CUSTOMER PERSONALITY ANALYSIS

By Ian Hash and Pleng Witayaweerasak

Executive Summary

PROBLEM DOMAIN	We aim to explore how supermarket businesses can perform customer segmentation analysis using clustering techniques to optimize marketing strategy, identify high value customers, and increase revenue
DATASET	Our dataset is sourced from Kaggle – it offers a comprehensive view of customer demographics, purchasing behavior, marketing campaign responses, and interaction channels
METHODS	We will be using dimensionality reduction with PCA, unsupervised learning and clustering techniques ie. K-Means and Apriori Algorithm to find insights about different customer segments
FINDINGS	From K-Means analysis, we segmented customers into 4 clusters based on their income, spending, and shopping behavior. From Apriori Algorithm, we found products frequently bought together for each segment
RECOMMENDATION	We suggest product recommendation that market and cross-sell frequently bought together items, and recommend marketing and inventory strategies that will improve KPIs for the business

PROBLEM DOMAIN

- Retailers often fail to deliver personalized experiences, leading to irrelevant ads, poor product recommendations, and lost sales opportunities.
- We aim to perform customer segmentation and build a machine learning model to predict customer groups based on behavior and demographics, and suggest relevant products that will increase sales.

BUSINESS QUESTIONS & KPIS

- How can segmentation and machine learning drive personalized marketing to boost sales and ROI?
- How can retailers allocate resources more efficiently by identifying and targeting high-value customer segments?
- How can customer insights inform product development and retention strategies to increase revenue and customer lifetime value?

OUR DATASET

Source

Customer Personality Analysis

2,240 rows x 29 features

Features include demographics, shopping behaviors, total spent on products, and customer responses

24 missing values

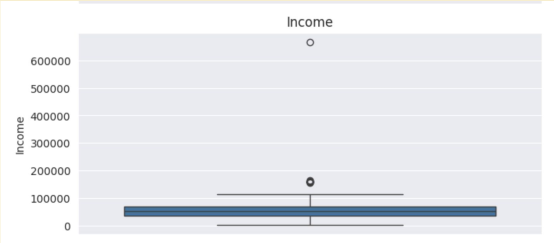
Dataset is already pretty clean

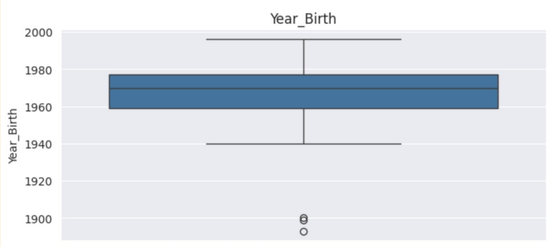
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2240 entries, 0 to 2239
Data columns (total 29 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   ID                   2240 non-null   int64
 1   Year_Birth           2240 non-null   int64
 2   Education            2240 non-null   object
 3   Marital_Status       2240 non-null   object
 4   Income               2216 non-null   float64
 5   Kidhome              2240 non-null   int64
 6   Teenhome             2240 non-null   int64
 7   Dt_Customer          2240 non-null   object
 8   Recency              2240 non-null   int64
 9   MntWines             2240 non-null   int64
 10  MntFruits            2240 non-null   int64
 11  MntMeatProducts      2240 non-null   int64
 12  MntFishProducts      2240 non-null   int64
 13  MntSweetProducts     2240 non-null   int64
 14  MntGoldProds         2240 non-null   int64
 15  NumDealsPurchases    2240 non-null   int64
 16  NumWebPurchases      2240 non-null   int64
 17  NumCatalogPurchases  2240 non-null   int64
 18  NumStorePurchases    2240 non-null   int64
 19  NumWebVisitsMonth    2240 non-null   int64
 20  AcceptedCmp3         2240 non-null   int64
 21  AcceptedCmp4         2240 non-null   int64
 22  AcceptedCmp5         2240 non-null   int64
 23  AcceptedCmp1         2240 non-null   int64
 24  AcceptedCmp2         2240 non-null   int64
 25  Complain             2240 non-null   int64
 26  Z_CostContact        2240 non-null   int64
 27  Z_Revenue            2240 non-null   int64
 28  Response             2240 non-null   int64
dtypes: float64(1), int64(25), object(3)
memory usage: 507.6+ KB
```

DATA PREPROCESSING + FEATURE ENGINEERING

- We imputed 24 missing income values with median income
- Age and Income have outliers so we set a cap max (Age < 90 and income < 600K) and dropped 4 rows
- We created new features such as Total Spending, Age, Family Size, Recency, and some interaction features
- Categorical features were grouped for clarity – Education was grouped into undergraduate, graduate, and postgraduate. Marital status was grouped into Living_With partner or alone
- Some irrelevant features were also dropped such as Customer_ID
- Categorical features were one-hot encoded and the rest of numerical features were scaled using StandardScaler






