { $set: { image_url: imagen.link } }
20
        );
21
23
      console.log("Actualización de productos con imágenes completada.");
    } catch (error) {
     console.error("Error al actualizar productos:", error);
    } finally {
26
27
      await client.close();
29 }
updateProductsWithImages();
```

#### Ejercicios sobre inserción y borrado de productos:

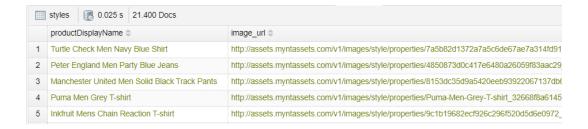
```
1 // Insertar un producto
2 db.styles.insertOne({
      "id": 60000,
      "gender": "Women",
"masterCategory": "Accessories",
4
5
      "subCategory": "Hats",
6
      "articleType": "Hats",
      "baseColour": "Black",
      "season": "Spring",
9
      "year": 2024,
10
      "usage": "Formal",
11
      "productDisplayName": "Beret",
12
      "image_url": "http://assets.myntassets.com/v1/images/style/properties/60000_images.jpg"
13
14 });
15
16 // Borrar el producto que acabamos de añadir
db.styles.deleteOne({
      "id": 60000,
18
      "gender": "Women",
      "masterCategory": "Accessories",
20
      "subCategory": "Hats"
21
      "articleType": "Hats",
22
      "baseColour": "Black",
23
      "season": "Spring",
24
      "year": 2024,
25
      "usage": "Formal",
26
      "productDisplayName": "Beret",
      "image_url": "http://assets.myntassets.com/v1/images/style/properties/60000_images.jpg"
28
29 });
```

#### Ejercicios sobre actualización de productos:

```
1 // Actualizar la categoría de los relojes para hombre a 'Luxury Accessories'
2 db.styles.updateMany(
     { "subCategory": "Watches", "gender": "Men" },
3
     { $set: { "masterCategory": "Luxury Accessories" } }
5);
7 // Volver a dejar la categoría como estaba
8 db.styles.updateMany(
     { "subCategory": "Watches", "gender": "Men" },
9
     { $set: { "masterCategory": "Accessories" } }
10
11 );
13 // Contar productos que no tengan url para comprobar que los hemos actualizado todos
      correctamente, y de no existir imagen para algún producto escribir 'Undefined' (aunque
      sabemos que todos lo tienen)
14 db.styles.find({ "image_url": { $exists: false } }).count();
db.styles.updateMany(
18);
```

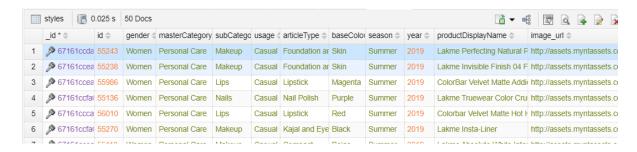
### 2.2. Ejercicios sobre Proyección y Filtrado

Mostrar sólo los productos con categoría 'Apparel' enseñando únicamente los campos image\_url y productDisplayName



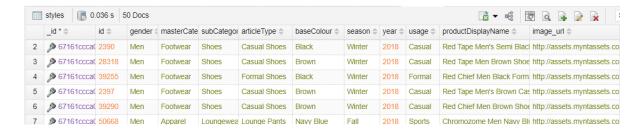
#### Ordenar por año de forma descendente y limitar la salida a las 50 primeras filas

db.styles.find({}).sort({ year: -1}).limit(50);



#### Ordenar por año de forma descendente y limitar la salida a las 50 segundas filas

db.styles.find({}).sort({ year: -1}).skip(50).limit(50);



#### Descartar productos que sean de años mayores de 2015 y menores de 2014 (\$nor)

db.styles.find({ \$nor: [ { "year" : { \$lt : 2014 } }, { "year" : { \$gt : 2015 } } ] });



## 2.3. Ejercicios sobre Pipeline de Agregación

Mostrar los 10 tipos de producto que tienen más variedad



Mostrar las 15 combinaciones baseColour-articleType predominantes y las menos comunes



styles 0.043 s 15 Docs				
	_id	totalProductos =		
	subCategory \$	baseColour \$	totalProductos	
1	Jewellery Set	Turquoise Blue	1	
2	Kajal and Eyeliner	Lavender	1	
3	Lounge Shorts	Charcoal	1	
4	Shorts	Off White	1	
5	Sports Shoes	Khaki	1	
6	Perfume and Body Mist	Olive	1	
7	Earrings	Olive	1	
8	Free Gifts	Steel	1	
9	Capris	Brown	1	
10	Lip Gloss	Gold	1	
11	Jeans	Brown	1	
12	Blazers	Charcoal	1	
13	Travel Accessory	Red	1	
14	Shapewear	Pink	1	
15	Face Wash and Cleanser	Red	1	

#### Mostrar el año u años con más y con menos productos

