

CUSTOMER SEGMENTATION

Study Case Industry



OUTLINE

INTRODUCTION CUSTOMER SEGMENTATION

Introducing customer analysis and segmentation, problem statement, and background.

METHOD FOR CUSTOMER SEGMENTATION

Explain about some various method for customer segmentation.

HANDS ON !!!

INTRODUCTION CUSTOMER SEGMENTATION

WHAT IS CUSTOMER SEGMENTATION ?

- Customer segmentation is the practice of dividing a customer base into **groups** of individuals that are **similar** in **specific ways** relevant to marketing, such as age, gender, interests and spending habits.
- Customer segmentation can be obtained by doing customer analytics. Customer analytics is critical for gaining a thorough understanding of customers' **purchasing habits**, **use trends**, **demographic distribution**, and **profitability**. Organizations must devote a significant amount of time and resources to learning about their **consumers** and **analyzing** the data created by their interactions with them.
- Customer segmentation is important for businesses to understand their target audience. Different advertisements can be curated and sent to different audience segments based on their **demographic profile**, **interests**, and **affluence level**.

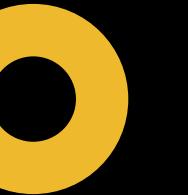
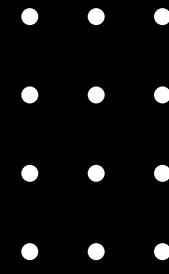
Customer Segmentation Procedures

- Deciding **what data** will be **collected** and how it will be gathered
- **Collecting data** and **integrating data** from various sources
- Developing **methods of data analysis** for segmentation
- Establishing **effective communication** among relevant business units (such as marketing and customer service) about the segmentation
- **Implementing applications** to effectively deal with the data and respond to the information it provides

Benefits of Customer Segmentation

By enabling companies to **target specific groups** of customers, a customer segmentation model allows for the **effective allocation** of marketing resources and the **maximization of cross- and up-selling opportunities**. When a group of customers is sent personalized messages as part of a marketing mix that is designed around their needs, it's easier for companies to send those customers special offers meant to **encourage** them to **buy more products**. Customer segmentation can also **improve customer service** and **assist in customer loyalty** and **retention**.

As a by-product of its personalized nature, marketing materials sent out using customer segmentation tend to be more valued and appreciated by the customer who receives them as opposed to impersonal brand messaging that doesn't acknowledge **purchase history** or any kind of **customer relationship**.



METHOD FOR CUSTOMER SEGMENTATION



RFM (Recency, Frequency, and Monetary)

ABOUT RFM

RFM segmentation enables marketers to target specific groups of consumers with communications that are far more relevant to their individual behaviours, resulting in much greater response rates and improved loyalty and customer lifetime value. RFM segmentation, like other segmentation approaches, is an effective tool to identify groups of consumers who should be treated differently. RFM stands for recency, frequency, and monetary

Why Should We Choose RFM?

