E-Commerce Retail Data Analysis: Uncovering Revenue Patterns and Customer Behaviour

By Filzah Annisa

Agenda

01 Business Overview

02 Objective & Goals

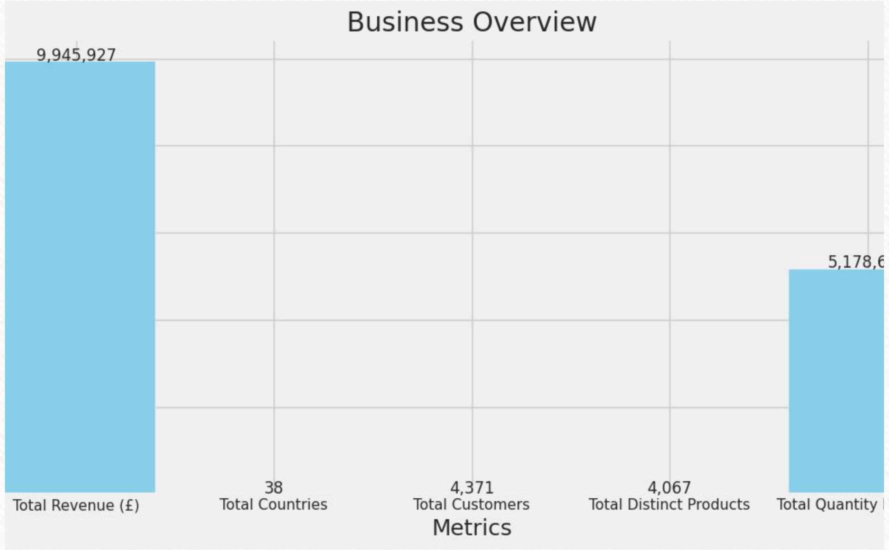
03 Analysis

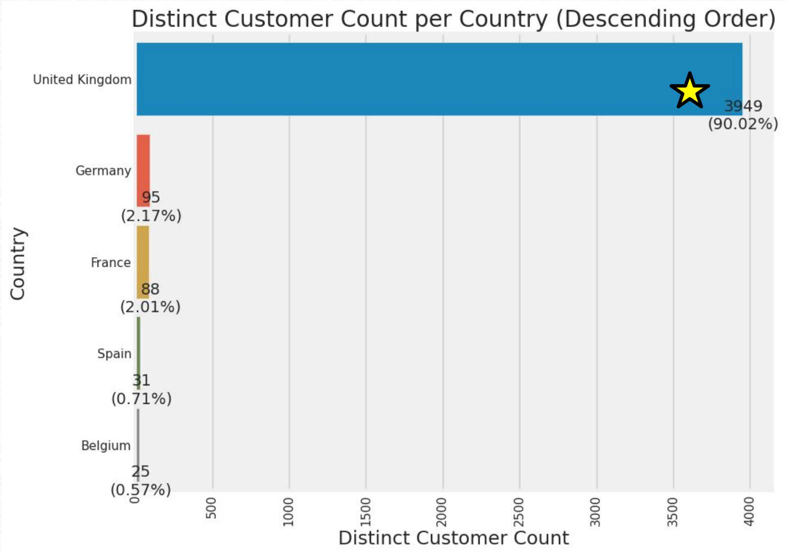
04 What's Next?