```
--- Education ---
Education
Graduation    1127
PhD            486
Master         370
2n Cycle       203
Basic           54
Name: count, dtype: int64

--- Marital_Status ---
Marital_Status
Married     864
Together    580
Single      480
Divorced    232
Widow        77
Alone         3
Absurd        2
YOLO          2
Name: count, dtype: int64
```

OUR METHODS

DIMENSIONALITY REDUCTION

We used PCA to select features that explained 80% of the variance

CHOOSE N CLUSTERS

We used the Elbow Method and Silhouette Score to find n clusters

K-MEANS

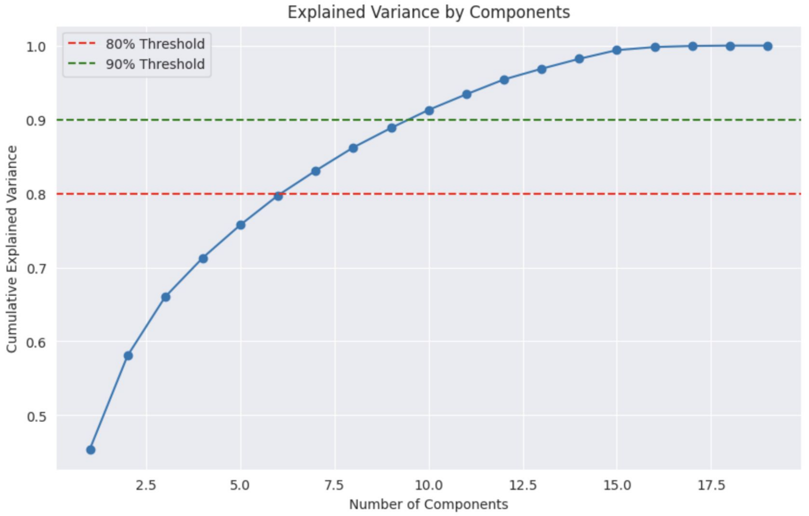
We used K-Means clustering to segment customers into 4 groups

APRIORI ALGORITHM

We used Apriori algorithm to identify product association rule

DIMENSIONALITY REDUCTION WITH PCA

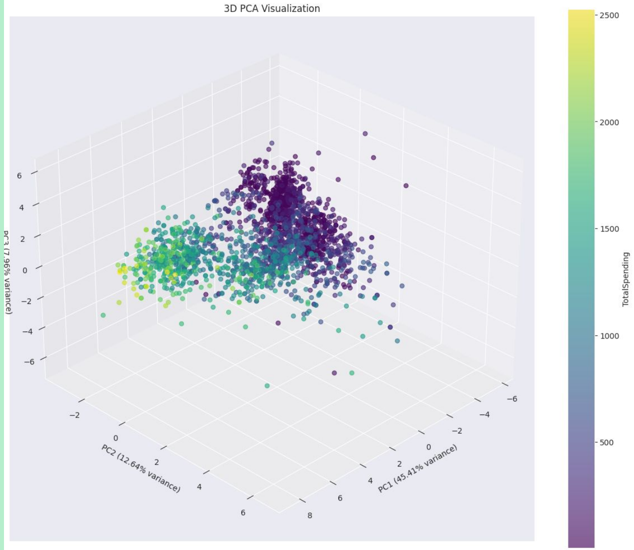
With cumulative 80% explained variance threshold, we chose 7 components for our models



Number of components explaining 80% variance: 7

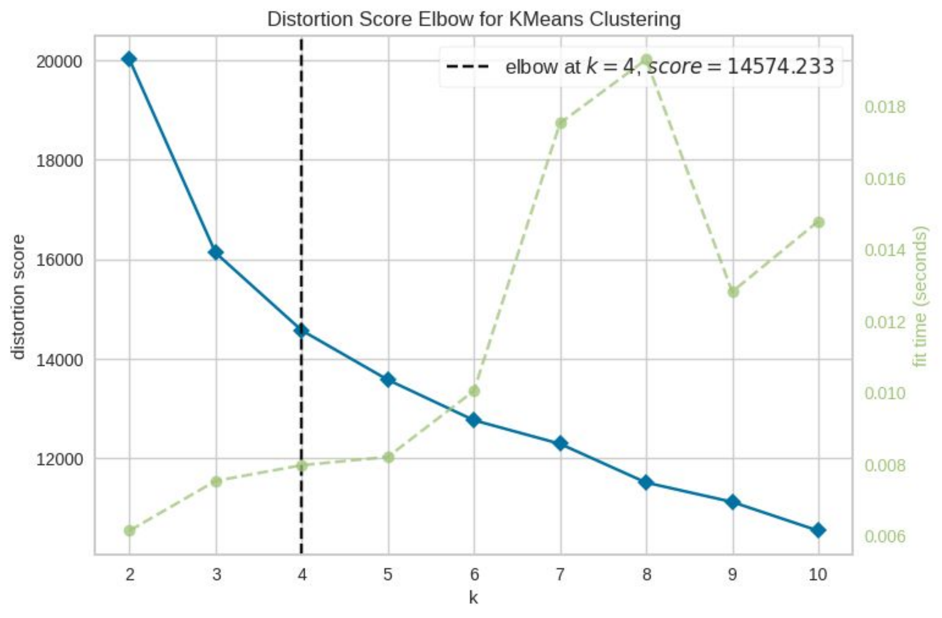
	PC1	PC2	PC3	PC4
TotalSpending	0.326086	0.055583	0.069421	0.076208
Spending_Age_Interaction	0.315062	0.092384	0.020131	0.221933
Income_FamilySize_Ratio	0.296009	0.206710	0.086136	0.111021
Income	0.292624	0.098694	0.271070	0.027492
MntMeatProducts	0.282401	0.082303	0.008408	0.010237
NumCatalogPurchases	0.278912	0.022253	0.004640	0.053540
MntWines	0.269080	0.169720	0.053228	0.289796
NumStorePurchases	0.246851	0.149286	0.004252	0.011820
MntFishProducts	0.242923	0.085016	0.079835	0.294719
MntFruits	0.233131	0.075360	0.075566	0.358437
MntSweetProducts	0.232245	0.057628	0.072423	0.352222
NumWebVisitsMonth	0.217156	0.201946	0.405028	0.109689
FamilySize_est	0.196050	0.417078	0.216578	0.238622
MntGoldProds	0.188755	0.094541	0.223057	0.174699
NumWebPurchases	0.179472	0.307604	0.199664	0.035293
Income_FamilySize_Interaction	0.056238	0.493436	0.389325	0.239700
Age	0.054716	0.214665	0.289025	0.555416
NumDealsPurchases	0.046890	0.469281	0.186581	0.134776