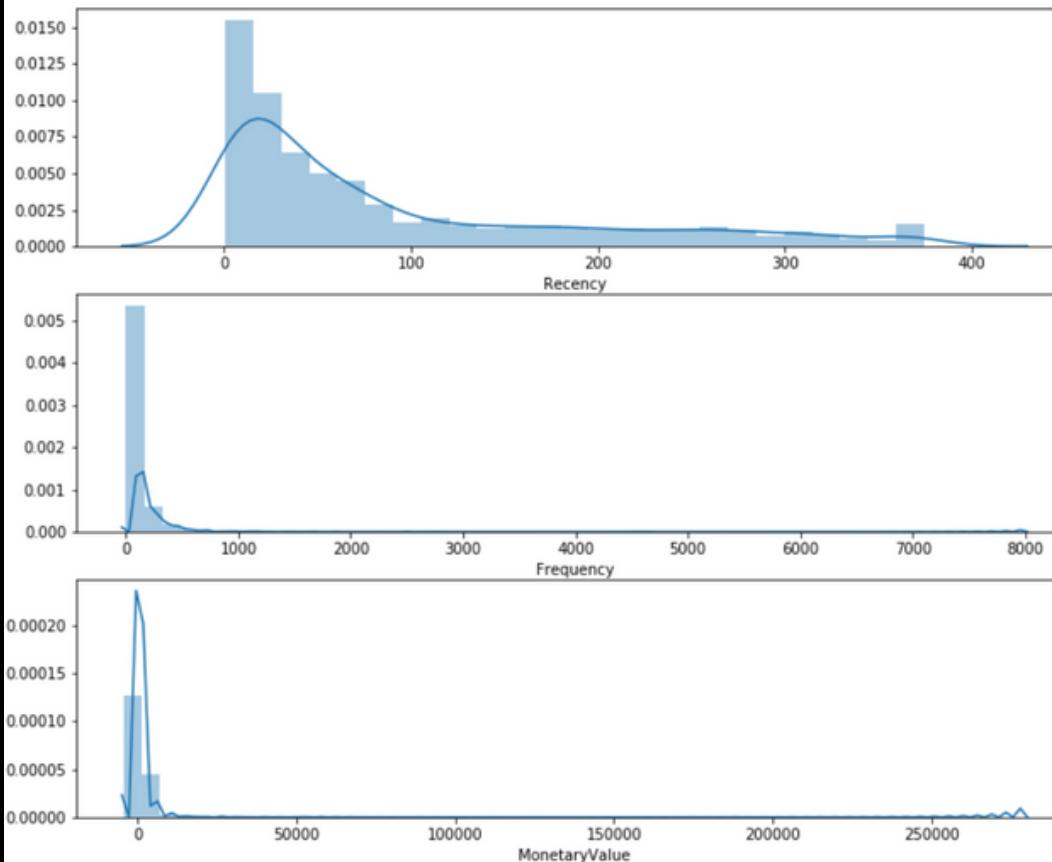
- It employs objective numerical scales to produce a high-level picture of consumers that is both succinct and instructive.
- It's simple enough that marketers can utilize it without expensive tools.
- It's simple — the segmentation method's output is simple to comprehend and analyze.

How to Build RFM with Python

Step 1: Importing Required Libraries

```
1 import pandas as pd
2 import numpy as np
3 from scipy import stats
4 from datetime import timedelta
5 import matplotlib.pyplot as plt
6 %matplotlib inline
7 import seaborn as sns
8 import squarify
```

Step 2: Explorative Data Analysis (EDA)

