

DataMgmtLab4

authors: Massimo Stefani, Eva Ray

Exercise 1

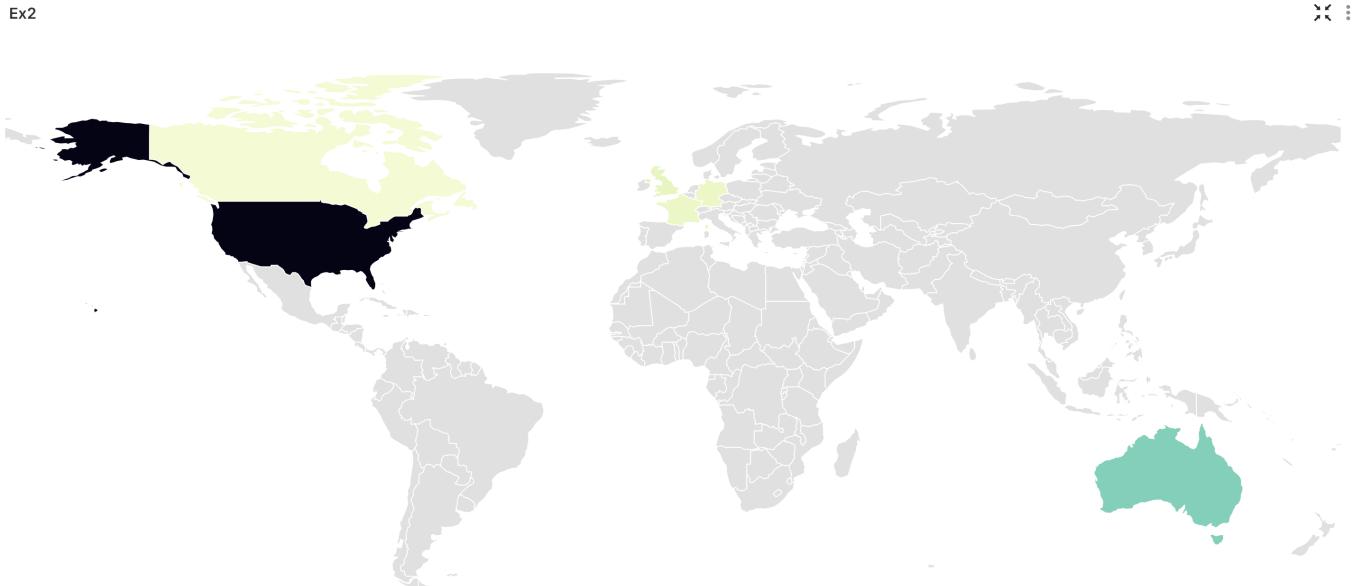
```
SELECT *
FROM dimcustomer AS c
INNER JOIN dimgeography AS g
    ON c.GeographyKey = g.GeographyKey
WHERE g.FrenchCountryRegionName = 'France';
```