```
1 var fase1 = { $group: { "_id": "$year", "cuenta": { $sum: 1 } } };
2 var fase2 = { $group: { "_id": "$cuenta", "anios": { $push: "$_id" } };
3 var sortMax = { $sort: { "_id": -1 } }; // Para encontrar el máximo número de productos
4 var sortMin = { $sort: { "_id": 1 } }; // Para encontrar el mínimo número de productos
5 var limit = { $limit: 1 };
6 var unwind = { $unwind: "$anios" };
8 var pipelineMax = [ fase1, fase2, sortMax, limit, unwind ];
9 var pipelineMin = [ fase1, fase2, sortMin, limit, unwind ];
10
db.styles.aggregate([
12
13
        $facet: {
14
           "maxYear": pipelineMax,
           "minYear": pipelineMin
15
     }
17
18]);
```

Nos sale como año 'null' porque hay un producto que no le viene asignado ningún año, entonces vamos a añadir un match al principio para evitar esto

```
var matchValidYear = { $match: { year: { $ne: null } };
2 var fase1 = { $group: { "_id": "$year", "cuenta": { $sum: 1 } };
3 var fase2 = { $group: { "_id": "$cuenta", "anios": { $push: "$_id" } };
4 var sortMax = { $sort: { "_id": -1 } };
5 var sortMin = { $sort: { "_id": 1 } };
6 var limit = { $limit: 1 };
7 var unwind = { $unwind: "$anios" };
9 var pipelineMax = [ matchValidYear, fase1, fase2, sortMax, limit, unwind ];
10 var pipelineMin = [ matchValidYear, fase1, fase2, sortMin, limit, unwind ];
11
db.styles.aggregate([
13
    -{
14
       $facet: {
          "maxYear": pipelineMax,
15
          "minYear": pipelineMin
16
17
   }
18
19]);
```

```
0.104 s 1 Doc
styles
  1 * {
          "maxYear" : [
  2 🔻
  3 ₹
                   "_id" : 16290,
  4
                   "anios" : 2012
  5
  6
               }
  7
          ],
          "minYear" : [
  8 +
  9 +
               {
 10
                   "_id" : 1,
                   "anios" : null
 11
 12
               }
 13
          1
 14 }
```

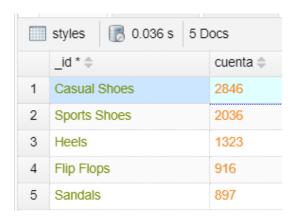
```
0.163 s 1 Doc
styles
  1 * {
  2 =
          "maxYear" : [
  3 ₩
              {
                   "_id" : 16290,
  4
  5
                   "anios" : 2012
  6
              }
  7
          ],
          "minYear" : [
  8 +
  9 +
              {
                   "_id" : 2,
 10
                   "anios" : 2007
 11
 12
              }
 13
          ]
 14 }
```

Agrupar por temporada y contar cuántos productos se lanzaron en cada una, ordenando de forma descendente

	styles   🕝 0.036 s   5 D	ocs
	_id * \$	cuenta 🗢
1	Summer	21476
	Fall	11445
3	Winter	8519
4	Spring	2985
5		21

Mostrar los 5 tipos de artículo de categoría Footwear de los que hay más productos