Recency	0.034429	0.182317	0.568427	0.143257

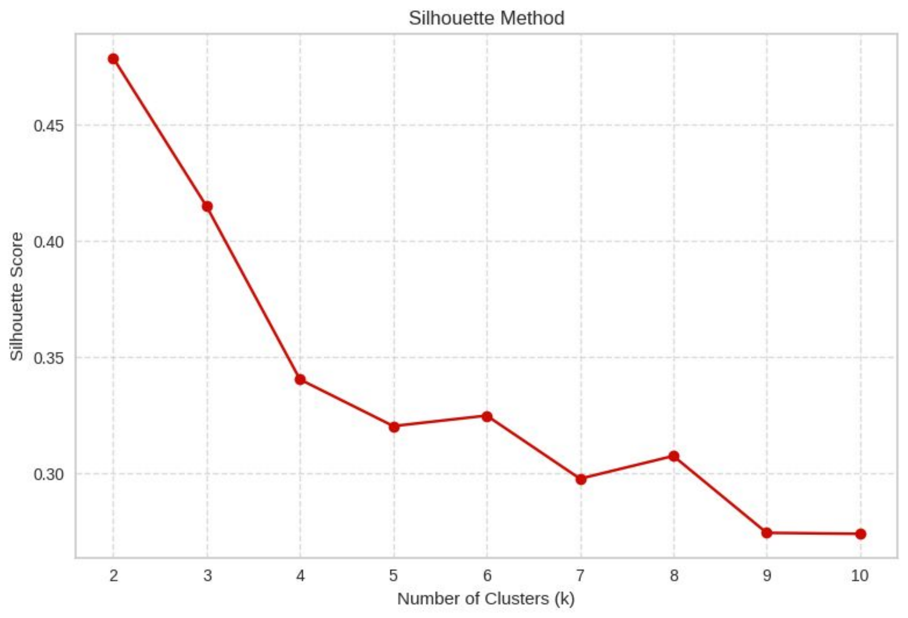
	PC5	PC6	PC7
TotalSpending	0.136813	0.025345	0.023067
Spending_Age_Interaction	0.033027	0.053789	0.032451
Income_FamilySize_Ratio	0.110310	0.027119	0.022863
Income	0.162348	0.035519	0.065183
MntMeatProducts	0.207013	0.273350	0.216573
NumCatalogPurchases	0.138063	0.128070	0.362787
MntWines	0.264049	0.132367	0.132562
NumStorePurchases	0.055408	0.220246	0.299405
MntFishProducts	0.285494	0.162961	0.001811
MntFruits	0.254757	0.127992	0.175432
MntSweetProducts	0.247159	0.104205	0.294021
NumWebVisitsMonth	0.009765	0.026512	0.137844
FamilySize_est	0.007385	0.183895	0.017744
MntGoldProds	0.332786	0.387582	0.553866
NumWebPurchases	0.066108	0.500873	0.289414
Income_FamilySize_Interaction	0.119587	0.124952	0.095368
Age	0.679074	0.171050	0.017857
NumDealsPurchases	0.093301	0.045152	0.385991
Recency	0.007012	0.548631	0.149780



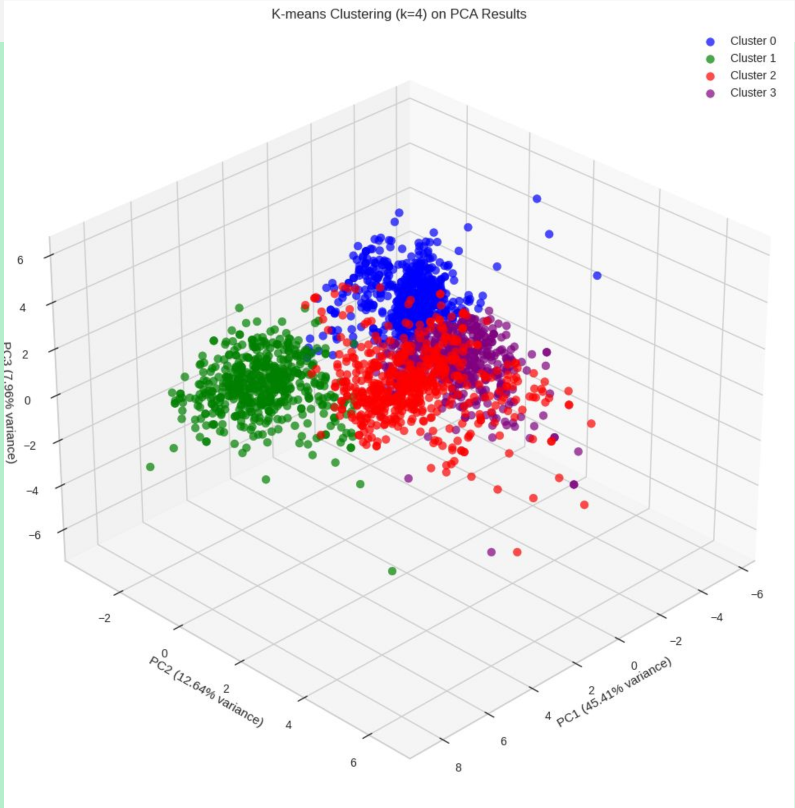
CHOOSE N CLUSTERS

The **elbow method** evaluates the distortion score, which measures how tightly the data points fit within each cluster.
The **silhouette score** measures how similar each point is to its own cluster compared to others, ranging from -1 to 1.
A higher score means more distinct, well-separated clusters. With both methods, we chose 4 clusters for K-Means





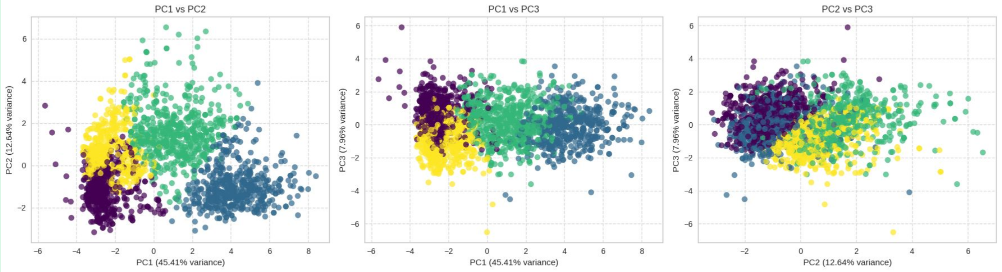
K-MEANS



K-Means is an **unsupervised** machine learning algorithm that partitioned our data into 4 distinct clusters based on feature similarity. It iteratively assigns data points to the nearest cluster centroid and updates centroids until convergence.

The visualizations show the clustering results projected onto the first three principal components (PC1–PC3), which capture the most variance in the data.

We can see that clusters are well-separated with minor overlapping in the reduced dimensional space.



K-MEANS FINDINGS