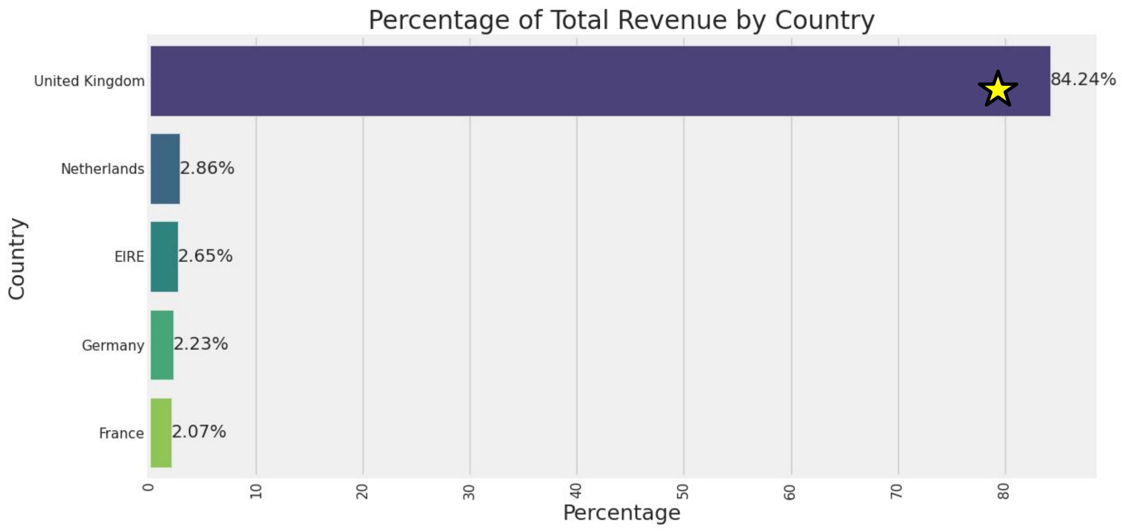
01

Business Overview – High Level

- The top 5 countries generate:
  - 94.05% of total customers
  - 95. 48% of total revenue