```
1 var fasefilter={ $match: { "masterCategory": "Footwear" } }
2 var fasegroup={ $group: { "_id": "$articleType", "cuenta": { $sum: 1 } } }
3 var fasesort={ $sort: { "cuenta": -1 } }
4 var faselimit={ $limit: 5 }
5 var etapas=[ fasefilter, fasegroup, fasesort, faselimit ]
6 db.styles.aggregate(etapas);
```



Mostramos las subcategorías junto con el número de productos que tiene cada subcategoría y la información del producto más reciente en esa subcategoría.

```
db.styles.aggregate([
       { $sort: { "year": -1 } },
2
       { $group: {
3
           _id: "$subCategory",
           totalProductos: { $sum: 1 },
5
           mostRecentProduct: { $first: "$$ROOT" }
6
      }},
      { $sort: { "_id": 1 } }, 
{ $project: {
8
9
           _id: 1,
10
           totalProductos: 1,
11
           \verb|"mostRecentProduct.productDisplayName": 1,\\
12
           "mostRecentProduct.image_url": 1,
13
           \verb|"mostRecentProduct.year": 1|\\
14
      }}
16]);
```

Quiero apuntar que al no haber mes, día ni hora de publicación van a existir muchos productos de cada subcategoría publicados en el último año de publicación de esta misma, cógeremos simbólicamente el primero de ellos.

	_id * \$	totalProdu	mostRecentProduct			
			year 🔺	productDisplayName	image_url \$	
34	Saree	427	2018	Xoxo Women Maroon Pure Georgette Solid Ready-to	http://assets.myntassets.com/assets/ima	
35	Sandal	963	2018	Red Chief Men Khaki Sandals	http://assets.myntassets.com/v1/images/	
36	Loungewear and N	470	2018	Jockey Women Charcoal Grey Lounge Pants	http://assets.myntassets.com/assets/ima	
37	Innerwear	1808	2018	Amante Women Pack of 2 Bikini Briefs PFFL03	http://assets.myntassets.com/v1/image/s	
38	Flip Flops	915	2018	Red Tape Men's Black Flip Flop	http://assets.myntassets.com/v1/images/	
39	Bottomwear	2694	2018	Jockey Women Grey Melange Knit Capris 1300	http://assets.myntassets.com/assets/ima	
40	Bags	3055	2018	Wildcraft Unisex Grey & Green Rucksack	http://assets.myntassets.com/v1/images/	
41	Nails	329	2019	Lakme Truewear Color Crush 04 Nail Polish	http://assets.myntassets.com/assets/ima	
42	Makeup	307	2019	Lakme Nine to Five Flawless Makeup Shell Foundation	http://assets.myntassets.com/assets/ima	
43	Lips	527	2019	Colorbar Velvet Matte Hot Hot Hot Red Lipstick 01R	http://assets.myntassets.com/assets/ima	
44	Eyes	43	2019	Colorbar Women Lash Illusion Mascara Duo DM001	http://assets.myntassets.com/assets/ima	
45	Bath and Body	12	2019	Colorbar Acetone Free Nail Enamel Remover	http://assets.myntassets.com/assets/ima	

Agrupar los productos por género y color base, mostrando los 5 colores más populares para cada género.