1810

Nombre de clients français

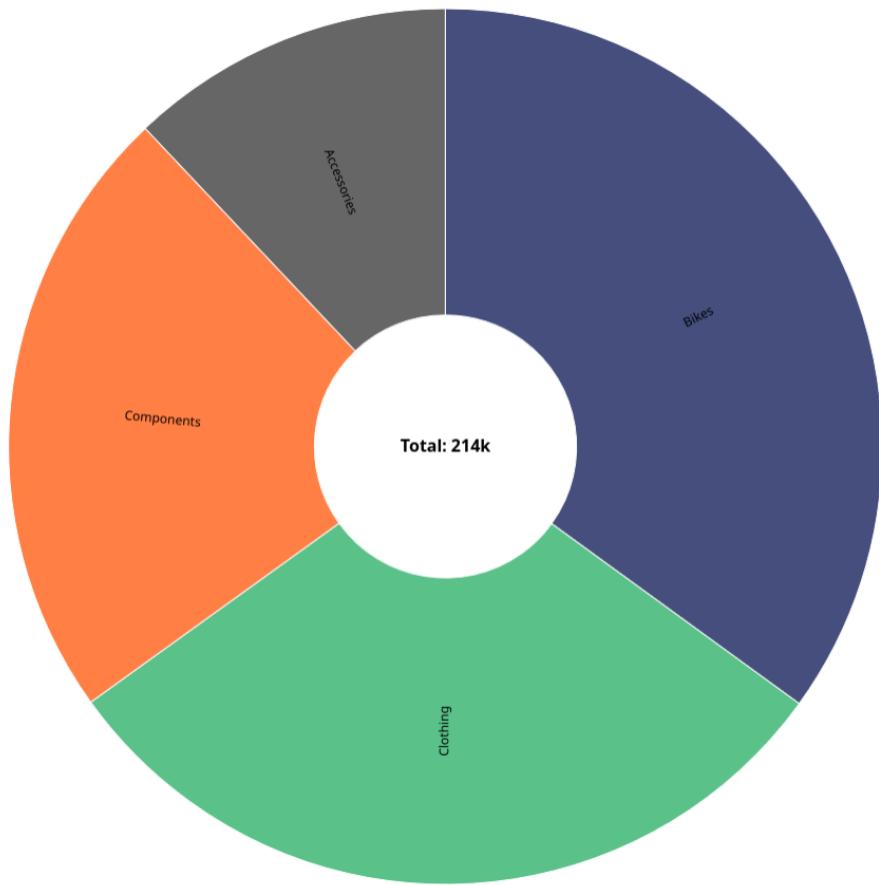
Exercise 2

```
SELECT dg.EnglishCountryRegionName
FROM dimcustomer dc
JOIN dimgeography dg
ON dc.GeographyKey = dg.GeographyKey
```



Exercise 3

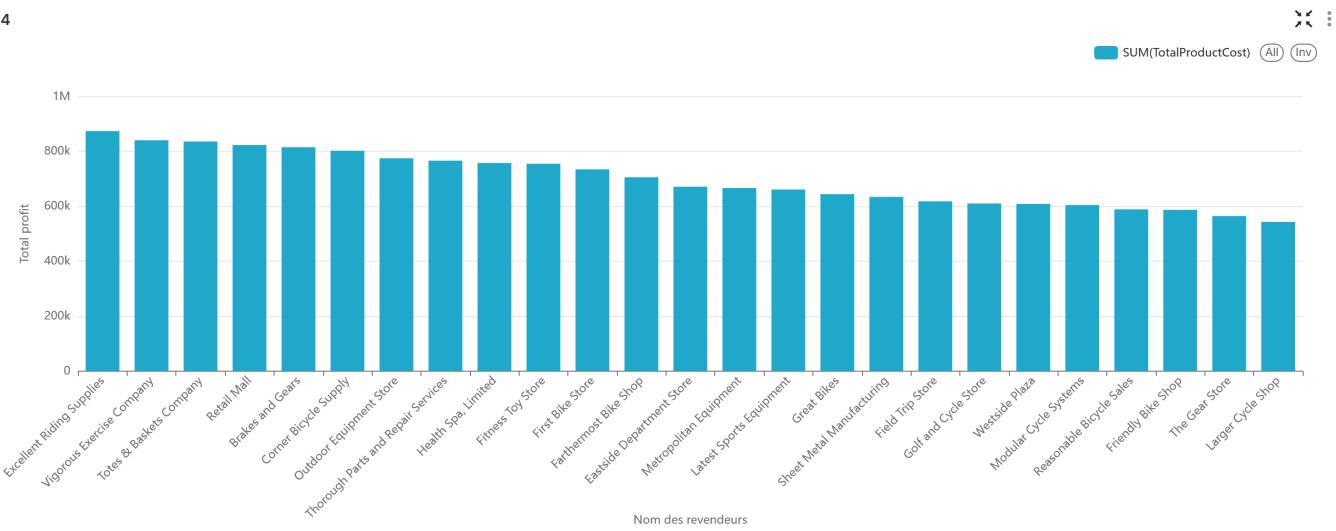
```
SELECT
    dpc.EnglishProductName,
    fs.OrderQuantity
FROM factresellersales fs
JOIN dimproduct dp
    ON dp.ProductKey = fs.ProductKey
JOIN dimproductsubcategory dpsc
    ON dp.ProductSubcategoryKey = dpsc.ProductSubcategoryKey
JOIN dimproductcategory dpc
    ON dpsc.ProductCategoryKey = dpc.ProductCategoryKey
```