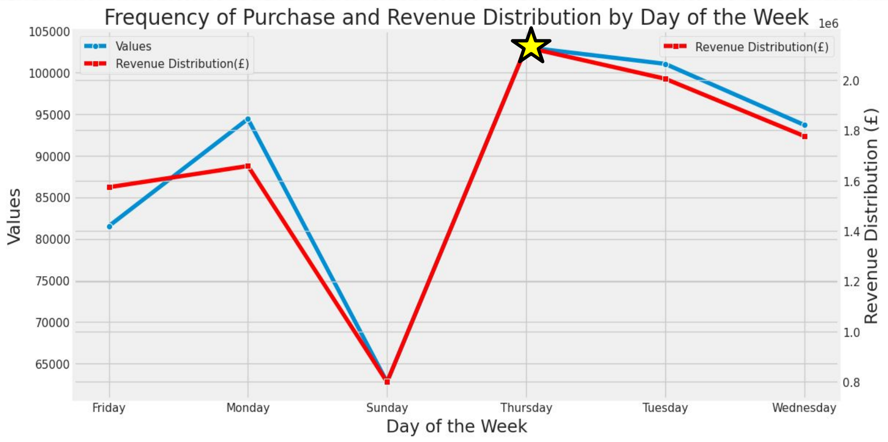


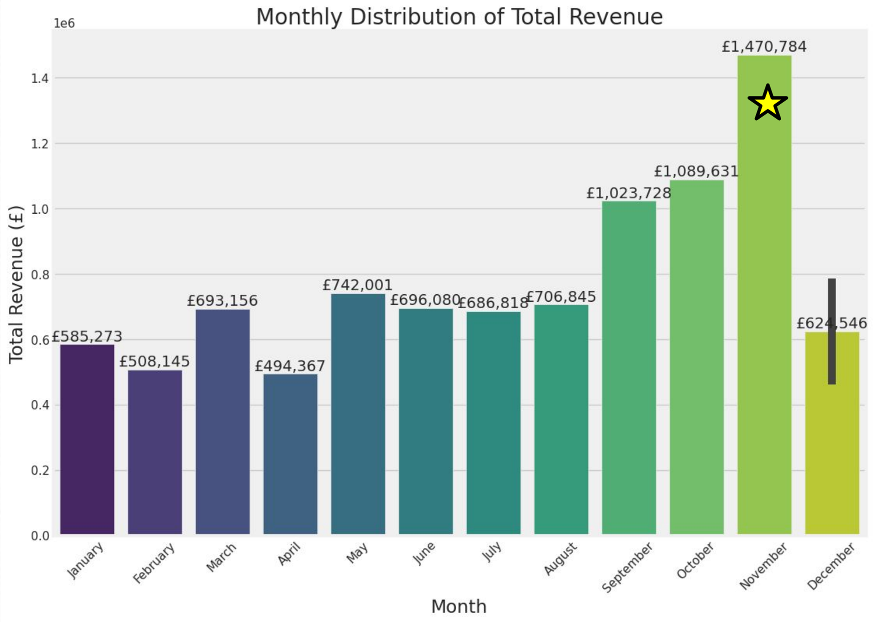
☆ : Best

01

Business Overview – Patterns

- **Thursday** is the TOP purchasing day
  - Followed by Tuesday
- Saturday seems to be holiday for the store, no revenue generated
- **November** is the best month for sales





☆ : Best

01

Business Overview – Product Level

Which ones are Best sellers?

Which ones are best for our business?

Best Seller Products	Quantity purchased
SMALL POPCORN HOLDER	56.427
WORLD WAR 2 GLIDERS ASSTD DESIGNS	53.751
JUMBO BAG RED RETROSPOT	47260
mailout	44024
WHITE HANGING HEART T-LIGHT HOLDER	39103

BEST SALES PRODUCTS	TotalRevenue
DOTCOM POSTAGE	£ 206.245
REGENCY CAKESTAND 3 TIER	£ 164.459
WHITE HANGING HEART T-LIGHT HOLDER	£ 99.790
PARTY BUNTING	£ 98.243
JUMBO BAG RED RETROSPOT	£ 92.175

01 Business Overview – Returns

- A lot of products (241) were returned manually so we couldn't find out the exact product name.
- Some are Unknown
- All samples were returned. In the future we do not need to sent out samples anymore.
- 5.52% products are returned.

- It seems that customers with more returns = more revenue
- But more quantity purchased =/= more or less returns

Product Name	Returned Quantity
Manual	241
REGENCY CAKESTAND 3 TIER	183
Unknown	97
JAM MAKING SET WITH JARS	87
SET OF 3 CAKE TINS PANTRY DESIGN	75
STRAWBERRY CERAMIC TRINKET BOX	60
SAMPLES	60

```
Correlation between Total Revenue and Returns: 0.95
```

```
from scipy.stats import pointbiserialr
correlation_quantity_returns, _ = pointbiserialr(data['Return'], data['Quantity'])
print(f"Correlation between Quantity and Returns: {correlation_quantity_returns:.2f}")
```