CLUSTER	0	1	2	3
AGE	61	57	59	49
SPEND	$170	$1415	$875	$97
FAMILY SIZE	2.8	1.1	2.1	1.8
BEHAVIOR	Moderate use of deals and store purchases	Prefer catalog and store channels – Least responsive to deals	Most deal purchases and highest web purchases	High web visits but low purchase activity

K-MEANS CLUSTER FINDINGS

Low-Spending, Older Families

- Moderate income
- Older age
- Very low spending
- **Value-conscious**

$170

Average spend per customer

Very High-Income, High-Spending

- Highest income
- Slightly younger
- Very high spending
- **Loyal and affluent**

$1,415

Average spend per customer

High-Income, Digitally Active

- High income
- Slightly older
- High spending
- **Active and valuable**

$875

Average spend per customer

Young, Low-Income, Low-Spending

- Low income
- Youngest group
- Lowest spending
- **Price-sensitive browsers**

$97

Average spend per customer

HIGH VALUE PERSONA – High Income High Spending

$1,415

Average spend per customer

Education

- Graduate

Age

- 57

Products

- High spending across all product categories

Income

- $77K

Household

- 1.1, mostly educated professionals

Promotion

- Prefer catalog/stores
- Least responsive to deals

K-MEANS CLUSTER RECOMMENDATION

Low-Spending, Older Families

- Value bundles on essentials
- Loyalty programs tailored to older family shoppers “Family Saver” or “Senior Advantage”

Very High-Income, High-Spending

- VIP membership
- Cross-sell premium items (wine/cold cuts)
- Emphasize high-quality service
- Avoid discount-heavy messages

High-Income, Digitally Active

- Targeted online promos + flash sale
- Personalized deal recommendation
- Optimize digital ads
- Loyalty apps and gamified rewards

Young, Low-Income, Low-Spending

- New customer voucher
- Students or new starters discount
- Engage through social media and influencer marketing
- Aim for retention as their income grows

APRIORI ALGORITHM