Exercise 4

```
SELECT ds.ResellerName, frs.TotalProductCost
FROM dimreseller ds
JOIN factresellersales frs
ON ds.ResellerKey = frs.ResellerKey
```

Ex4



4

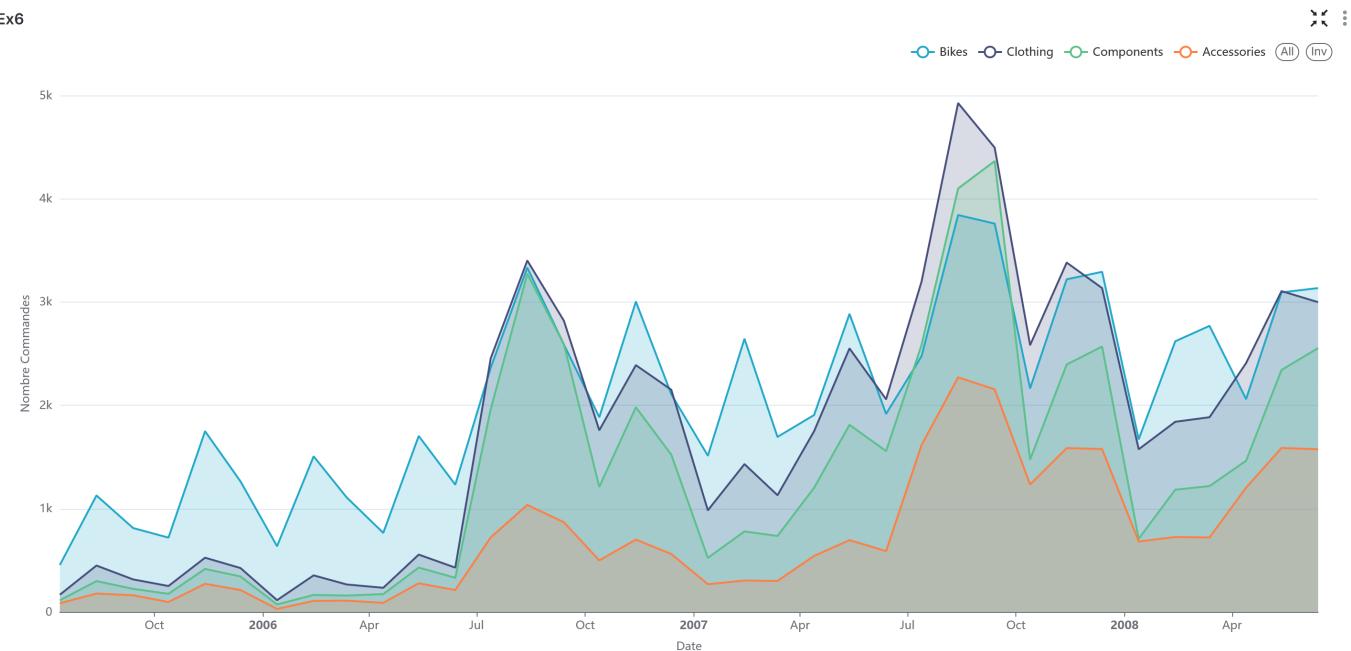
Exercise 5

```
SELECT
    dt.FullDateAlternateKey AS sale_date,
    fs.OrderQuantity
FROM factresellersales fs
JOIN dimdate dt
ON fs.DueDateKey = dt.DateKey
```



Exercise 6

```
SELECT frs.OrderQuantity, d.FullDateAlternateKey AS SaleDate,
pc.EnglishProductCategoryName AS Category
FROM factresellersales frs
JOIN dimdate d
ON frs.DueDateKey = d.DateKey
JOIN dimproduct p
ON frs.ProductKey = p.ProductKey
JOIN dimproductsubcategory psc
ON p.ProductSubcategoryKey = psc.ProductSubcategoryKey
JOIN dimproductcategory pc
ON psc.ProductCategoryKey = pc.ProductCategoryKey
```



Exercise 7

```

SELECT
    d.FullDateAlternateKey AS SaleDate,
    f.CustomerKey,
    SUM(f.SalesAmount) OVER (
        PARTITION BY f.CustomerKey
        ORDER BY d.FullDateAlternateKey
        ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW
    ) AS CumulativeSales
FROM factinternetsales f
JOIN dimdate d
    ON f.DueDateKey = d.DateKey
ORDER BY
    f.CustomerKey ASC,
    d.FullDateAlternateKey ASC;