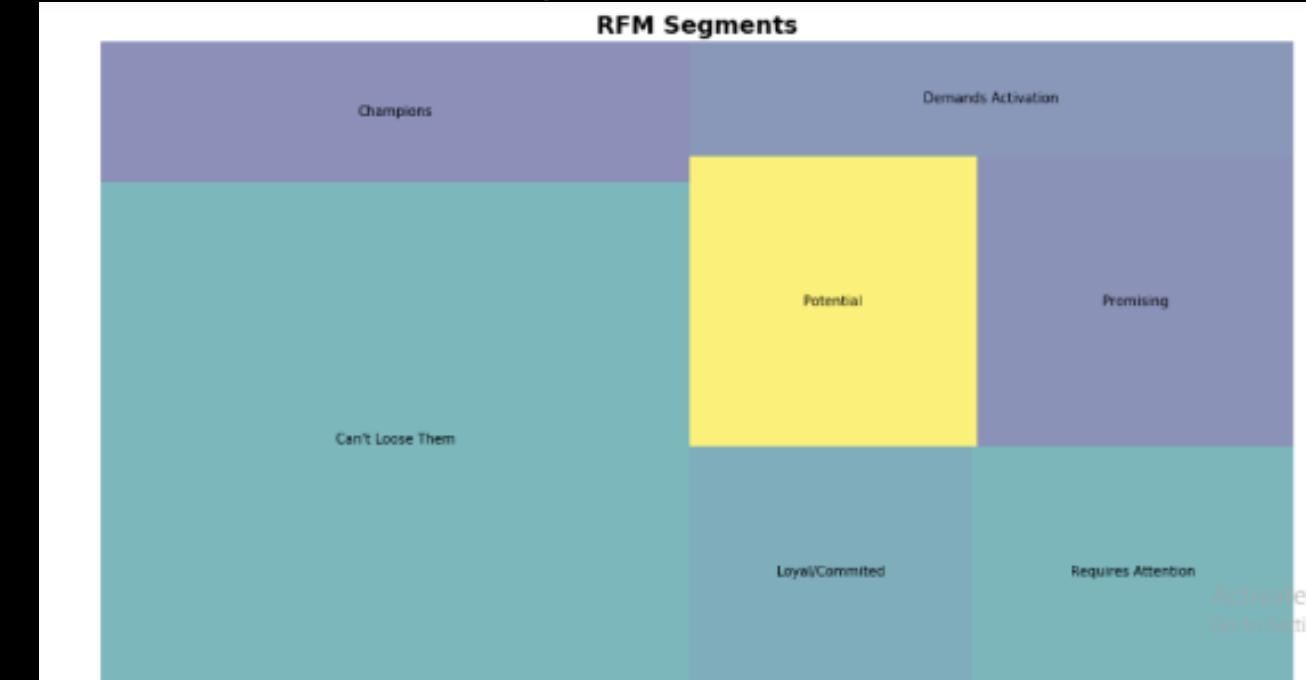


Step 3: Data Preprocessing

```
# Concatenating the RFM quartile values to create RFM Segments
def join_rfm(x): return str(x['R']) + str(x['F']) + str(x['M'])
FMCG_Preprocess['RFM_Segment_Concat'] = FMCG_Preprocess.apply(join_rfm, axis=1)
FMCG_rfm = FMCG_Preprocess
FMCG_rfm.head()
```

	Recency	Frequency	MonetaryValue	R	F	M	RFM_Segment_Concat
CustomerID							
12346.0	326	2	0.00	1	1	1	111
12347.0	2	182	4310.00	4	4	4	444
12348.0	75	31	1797.24	2	2	4	224
12349.0	19	73	1757.55	3	3	4	334
12350.0	310	17	334.40	1	1	2	112