```
db.styles.aggregate([
    { $group: {
2
        _id: { gender: "$gender", baseColour: "$baseColour" },
        totalProductos: { $sum: 1 }
      }
5
    },
    { $sort: { totalProductos: -1 } },
7
8
    { $group: {
        _id: "$_id.gender",
        topColours: { $push: { colour: "$_id.baseColour", count: "$totalProductos" } }
10
11
    },
12
    { $project: { _id: 0, gender: "$_id", topColours: { $slice: ["$topColours", 5] } } }
13
```

Agrupar los productos por año y montrar el número total de productos lanzados por cada año.

Mostrar cómo se distribuyen los productos lanzados por año y estación

```
db.styles.aggregate([
2
   { $group: {
3
        _id: { year: "$year", season: "$season" },
        totalProductos: { $sum: 1 }
      }
    },
6
    { $sort: { "_id.year": 1, "_id.season": 1 } },
7
    { $project: {
        year: "$_id.year",
9
        season: "$_id.season",
10
        totalProductos: 1,
11
        _id: 0
12
13
  }
14
15]);
```





Mostrar el número de productos por género y las subcategorías más representativas de cada uno

```
db.styles.aggregate([
    { $match: { gender: { $in: ["Men", "Women"] } } },
2
    { $group: {
3
         _id: { gender: "$gender", subCategory: "$subCategory" },
4
        totalProductos: { $sum: 1 }
      }
6
    },
7
    { $sort: { totalProductos: -1 } },
    { $group: {
    _id: "$_id.gender",
9
10
        subCategories: {
11
12
          $push: {
             subCategory: "$_id.subCategory",
13
             totalProductos: "$totalProductos"
14
          }
15
16
        },
        totalProductos: { $sum: "$totalProductos" }
17
```

```
18
      }
    },
19
    { $project: {
20
         _id: 1,
21
         totalProductos: 1,
22
23
         firstSubCategory: \{ \ \$arrayElemAt: \ ["\$subCategories.subCategory", \ 0] \ \},
         secondSubCategory: { $arrayElemAt: ["$subCategories.subCategory", 1] },
24
         thirdSubCategory: { $arrayElemAt: ["$subCategories.subCategory", 2] },
25
      }
26
27
    { $addFields: {
28
29
         description: {
          $concat: [
30
             "En la categoría ", "$_id",
31
             ", la subcategoría más popular es ", "$firstSubCategory",
32
             ", seguida de ", "$secondSubCategory",
33
             "y ", "$thirdSubCategory", "."
34
35
        }
36
37
      }
38
    },
    { $project: { _id: 1, totalProductos: 1, description: 1 } }
39
```

	styles		Docs
	_id * \$	totalProductos	description $\Rightarrow$
1	Women	18632	En la categoría Women, la subcategoría más popular es Topwear, seguida de Shoes y Bags.
2	Men	22165	En la categoría Men, la subcategoría más popular es Topwear, seguida de Shoes y Watches.

Mostrar los productos de la colección Primavera-Verano del año 2008 para destacar las imágenes de estos mismos.













Adjunto por último aquí el código de python (Print\_images.py) del que he hecho uso para imprimir las imágenes de la última consulta.