```

| SaleDate | CustomerKey | SUM(CumulativeSales) |
|---------------------|-------------|----------------------|
| 2008-06-13 00:00:00 | 11420 | 55.8k |
| 2007-12-19 00:00:00 | 12333 | 53.9k |
| 2008-03-03 00:00:00 | 11241 | 53.5k |
| 2008-05-01 00:00:00 | 12300 | 52.8k |
| 2008-05-09 00:00:00 | 12296 | 52.6k |
| 2008-03-03 00:00:00 | 17221 | 50.7k |
| 2008-06-23 00:00:00 | 15614 | 47.7k |
| 2007-10-05 00:00:00 | 12335 | 45k |
| 2007-12-05 00:00:00 | 12321 | 43.2k |
| 2007-12-21 00:00:00 | 12631 | 43.1k |
| 2007-11-14 00:00:00 | 12307 | 43k |
| 2007-08-29 00:00:00 | 12132 | 42.7k |
| 2008-01-16 00:00:00 | 13600 | 42.1k |
| 2007-12-21 00:00:00 | 12655 | 41.9k |
| 2008-07-08 00:00:00 | 11429 | 41.8k |
| 2008-01-02 00:00:00 | 13263 | 41.8k |
| 2008-01-29 00:00:00 | 14940 | 41.7k |
| 2007-12-19 00:00:00 | 14775 | 41.7k |
| 2008-01-23 00:00:00 | 13591 | 41.6k |
| 2008-02-10 00:00:00 | 12118 | 41.6k |
| 2008-01-15 00:00:00 | 12079 | 41.6k |
| 2007-12-29 00:00:00 | 11854 | 41.5k |
| 2008-01-23 00:00:00 | 11766 | 41.2k |
| 2008-02-06 00:00:00 | 11343 | 40.9k |
| 2007-11-16 00:00:00 | 11000 | 40.9k |
| 2007-12-15 00:00:00 | 11457 | 40.8k |
| 2008-03-07 00:00:00 | 11914 | 40.6k |
| 2008-07-10 00:00:00 | 12302 | 40.2k |
| 2008-02-24 00:00:00 | 11032 | 40.2k |
| 2007-09-29 00:00:00 | 11439 | 40.2k |

Exercise 8

```

SELECT fis.OrderQuantity AS 'Order Quantity', d.FullDateAlternateKey AS 'Delivery Date',
       fis.CustomerKey AS 'Customer Id'
  FROM factinternetsales fis
 JOIN dimdate d
    ON fis.DueDateKey = d.DateKey
  
```

Ex8

| Customer Id | Metric | SUM(Order Quantity) | | | | | | | | | | | | | | Total (Sum) |
|-------------|---------------|---------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-------------|
| | Delivery Date | Jul 2007 | Aug 2007 | Sep 2007 | Oct 2007 | Nov 2007 | Dec 2007 | Jan 2008 | Feb 2008 | Mar 2008 | Apr 2008 | May 2008 | Jun 2008 | Jul 2008 | Aug 2008 | |
| 11000 | | | | 2 | | | | 5 | | | | | | | | 7 |
| 11001 | | | 6 | | | | | | | | | | | | | 10 |
| 11003 | | 4 | | | | | | 4 | | | | | | | | 8 |
| 11004 | | | 3 | | | | | | | | | | | | | 3 |
| 11005 | | | | 4 | | | | | | | | | | | | 4 |
| 11006 | | 3 | | | | | | | | | | | | | | 3 |
| 11007 | | | 5 | | | | | | | | | | | | | 5 |
| 11008 | | 4 | | | | | | | | | | | | | | 4 |
| 11009 | | | 2 | | | | | | | | | | | | | 2 |
| 11012 | | | | 3 | | | | | | | | | | | | 3 |
| 11013 | | | | | | | | 4 | | | | | | | | 3 |
| 11014 | | | | | | | | | | | | | | | | 4 |
| 11015 | | | 3 | | | | | | | | | | | | | 3 |
| 11016 | | | | 3 | | | | | | | | | | | | 3 |
| 11017 | | 2 | | | | | | | | | | | | | | 2 |
| 11018 | | | 2 | | | | | | | | | | | | | 6 |
| 11019 | | | 3 | 4 | | | | 3 | 3 | | | | | 4 | 6 | 27 |
| 11020 | | 2 | | | | | | | | | | | | | | 2 |
| 11021 | | | 3 | | | | | | | | | | | | | 3 |
| 11022 | | | 2 | | | | | | | | | | | | | 2 |
| 11023 | | | | 4 | | | | | | | | | | | | 4 |
| 11024 | | | | | | | | | 3 | 3 | | | | | | 6 |
| 11025 | | | 3 | | | | | | | | | | | | | 3 |
| 11026 | | | | | 3 | | | | | | | | | | | 6 |
| 11027 | | | | | | | 4 | | | | | | | | | 8 |