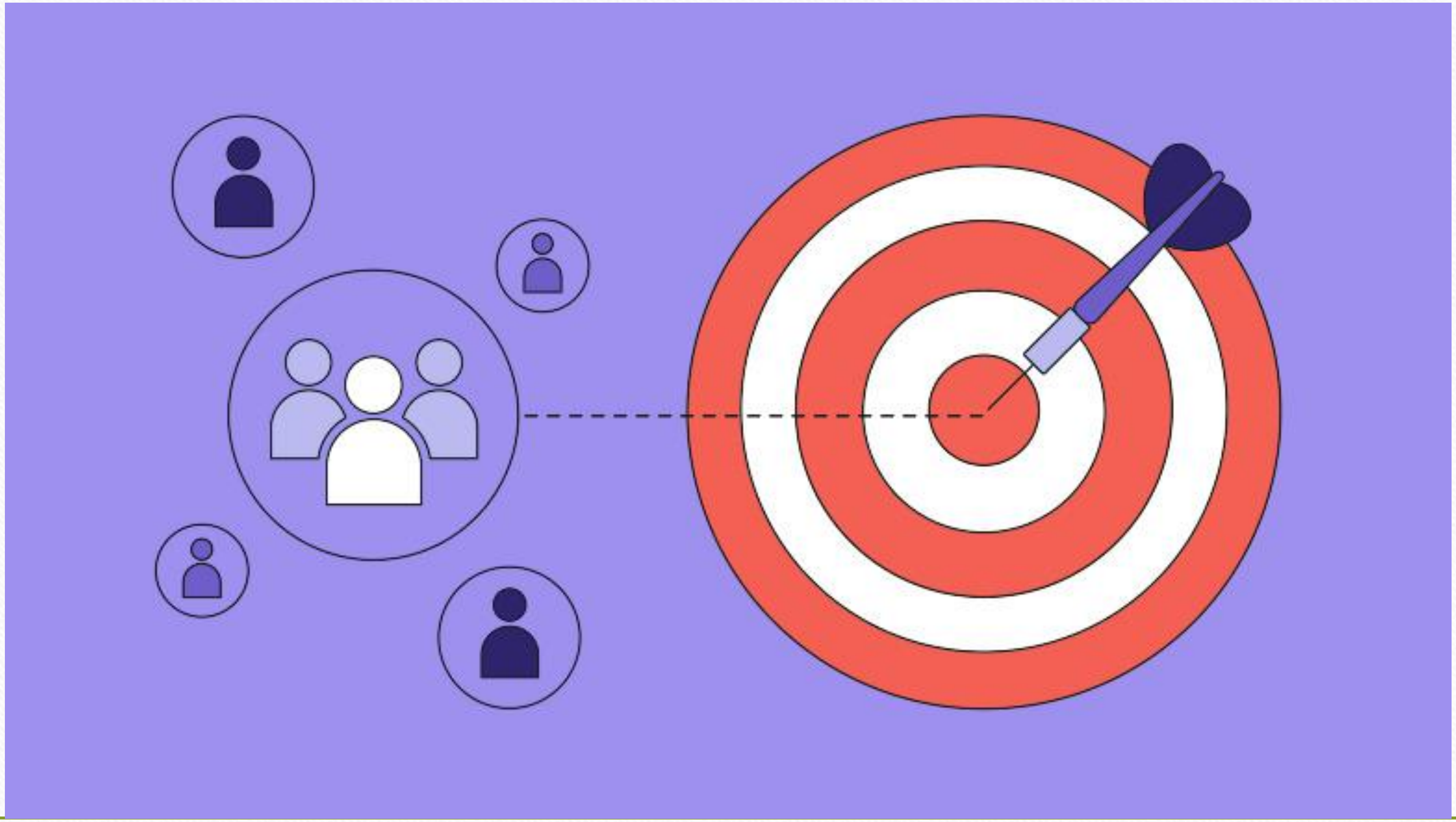
```
Correlation between Quantity and Returns: -0.12
```

02

Objective

What do we want to achieve?