```
1 from PIL import Image
2 import requests
3 from io import BytesIO
4 import matplotlib.pyplot as plt
6 # Lista de URLs de las imágenes
7 urls = [
      "http://assets.myntassets.com/v1/images/style/properties/Raymond-Men-Raymond-Leather-
8
          Belts-White-Belts_f942698ad78d2e13a4007024e6c73f3c_images.jpg",
      "http://assets.myntassets.com/v1/images/style/properties/US-Polo-Assn-Kids-Boys-Orange-T-
          shirt_a7c1b583f1c04c05f9bd6ffca5358241_images.jpg",
      "http://assets.myntassets.com/v1/images/style/properties/Reebok-Men-REALFLEX-OPTIMAL-
10
          White-Sports-Shoes_6e49515f918bc0fb66df906799386112_images.jpg",
      "http://assets.myntassets.com/v1/images/style/properties/Reid---Taylor-Men-Blue-Striped-
11
          {\tt Shirt\_b49d226c8a40fc2049dfb6280149e4ff\_images.jpg",}
      "http://assets.myntassets.com/v1/images/style/properties/Adidas-Men-SUKOI-White-Sports-
12
          Shoes_2d184f92b877bb5f06e2ba7c725590f7_images.jpg"
13 ]
14
15 # Descargar y mostrar cada imagen
16 images = []
17 for url in urls:
      response = requests.get(url)
19
      img = Image.open(BytesIO(response.content))
      images.append(img)
20
22 # Crear una cuadrícula de 1x5 para mostrar las imágenes
23 fig, axes = plt.subplots(1, 5, figsize=(15, 5))
24 for ax, img in zip(axes, images):
      ax.imshow(img)
25
26
      ax.axis('off') # Quitar los ejes para una mejor visualización
28 plt.show()
```

3

## Conclusiones

En este análisis del catálogo de productos de moda, se observó que las categorías de productos con mayor diversidad incluyen tipos como "Tshirts", "Shirts", "Casual Shoes" y "Watches", lo que sugiere que estos artículos son clave en la estrategia de la marca debido a su alta variedad. Al estudiar las combinaciones de color y tipo de artículo, se identificaron combinaciones predominantes como "White Tshirts", "Black Tshirts", y "Black Watches", destacando preferencias claras de color y tipo en la oferta. Mientras que en las combinaciones menos frecuentes podemos observar que se trata de colores y de tipos de artículo menos básicos, como "Turquoise Blue Jewelery Set", "Lavender Kajal and Eyeliner", y "Charcoal Lounge Shorts". En cuanto a la estacionalidad, cabe destacar la predominancia en cuanto a la publicación de artículos de las colecciones Verano y Primavera frente a Invierno y Otoño.

El análisis de los años con mayor y menor número de lanzamientos reveló que los años con más productos fueron 2011 y 2012, mientras que tanto años anteriores como posteriores, tuvieron un menor volumen de lanzamientos. Además, la consulta que agrupó productos por género y color base mostró que para las categorías "Men" y "Women", que son las más pobladas, los colores predominantes incluían el negro, el blanco, el azul y el marrón, reflejando una muy poco representativa segmentación de color basada en preferencias de género. Por último, las subcategorías más representativas para estos mismos géneros ("Men" y "Women") fueron "Topwear", "Shoes" y "Watches" para hombres, y "Topwear", "Shoes" y "Bags" para mujeres, lo que subraya las prioridades de inventario y la oferta enfocada en los productos más buscados por cada segmento.

Para enriquecer el análisis del catálogo de productos de moda y abordar cuestiones más profundas y personalizadas, propongo la inclusión de varios campos adicionales en el dataset. Uno de los campos que sugiero es el precio del producto, que permitiría explorar cómo el costo influye en la popularidad, identificar rangos de precios y evaluar la accesibilidad o exclusividad de ciertos artículos. Además, un campo de fecha de añadido posibilitaría un análisis temporal detallado (por día, mes o trimestre) y facilitaría la detección de patrones de lanzamiento en fechas clave, como festividades o temporadas altas. Incluir tamaño también sería útil, ya que permitiría identificar preferencias de talla en cada categoría, ajustar inventarios según demanda y analizar la rotación de productos con mayor precisión.

Para profundizar en la percepción del cliente, sugiero agregar una valoración promedio de usuarios junto con el número de reseñas asociadas daría una visión integral de la satisfacción del cliente y popularidad del producto, permitiendo segmentar por productos de alta o baja valoración. Asimismo, incluir un nivel de inventario que permitiría monitorear las existencias en tiempo real, identificar productos de alta demanda y planificar mejor los reabastecimientos. En el ámbito digital, un campo de popularidad en redes sociales ofrecería una perspectiva sobre la presencia en redes de cada producto, permitiendo correlacionar la popularidad en estas plataformas con las ventas y destacando productos con potencial viral.