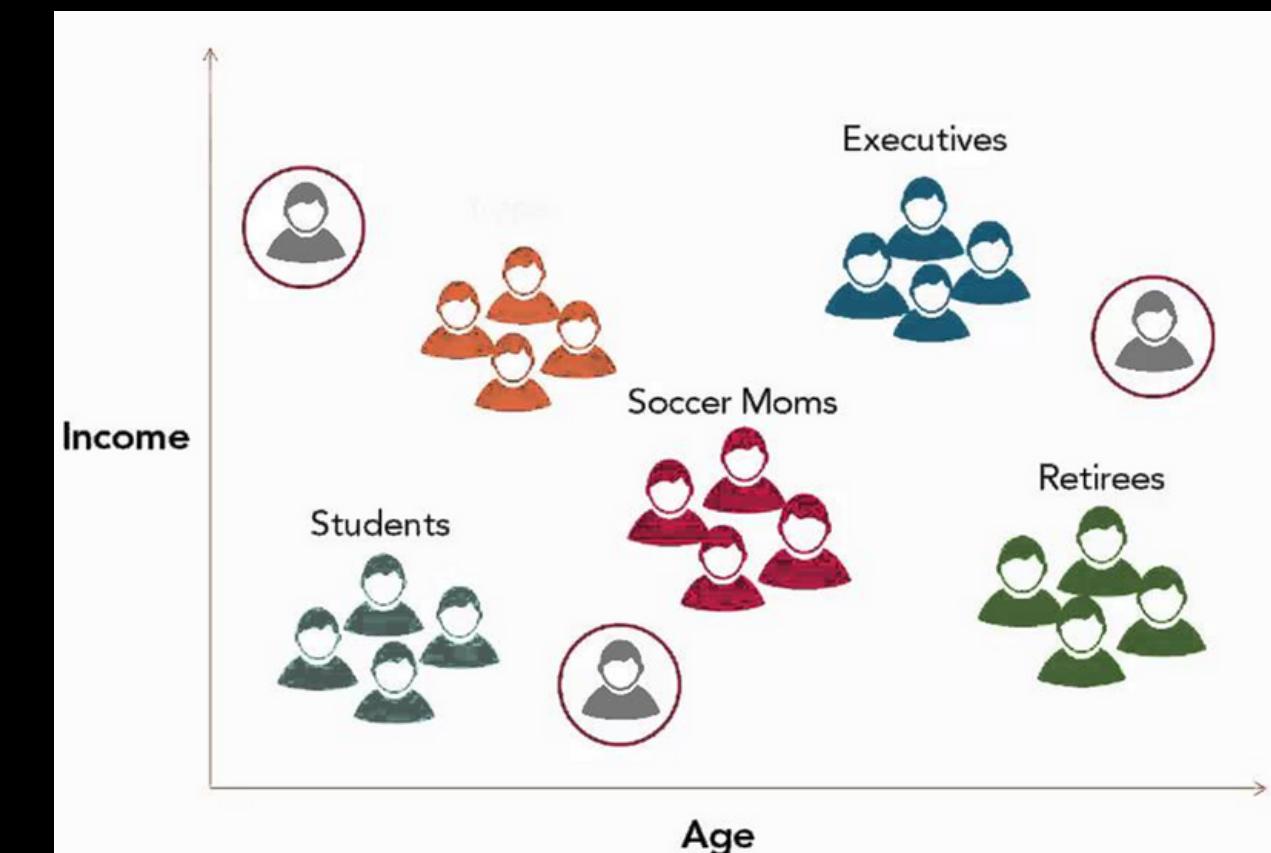
Step 4: Building the RFM Groups



Clustering Method

K-Means Clustering

K-Means clustering is a commonly used technique by data scientists to help companies with customer segmentation. The K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible. The ‘means’ in the K-means refers to averaging of the data; that is, finding the centroid.



K-Means Clustering with Python

Imports and reading the data frame

```
# Imports

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import sea
from kneed import KneeLocator
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from mpl_toolkits.mplot3d import Axes3D

# reading the data frame

df = pd.read_csv('Mall_Customers.csv')
```

	CustomerID	Gender	Age	Annual Income (k\$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

Standardizing variables

```
col_names = ['Annual Income (k$)', 'Age', 'Spending Score (1-100)']
features = df[col_names]
scaler = StandardScaler().fit(features.values)
features = scaler.transform(features.values)
scaled_features = pd.DataFrame(features, columns = col_names)
scaled_features.head()
```

	Annual Income (k\$)	Age	Spending Score (1-100)
0	-1.738999	-1.424569	-0.434801
1	-1.738999	-1.281035	1.195704
2	-1.700830	-1.352802	-1.715913
3	-1.700830	-1.137502	1.040418
4	-1.662660	-0.563369	-0.395980

K-Means Clustering with Python (cont'd)

One hot encoding

```
gender = df['Gender']
newdf = scaled_features.join(gender)

newdf = pd.get_dummies(newdf, prefix=None, prefix_sep='_', dummy_na=False, columns=None, s| 
newdf = newdf.drop(['Gender_Male'],axis=1)

newdf.head()
```

	Annual Income (k\$)	Age	Spending Score (1-100)	Gender_Female
0	-1.738999	-1.424569	-0.434801	0
1	-1.738999	-1.281035	1.195704	0
2	-1.700830	-1.352802	-1.715913	1
3	-1.700830	-1.137502	1.040418	1
4	-1.662660	-0.563369	-0.395980	1

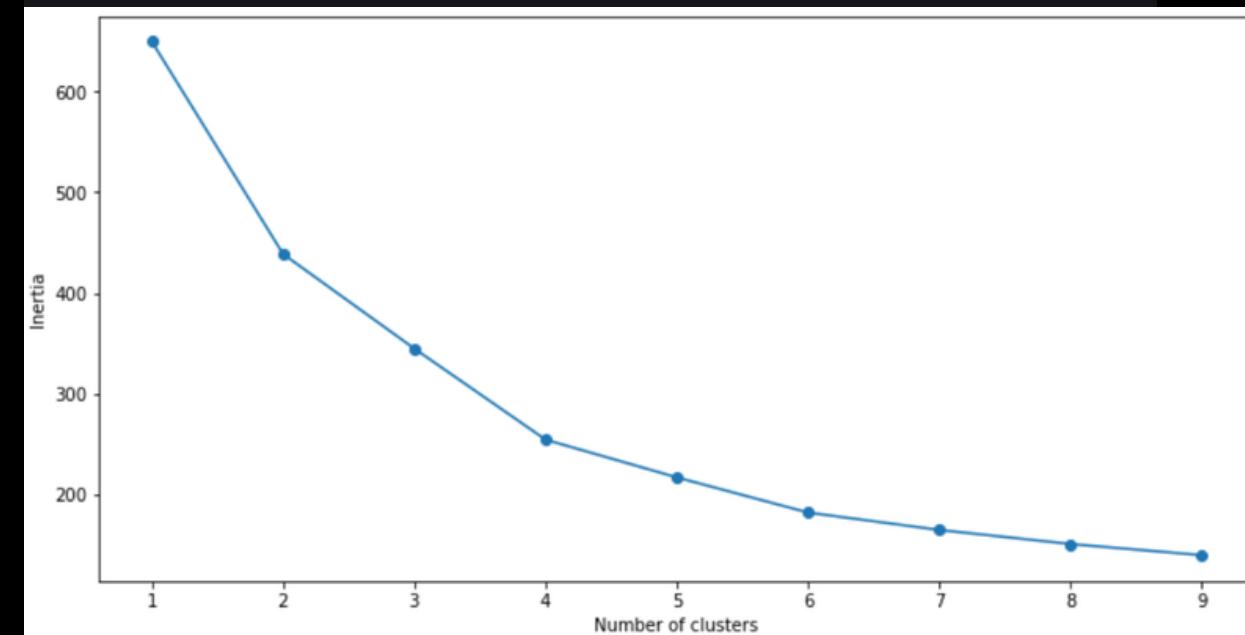
Building the clustering model

```
SSE = []

for cluster in range(1,10):
    kmeans = KMeans(n_jobs = -1, n_clusters = cluster, init='k-means++')
    kmeans.fit(newdf)
    SSE.append(kmeans.inertia_)

# converting the results into a dataframe and plotting them

frame = pd.DataFrame({'Cluster':range(1,10), 'SSE':SSE})
plt.figure(figsize=(12,6))
plt.plot(frame['Cluster'], frame['SSE'], marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Inertia')
```