02 HOW?



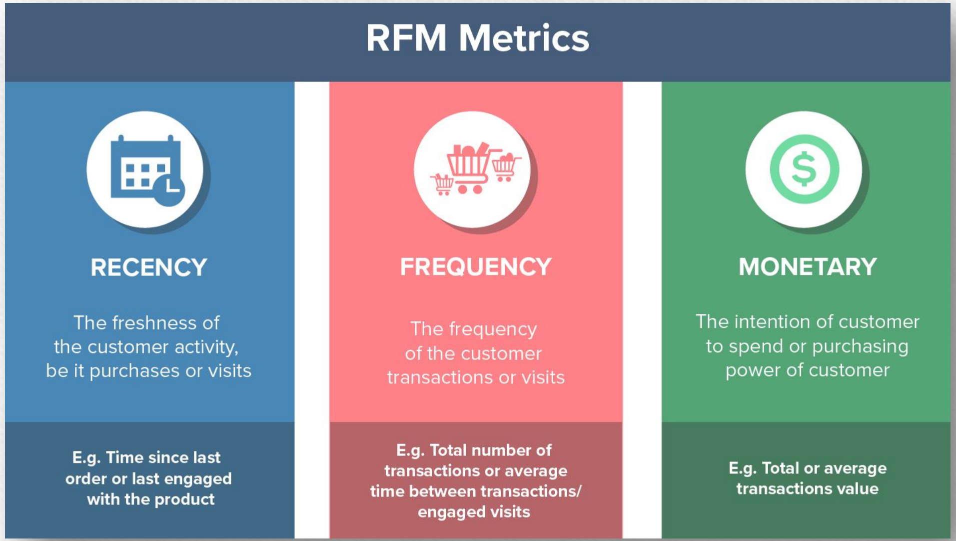
03

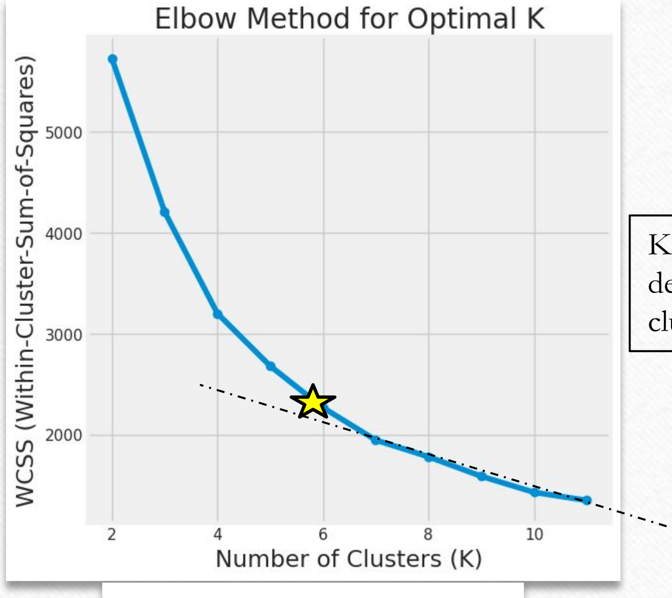
Analysis

The Right Target

The Right Products

03 The Right Target: Customer Segmentation with RFM





K-Means to decide # of clusters

	R_Score	F_Score	M_Score
0	3.625937	2.951274	3.597451
1	2.456808	1.000000	1.443631
2	1.672932	2.953008	3.550752
3	3.371667	2.355000	2.083333
4	1.616698	2.220114	2.159393
5	1.000000	1.043165	1.244604