The **Apriori algorithm** is a classic data mining technique used to **identify frequent itemsets and generate association rules**, most notably applied in market basket analysis to find which products are often purchased together

Apriori relies on the principle that **all non-empty subsets of a frequent itemset must also be frequent**. If a set of items is not frequent, none of its supersets can be frequent either

It can effectively help us answer the following types of questions:

- **Which items are frequently bought together?**
- **What are the most common combinations of items?**
- **If a customer buys item X, how likely are they to buy item Y?**
- **What customer behaviors or preferences are linked?**

We answered these questions for across all clusters, as well as explored them for each specific cluster to inform customer segment specific recommendations

APRIORI PRODUCT INSIGHTS CUSTOMER SEGMENTS

Low-Spending, Older Families

- Focus Marketing on Associated Product Groups
- Not High Return Individual Customers, Group Potential

1. **Wines+Meats**
2. **Fruits+Meats**

Very High-Income, High-Spending

- 'Biggest Buyers'
- Focus Marketing on Top Ranked Associated Product Groups

1. **Meat+Fish+Sweets**
2. **Meat+Fish**
3. **Meat+Wine**

High-Income, Digitally Active

- Individual Product Associations Rank Higher That Groups
- Focus Marketing on Top Ranked Individual Products

1. **Meat**
2. **Gold**
3. **Wines**

Young, Low-Income, Low-Spending

- 'Non-buyer' & 'no-buyer'
- Focus Marketing on Associated Product Groups
- Not High Return Individual Customers, Group Potential

1. **Fruits+Sweets**
2. **Wines+Meats**

APRIORI PRODUCT INSIGHTS CUSTOMER CROSS-SEGMENT

Customers who buy Sweets tend to buy Fish and Fruits more than all other product combinations

- in about **13%** of all transactions
- in about **77%** of transactions with sweets as the antecedent
- are purchased much more often together than by random chance