CLUSTER ANALYSIS

Main attributes of each segment

Cluster 0:

- High average annual income, low spending.
- Mean age is around 40 and gender is predominantly male.

Cluster 1:

- Low to mid average income, average spending capacity.
- Mean age is around 50 and gender is predominantly female.

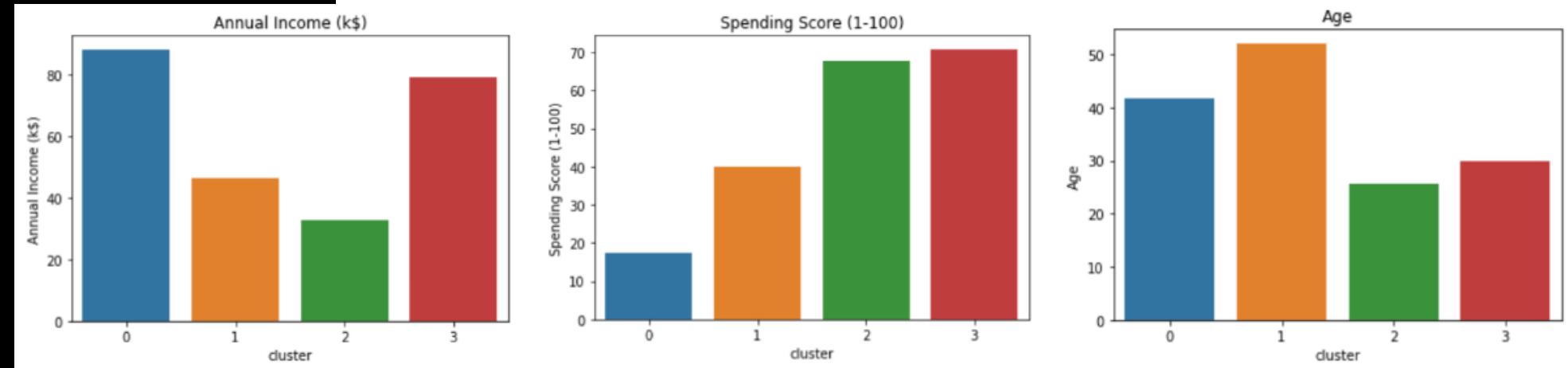
Cluster 2:

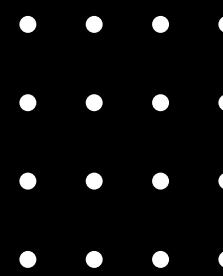
- Low average income, high spending score.
- Mean age is around 25 and gender is predominantly female.