And another screenshot that shows values > 7.

| R88UT | 3 | | | | | | | | | | | | | | | | / |
|-------|---|---|---|---|---|---|---|---|----|---|--|---|---|--|--|----|---|
| 11089 | 3 | | | | | | | | | | | | | | | 7 | |
| 11090 | | 3 | | | | | | | | | | | | | | 3 | |
| 11091 | | | 3 | 5 | 8 | 5 | 4 | 3 | 10 | 4 | | 6 | 7 | | | 55 | |
| 11092 | 2 | | | | | | | | | | | | | | | 2 | |
| 11093 | 4 | | | | | | | | | | | | | | | 4 | |

Exercise 9

- Give the main advantage of performing a visualization on the Web compared to a document (printed report for example). Then describe at least two examples of this advantage. A Web visualization is dynamic and allows the user to manipulate the data, unlike a printed or static report.
- Example 1: Users can zoom, filter, highlight data points, change time ranges, and navigate through dimensions that cannot be shown in a fixed document.
- Example 2: A Web dashboard can refresh automatically using APIs or data streams. A printed document cannot reflect continuous or live changes.
- Explain what the Shannon communication model (1948) has indirectly brought to the field of data visualization

The Shannon-Weaver model is one of the foundational theories of communication. It describes how information is transmitted from a sender to a receiver through a channel, with potential noise that can distort the message.

It emphasizes the need to minimize noise (misinterpretation or confusion) and ensure that the intended message is accurately conveyed to the audience. This has led to the development of best practices in data visualization, such as choosing appropriate chart types, using clear labels, and designing for accessibility, all aimed at enhancing the clarity and effectiveness of visual communication.

- Noise reduction. The model highlights the need to eliminate unnecessary elements. In visualization, removing visual clutter (chartjunk, redundant markers) increases the "capacity" of the graphic to convey information clearly.

- Efficient encoding of information. As Shannon focuses on optimal encoding for maximum transmission, visualization must choose perceptually efficient encodings (position - length - angle - color) to minimize information loss.
- Receiver-centric design. The model stresses correct decoding by the receiver. In data visualization, this translates to designing clear, interpretable graphics aligned with human perception.

Source: https://en.wikipedia.org/wiki/Shannon%20Weaver_model

3. A scale has been applied to the time series data of the first diagram in Figure 4, resulting in the display of the second diagram. Explain what this scale is and describe what information it gives compared to the first diagram.

The second diagram results from applying a logarithmic scale to the values of the original time series.

A logarithmic transformation (typically $\log(x)$ or $\log_{10}(x)$) is applied to the y-axis. Instead of representing raw values, the graph displays their logarithm.