My RFM scores range from 1-4

- 1: HIGHEST/BEST
- 4:LOWEST / WORST

☆ : Best

03 The Right Target: Customer Segmentation with RFM

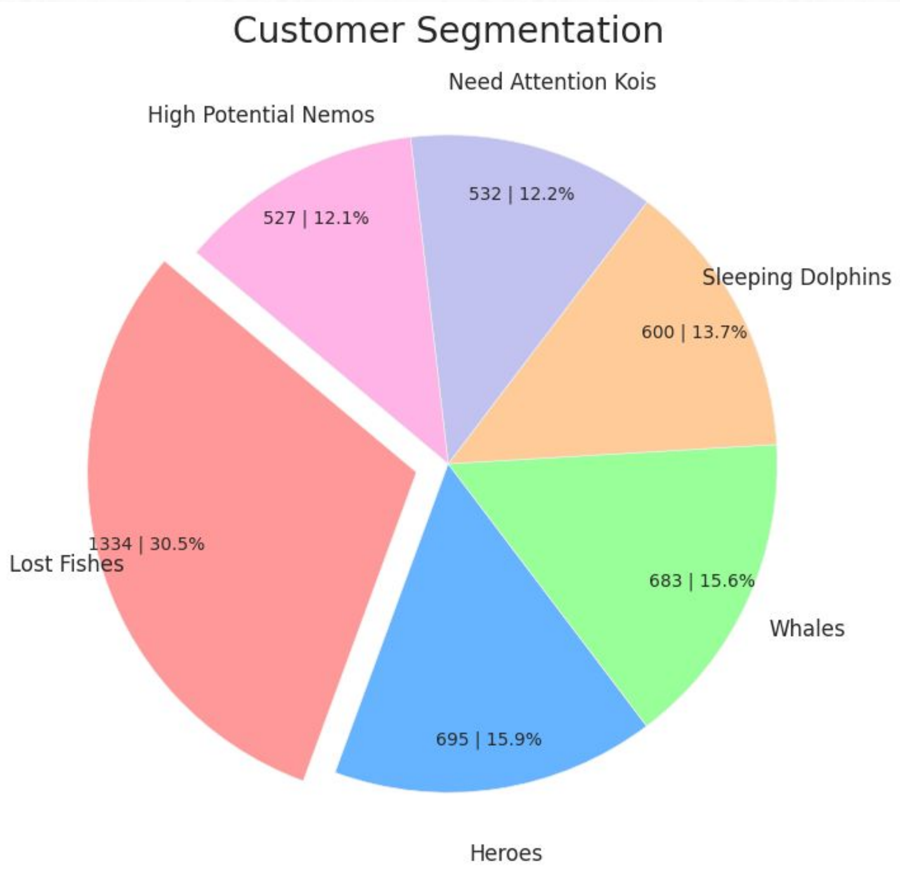
Code snippet:

```
# RFM - KMEANS ANALYSIS RESULT IN CLUSTERING

#def segment_customers:
cluster_names = {
    0: 'Lost Fishes', #low RFM
    1: 'Whales', #high M, older cust
    2: 'Need Attention Kois', #new customers, curious but not buying
    3: 'Sleeping Dolphins', #not buying anytime recent, but brings average value/money
    4: 'High Potential Nemos', #new customers, already buying
    5: 'Heroes' # HIGH RFM
}


rfm_df['Cluster_Name'] = rfm_df['Cluster'].map(cluster_names)
```

```
sample_purchases = data[data['Description'] == 'SAMPLES']


cluster_names_for_samples = sample_purchases['Cluster_Name'].unique()

print(cluster_names_for_samples)
```

```
['Heroes']
```

- I noticed only 'Heroes' customers receive samples



Cluster #	Customer Segmentation	Definition
0	Lost Fishes	customers who are not buying from the business at all now
1	Whales	Customers that bring high volume of revenue, but not buying that recently
2	Need Attention Kois	NEW customers who are already actively buying and brings revenue into our business
3	Sleeping Dolphins	Customers who have bought considerable amount, but not anymore
4	High Potential Nemos	NEW customers who are not buying
5	Heroes ☆	Customers who actively brings revenue into our business and purchase frequently

03 The Right Product: Recommender systems using Collaborative Filtering

Code Snippet:

```python
def generate_recommendation(model, user_id, ratings_df, n_items):
    product_ids = ratings_df["StockCode"].unique()
    product_ids_user = ratings_df.loc[ratings_df["CustomerID"] == user_id, "StockCode"]
    product_ids_to_pred = np.setdiff1d(product_ids, product_ids_user)
    test_set = [[user_id, product_id, 4] for product_id in product_ids_to_pred]

    # Predict the ratings and generate recommendations
    predictions = model.test(test_set)
    pred_ratings = np.array([pred.est for pred in predictions])
    print("Top {0} product recommendations for user {1}:".format(n_items, user_id))

    # Rank top-n products based on the predicted ratings
    index_max = (-pred_ratings).argsort()[:n_items]
    for i in index_max:
        product_id = product_ids_to_pred[i]
        product_name = ratings_df[ratings_df["StockCode"] == product_id]["Description"].values
        print(product_name)
```

```
MAE:  0.0157
RMSE: 0.0261
Mean Absolute Error (MAE): 0.015656839543934712
Root Mean Squared Error (RMSE): 0.02605629211849689
MAE: 0.015656839543934712
RMSE: 0.02605629211849689
Precision: 1.0
Recall: 0.998652896273013
F1-Score: 0.999325994158616
MAP: 1.0
NDCG: 0.9999461508473664
```

