PREPARED FOR
Professor - Eyyub Kibis

PREPARED BY
GROUP -03

FINAL PROJECT

AMAZON - INDIA

2023

AMAZON'S INTERNAL MARKET SHARE (AMAZON VS MERCHANT) IN INDIA.

4'P OF MARKETING COMPARATION.

Prepared by,
Harsh Saxena
Mariana Tapieros Arias
Olid Sarker
Chirag Mahayan

ABSTRACT

Developed a dashboard that identified and compared the internal market apparel share between Amazon fulfillment (FBA) and its merchants. We observed and analyzed that FBA (Fulfilment by Amazon) has almost 70% market share based on revenues. On the other hand, all merchants contribute 30% of revenues or market share. Our team justified the market share based on 4'P of marketing (Price, Place, Product, and Promotion). FBA holds the maximum number or percentage on the criteria mentioned. Although merchants dominate particular products and states, the performance of the FBA was good for almost all marketing criteria, with many fluctuations. Merchants may need help with their performance because of external shipping services and delivery methods. Additionally, 22% of the merchants' orders were canceled, and for FBA, just 12% of their total.

INTRODUCTION

Dataset:

Rows: 128975
Columns: 24

Target: Amazon Sale Report.csv (Category)

Categorical Variables: OrderID, Date, Status, Fulfillment, Sales Channel, ship-service-level, SKU, Category, currency, ship-city, ship-state, ship-country, promotion-ids, B2B, fulfilled-by.

Numerical/Continuous Variables: index, Amount, ship-postal-code.

- Data Collection
- Data Cleaning
- Data Preprocessing
- Data Visualization
- Analysis & Results

Our team designed a dashboard with a friendly visualization that shows the percentage of internal apparel market share and the performance of the operations between FBA and Merchants. According to Kaggle, the E-Commerce Sales dataset released by THE DEVASTATOR Includes variables orderID, Date, Status, Fulfillment, Sales Channel, ship-service-level, SKU, Category, currency, ship-city, ship-state, ship-country, promotion-ids, B2B, fulfilled-by.

The primary objective of this project is to develop future marketing positioning strategies by identifying and comparing the market share of individuals. So, we visualized all measurement criteria per our superior's instruction so that they could make a final decision.

Data used: https://www.kaggle.com/datasets/thedevastator/unlock-profits-with-e-commerce-sales-data

Variables Description:

- **OrderID**: It is a unique identifier for each order of the category of dresses.
- **Date**: Date of the order.
- **Status**: It tells the status to the customer whether the item is Shipped, Canceled or Pending.
- **Fulfillment**: Tells if the item will be fulfilled by another merchant OR called as a third-party company or Amazon.
- **Sales Channel**: Showing the sales of the category/item where it is sold that is: Amazon.in
- **ship-service-level**: It shows whether the category/item of service is Standard or Expedited.
- **SKU**: Shows SKU number of each of the unique Category and it is in String datatype.
- **Category**: Shows different types of categories like Set, kurta, Western-Dress, Top, Blouse, and bottom.
- **Currency**: The currency used here is Indian Rupees (INR) or ₹
- **Ship-postal-code**: It shows which type of category is going to be shipped by postal code/zip code.
- **Ship-city**: Shows the city name for India.
- **Ship-state**: Shows the state name for India.
- **Ship-country**: Shows where the category/item is being shipped to and to which country.
- **Promotion-ids**: It is a unique id for different types of promotion for each category or an item.
- **B2B**: Shows if the category/item is Business-to-Business either False (0) or True (1)
- **Index**: It is an integer.
- **Amount**: Shows the amount in Indian Rupee (INR) in decimals, and integers.

Methodology: The present study uses data wrangling and data warehousing analysis to clean and visualize the data. Beside data visualization. The objective of this project was to compare FBA vs Merchant by 4'p of marketing to identify the power they have in the market for future positioning strategies.