Information gained compared to the first diagram

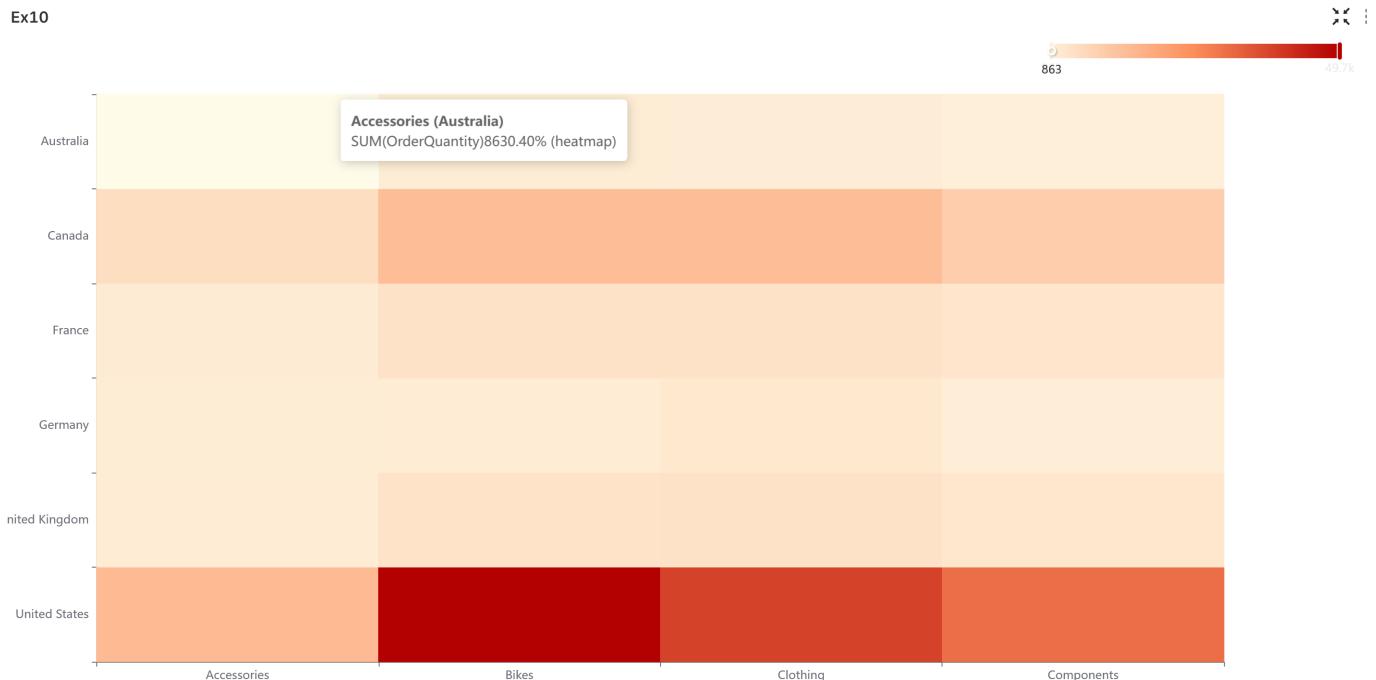
- On a linear scale, very large values dominate the chart and compress the smaller ones against the baseline. A logarithmic scale redistributes the space so both small and large values become interpretable. This prevents the graph from being visually distorted when data vary by factors of hundreds or thousands.
- Exponential or multiplicative processes form straight lines on a log scale. This transformation exposes patterns, consistency, or deviations that remain hidden on a linear axis, where the curve rapidly increases and becomes unreadable. It allows analysts to detect structure and assess growth behaviour more accurately.

Source: <https://www.forbes.com/sites/naomirobbins/2012/01/19/when-should-i-use-logarithmic-scales-in-my-charts-and-graphs/>