Case Example:

```python
# Generate recommendations for user with 5 items recommendation (i chose this)

user_id = 17850
n_items = 5
generate_recommendation(svd, user_id, data_with_ratings, n_items)
```

```
Top 5 product recommendations for user 17850:
SMALL POPCORN HOLDER
WORLD WAR 2 GLIDERS ASSTD DESIGNS
JUMBO BAG RED RETROSPOT
ASSORTED COLOUR BIRD ORNAMENT
PACK OF 72 RETROSPOT CAKE CASES
```

Results:

The recommendation system is **performing exceptionally well and** is **effective** in providing relevant product **recommendations** to users.

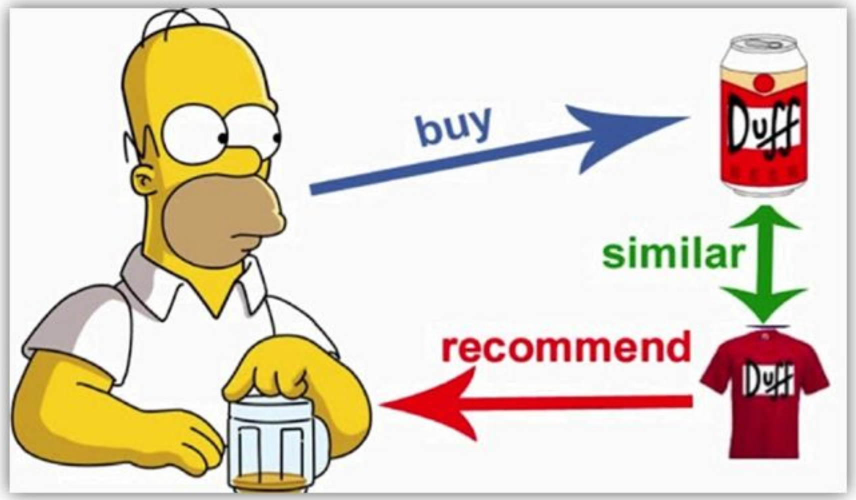
03 The Right Product: Recommender systems using Collaborative Filtering

- *Amazon has reported that the business of cross-selling and upselling make up as much as **35% of its revenue**.
- **Product recommendations** are responsible for an average of 10-30% of eCommerce site revenues.

- Lowballing this data. If we use system recommendation and the right marketing strategies, we could:

- **Increase revenue up to 45%**
- **Increase of up to £ 4.475.667**

```
cross_selling_percentage = 0.35

average_cross_selling_revenue = cross_selling_percentage * total_revenue
average_cross_selling_revenue

3481074.5704

[176] product_recommendations_percentage = 0.10
average_recommendations_revenue = product_recommendations_percentage * total_revenue

expected_total_revenue = total_revenue + average_cross_selling_revenue + average_recommendations_revenue
expected_total_revenue

14421594.6488
```