Cluster 3:

- High average income, high spending score.
- Mean age is around 30 and gender is predominantly female.

cluster	Age	Annual Income (k\$)	Spending Score (1-100)
0	0	41.685714	88.228571
1	1	52.144928	46.333333
2	2	25.605263	32.631579
3	3	30.000000	79.086207





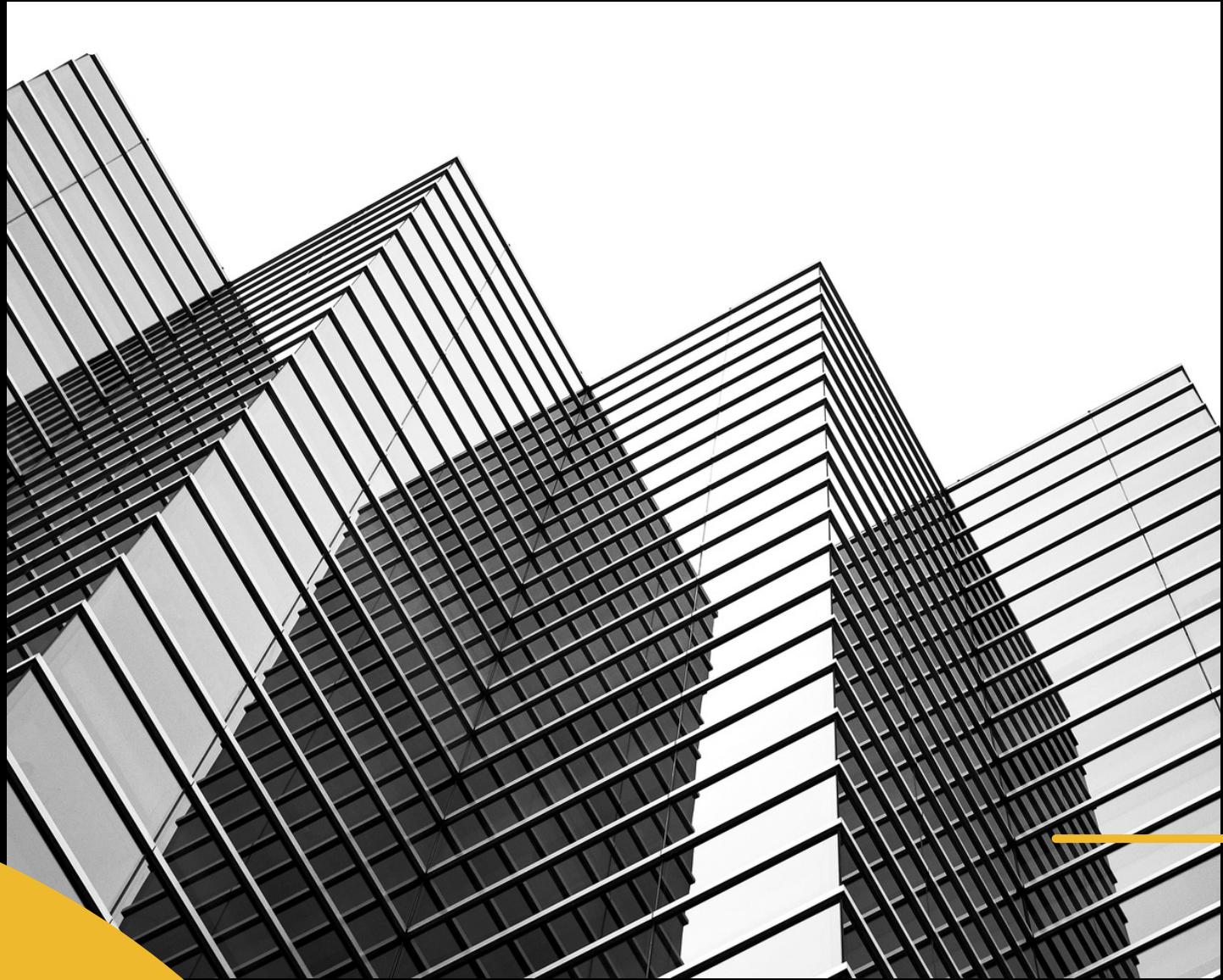
REFERENCES

<https://medium.com/@ogunbajoadeyinka/building-an-rfm-model-in-python-3481112a2816>

<https://www.natasshaselvaraj.com/customer-segmentation-with-python/>

<https://towardsdatascience.com/customer-segmentation-analysis-with-python-6afa16a38d9e>





THANK
YOU

Bisa.ai Kampus
Merdeka
INDONESIA JAYA