Data Collection

We collected our dataset to conduct our research about features, decision-making and strategy of Dresses the category like Western Dress, Kurta, Set, Ethnic Dress, Top, Blouse, Bottom with sizes of S, L, M, XL, XXL, 3XL, 4XL, 6XL.

The dataset was collected from Kaggle website (Amazon Sale Report.csv), and it contains two types of data: quantitative and qualitative. With the use of Python and libraries such as pandas, numpy, and Tableau.

In [1]:
```
import pandas as pd
import numpy as np
# suppressing warnings
import warnings
warnings.filterwarnings('ignore')
pd.options.display.float_format = '{:.6f}'.format
```

Exploring the Dataset:

As we mentioned previously, the dataset contains 128975 rows and 24 Columns.

```
In [173]: df.shape[0]
Out[173]: 128975

In [174]: df.shape[1]
Out[174]: 24
```

The data type we will be working on are numerical and categorical.

Data Type

```
In [175]: df.dtypes
Out[175]: index                    int64
          Order ID                object
          Date                    object
          Status                  object
          Fulfilment              object
          Sales Channel           object
          ship-service-level      object
          Style                   object
          SKU                     object
          Category                object
          Size                    object
          ASIN                    object
          Courier Status          object
          Qty                      int64
          currency                object
          Amount                 float64
          ship-city               object
          ship-state              object
          ship-postal-code       float64
          ship-country            object
          promotion-ids           object
          B2B                       bool
          fulfilled-by            object
          Unnamed: 22             object
          dtype: object
```

Statical Analysis

Descriptive statistic of our numerical and categorical variables: to identify relationship, patterns and correlations between our variables and outliers.

Mean, Standard Deviation, Min and Max using Descriptive Statistics

In [181]:
```
df.describe()
```

Out[181]:

	index	Qty	Amount
count	128975.000000	128975.000000	121180.000000
mean	64487.000000	0.904431	648.561465
std	37232.019822	0.313354	281.211687
min	0.000000	0.000000	0.000000
25%	32243.500000	1.000000	449.000000
50%	64487.000000	1.000000	605.000000
75%	96730.500000	1.000000	788.000000
max	128974.000000	15.000000	5584.000000

Median

In [182]:
```
pd.DataFrame(df[Numerical_Variables].median()).rename(columns={0:'Median'})
```

Out[182]:

	Median
Qty	1.000000
index	64487.000000
Amount	605.000000

Mode

In [183]:
```
df.mode()[:1].T.rename(columns={0:'Mode'})
```

Out[183]:

	Mode
index	0
Order ID	171-5057375-2831560
Date	05-03-22
Status	Shipped
Fulfilment	Amazon
Sales Channel	Amazon.in
ship-service-level	Expedited
Style	JNE3797
SKU	JNE3797-KR-L
Category	Set
Size	M
ASIN	B09SDXFFQ1
Courier Status	Shipped
Qty	1.000000
currency	INR
Amount	399.000000
ship-city	BENGALURU
ship-state	MAHARASHTRA
ship-postal-code	201301.000000
ship-country	IN
promotion-ids	IN Core Free Shipping 2015/04/08 23-48-5-108
B2B	False
fulfilled-by	Easy Ship
Unnamed: 22	False

Variance

```
In [184]: pd.DataFrame(df[Numerical_Variables].var()).rename(columns={0:'Variance'})
```

Out[184]:

	Variance
Qty	0.098190
index	1386223300.000000
Amount	79080.013034

Min

```
In [185]: pd.DataFrame(df.min()).rename(columns={0:'Min'})
```

Out[185]:

	Min
index	0
Order ID	171-0000547-8192359
Date	03-31-22
Status	Cancelled
Fulfilment	Amazon
Sales Channel	Amazon.in
ship-service-level	Expedited
Style	AN201
SKU	AN201-RED-M
Category	Blouse
Size	3XL
ASIN	B01LYC0N7Q
Qty	0
Amount	0.000000
ship-postal-code	110001.000000
B2B	False
Unnamed: 22	False