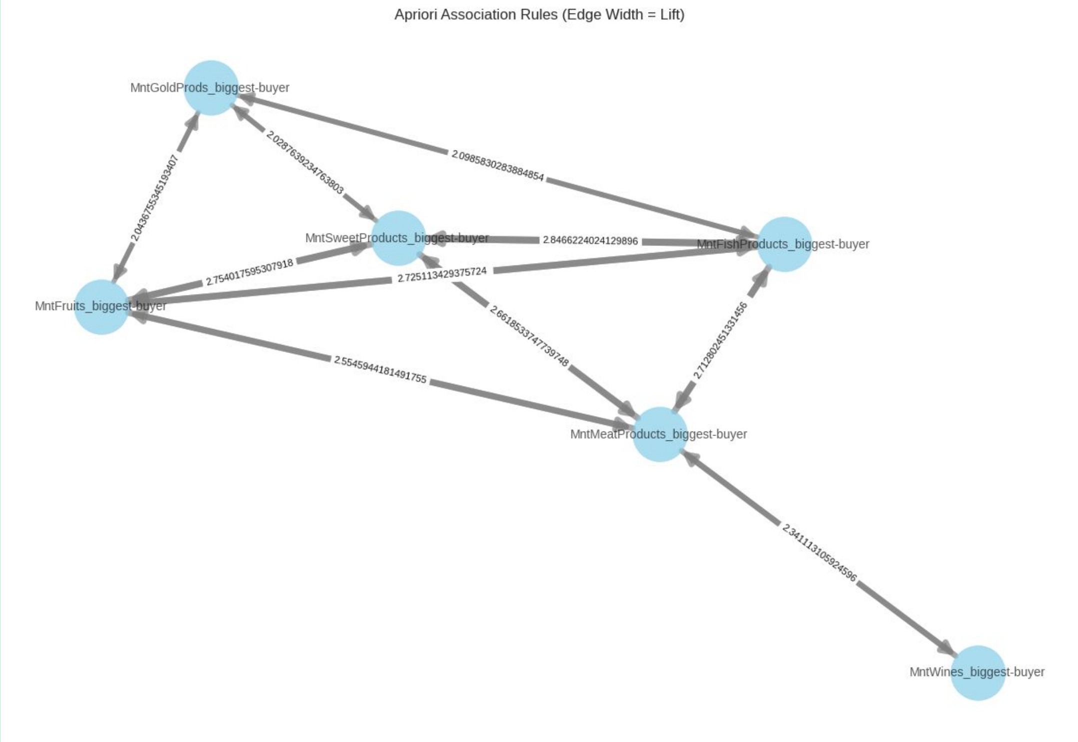
Customers who Sweets tend to buy Meat

- in about **17%** of all transactions
- in about **66%** of transactions with sweets as the antecedent
- more often together than by random chance

Customers who buy Meat tend to buy Wine

- in about **15%** of all transactions
- in about **59%** of transactions with meat as the antecedent
- more often together than by random chance

VISUALIZING PRODUCT PURCHASE RELATIONSHIPS



OUR PRODUCT RECOMMENDATIONS

1

Market **Fish Products** and Product Associations with Fish to All Customer Segments

2

Target **'Biggest Buyers'** with **Meat+Fish+Sweets** Combos

3

Market **Meat**, **Gold**, **Wine** Individually to **Mid-Income, Digitally Active Group**

4

Place **Sweets**, Such as Bakery, **Near Meat Butcher/Deli** - Online/In-App **Recommendations**

5

Market **Meat and Wines** Together

OUR BUSINESS RECOMMENDATIONS

Data-Driven Product Bundling

Action

1. Create product bundles using suggested combos
2. Run "Buy A, Get B at 10% Off offers"

Business Impact

- Increase average order value & total revenue
- Boost conversion rate
- Drive repeat purchase

Segment-Based Targeted Marketing

Action

1. Personalized campaigns targeting (eg. in-store VIP perks for cluster 1, digital deals for cluster 2)
2. Retarget via email, ads, loyalty platform

Business Impact

- Increase ROI on ad spend
- Improve customer retention and lifetime value

Smart Product Placement (Online + In-Store)

Action

1. Position suggested combo together
2. Suggest "Frequently Bought Together" on online platform

Business Impact

- Increase Basket Size
- Encourage product discovery and better UX

Product Development and Inventory Strategy

Action

1. Develop segment-specific SKUs (luxury collection vs. budget kits)
2. Align inventory with cluster-level demand forecast

Business Impact

- Improve product market fit
- Optimized inventory cost

THANK YOU