*https://vwo.com/blog/use-upsell-cross-sell/

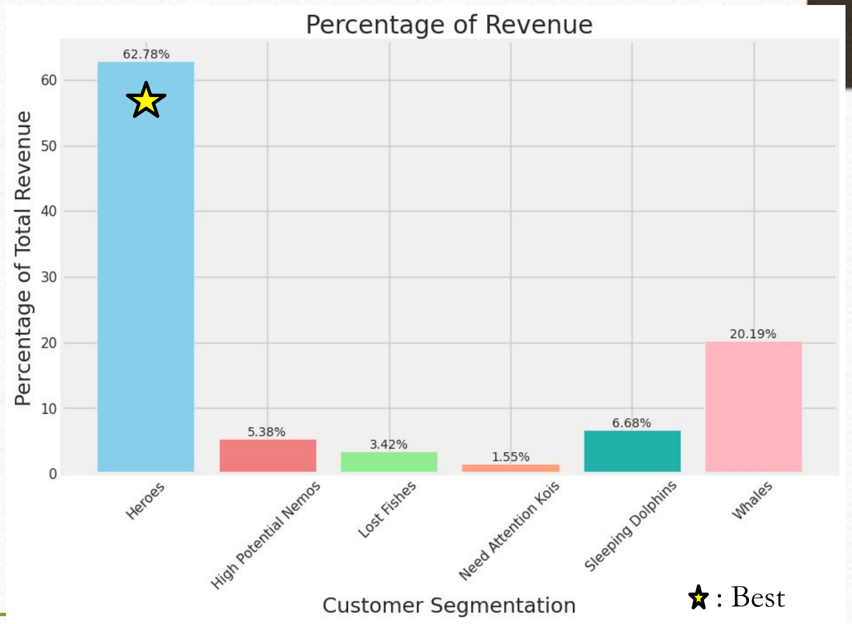
04 What's next?

04 Summary & Action Recommendation

- The business had a problem retaining and taking care of their current older customers.
  - 30% of lost customers have to be treated like a new customers IF we would like to re-activate them. Since they likely to have forgotten about our business.
- Being a UK-based business. It makes sense for UK to be the biggest contributor in terms of customer and sales
- No more samples for customers. All samples sent out were returned = not the right approach
  - Saved money: £ 3.039
- Based on R-F-M Scoring, there are 6 segments of customer. (1 being lost):

From most important to least

1. **Heroes**
2. **Whales**
3. Sleeping Dolphins
4. High Potential Nemos
5. Need Attention Kois
6. ~~Lost Fishes~~



04 Growth Strategies per Customer Segment: High Prio

Customer Segment	Priority	Customer Traits	Action Recommendation	Channels
Heroes	HIGHEST	Loyal, Frequent Buyer, Big Spender	**TOP priority** in customer service: **Special** membership, **loyalty** programmes and perks, cross-sell in webshop carts, webshop pages or (post-purchase) via e-mail marketing, exclusive products introduction, product newsletter	Sales & Marketing
Whales	HIGH	Frequent Buyer, Big Spender	Focus on making them into HEROES: **cross-sell** via e-mail marketing & webshop marketing, new products introduction, **newsletter** about interesting products, entice them into **joining loyalty programmes**	Sales & Marketing

04 Growth Strategies per Customer Segment: The Rest

Customer Segment	Priority	Customer Traits	Action Recommendation	Channels
Sleeping Dolphins	MID	May Forget Us, Average Spending	**Mass Direct Marketing**: introduce interesting products via email, with survey to understand why they are sleeping, **offer** discounts or **promotions** (1+1) on relevant products, newsletter email marketing.	Omni-channel Marketing
High Potential Nemos	MID	Newcomers, Curious, Whales Potential	**Customer support** to introduce products, onboard webshop, **newsletter** for products+cross-selling marketing, **special promotion** for a short period of time to make them interested to become loyal.	Omni-channel Marketing
Need Attention Kois	LOW	Newcomers, could be Lost fishes if not taken cared of.	**newsletter** for products + cross-selling marketing, emails about special **promotion** for a short period of time to make them interested to make a purchase.	E-mail marketing
~~Lost Fishes~~	-	No sales, no visits	**Minimum to no effort**: email newsletter. We can also consider this as none customers. If they start making a purchase again, treat them as 'need attention kois'.	E-mail marketing

Further analysis on webshop &omni-channel marketing interaction is needed in order to decide which channel, content, and funnel is the best way for customers.

REFERENCE & PYTHON CODING

Data used:

https://www.kaggle.com/datasets/atharvaarya25/e-commerce-analysis-uk

Python coding:

https://colab.research.google.com/drive/1vPfYMNCy1jHVTcA0XfM7zHP9J1T72piz#scrollTo=zX_cRm4Q5-YS

Thank you!