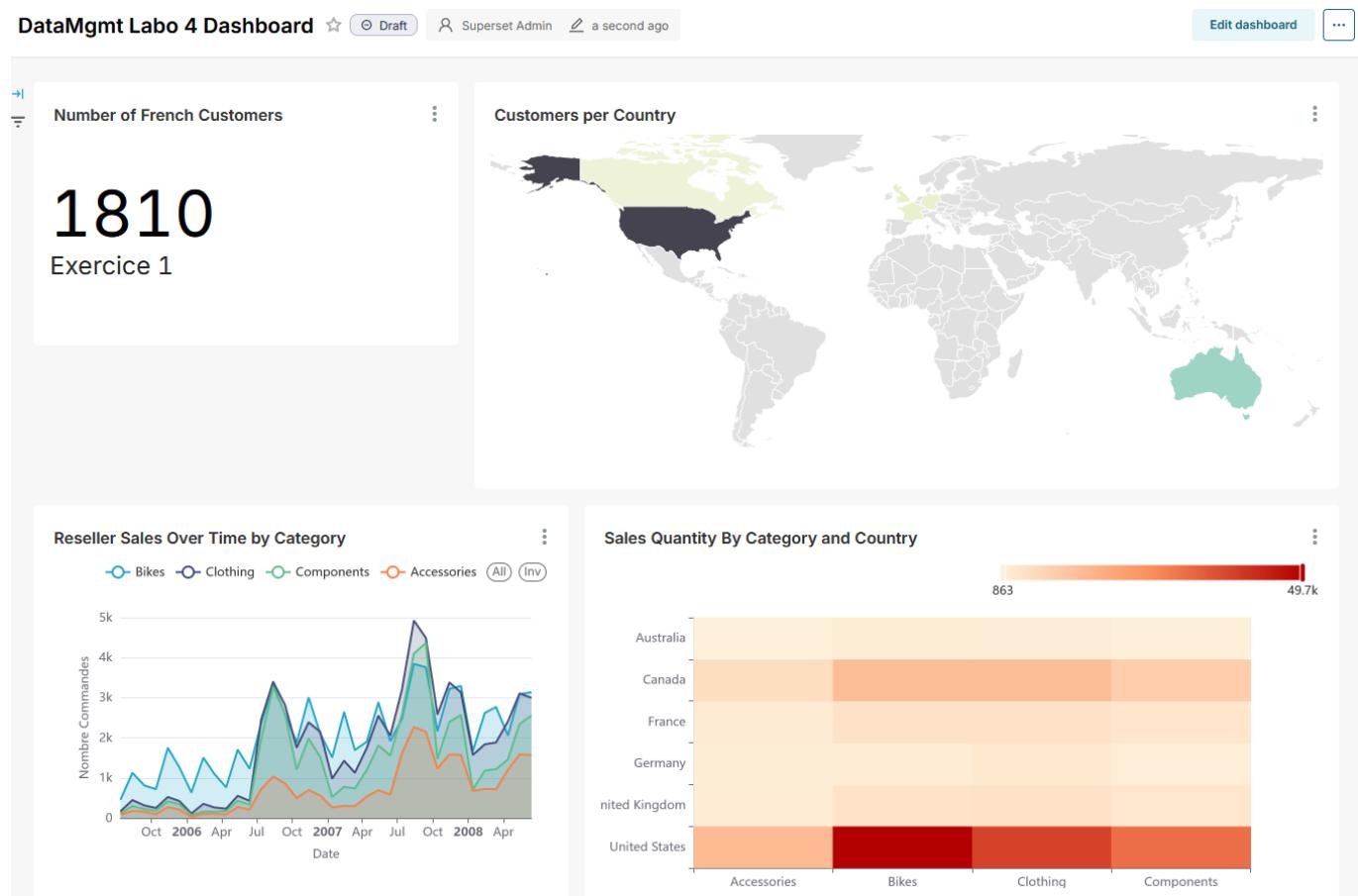
Exercise 10

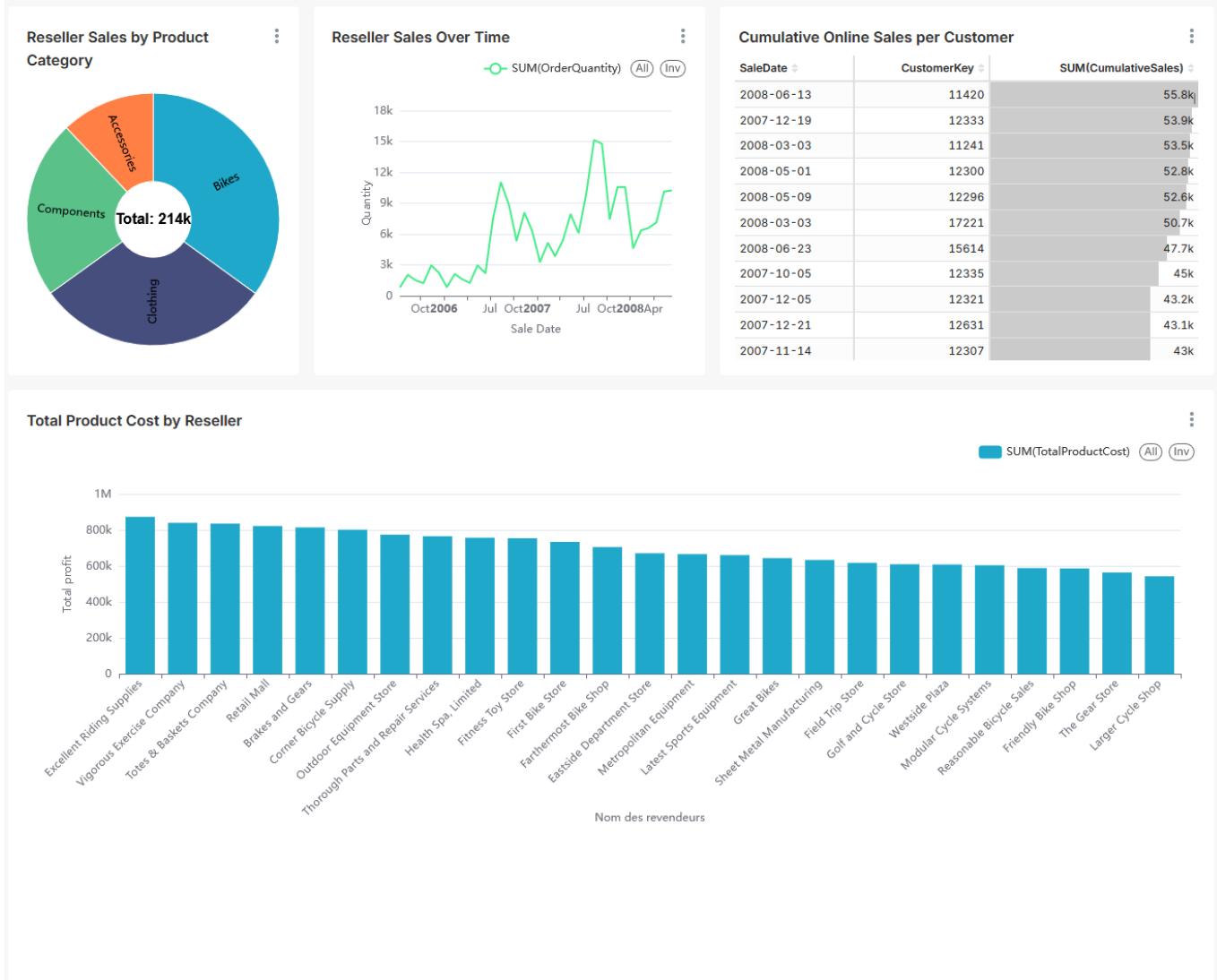
Let's create a heatmap that shows the number of sales per categories by country.

```
SELECT pc.EnglishProductCategoryName AS Category, g.EnglishCountryRegionName AS Country, frs.OrderQuantity
FROM factresellersales frs
JOIN dimproduct p
ON frs.ProductKey = p.ProductKey
JOIN dimproductsubcategory psc
ON p.ProductSubcategoryKey = psc.ProductSubcategoryKey
JOIN dimproductcategory pc
ON psc.ProductCategoryKey = pc.ProductCategoryKey
JOIN dimreseller dr
ON dr.ResellerKey = frs.ResellerKey
JOIN dimgeography g
ON dr.GeographyKey = g.GeographyKey
```



Here is a final display with a practical layout.





Order Quantities by Customer and Delivery Month

| Customer Id | Metric | SUM(Order Quantity) | | | | | | | | | | | | Total (Sum) | |
|-------------|--------|---------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-------------|----|
| | | Delivery Date | Jul 2007 | Aug 2007 | Sep 2007 | Oct 2007 | Nov 2007 | Dec 2007 | Jan 2008 | Feb 2008 | Mar 2008 | Apr 2008 | May 2008 | Jun 2008 | |
| 11000 | | | | 2 | | | | 5 | | | | | | | 7 |
| 11001 | | | | | 6 | | | | | | | | | 4 | 10 |
| 11003 | | | 4 | | | | | 4 | | | | | | | 8 |
| 11004 | | | | 3 | | | | | | | | | | | 3 |
| 11005 | | | | 4 | | | | | | | | | | | 4 |
| 11006 | | | 3 | | | | | | | | | | | | 3 |
| 11007 | | | | | 5 | | | | | | | | | | 5 |
| 11008 | | | | 4 | | | | | | | | | | | 4 |