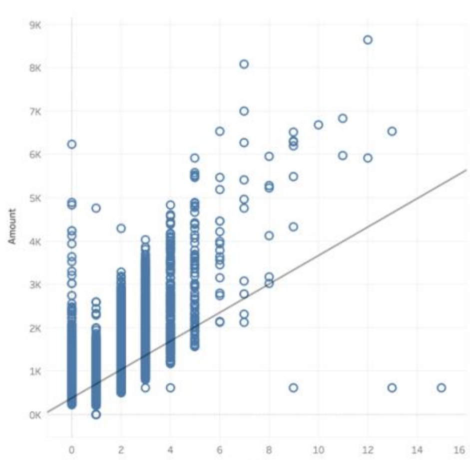
We can see that the variable amount strongly correlates with the variable Qty. Additionally, we can notice in our final dashboard how the sale performed is affected by the type of fulfillment (Merchants vs Amazon).

Correlation Matrix

```
In [188]: df.corr()
```

Out[188]:

	index	Qty	Amount
index	1.000000	0.010621	0.047571
Qty	0.010621	1.000000	0.066900
Amount	0.047571	0.066900	1.000000



Data Preprocessing

Our data reprocessing was a step in the data mining and analysis process that let us understand the data for further analysis—starting through identifying missing values (Data cleaning). Followed dropping certain variables by discretion.

Percentage of null value with matrix:

Percentage of Null Values

```
In [187]: pd.DataFrame(df.isnull().sum() * 100 / len(df)).rename(columns={0:'% Null Values'})
```

Out[187]:

	% Null Values
index	0.000000
Order ID	0.000000
Date	0.000000
Status	0.000000
Fulfilment	0.000000
Sales Channel	0.000000
ship-service-level	0.000000
Style	0.000000
SKU	0.000000
Category	0.000000
Size	0.000000
ASIN	0.000000
Courier Status	5.328164
Qty	0.000000
currency	6.043807
Amount	6.043807
ship-city	0.025586
ship-state	0.025586
ship-postal-code	0.025586
ship-country	0.025586
promotion-ids	38.110487
B2B	0.000000
fulfilled-by	69.546811
Unnamed: 22	38.030626

Null and unique value:

Table of df type, null values and unique values for better visualization

In [189]:
```
def printinfo():
    temp = pd.DataFrame(index= df.columns)
    temp['data_type'] = df.dtypes
    temp['null_count'] =df.isnull().sum()
    temp['unique_count'] = df.nunique()
    return temp
printinfo()
```

Out[189]:

	data_type	null_count	unique_count
index	int64	0	128975
Order ID	object	0	120378
Date	object	0	91
Status	object	0	13
Fulfilment	object	0	2
Sales Channel	object	0	2
ship-service-level	object	0	2
Style	object	0	1377
SKU	object	0	7195
Category	object	0	9
Size	object	0	11
ASIN	object	0	7190
Courier Status	object	6872	3
Qty	int64	0	10
currency	object	7795	1
Amount	float64	7795	1410
ship-city	object	33	8955
ship-state	object	33	69
ship-postal-code	object	33	9459
ship-country	object	33	1
promotion-ids	object	49153	5787
B2B	object	0	2
fulfilled-by	object	89698	1
Unnamed: 22	object	49050	1

Dropping numerical varia with zero variance:

We proceed to analyze numerical variables with zero variance however we did observe any remarkable numerical variable with zero variance. By our own discretion we dropped the index variable since we consider it redundant for this case of study.

Dropping Variables

Dropping Numerical Variables Zero variance

```
In [190]: df[Numerical_Variables].std()

Out[190]: Qty          0.313354
          index    37232.019822
          Amount     281.211687
          dtype: float64

In [191]: df = df.drop(df[Numerical_Variables].std()[df[Numerical_Variables].std()== 0].index, axis = 1)

In [192]: df[Numerical_Variables].std()

Out[192]: Qty          0.313354
          index    37232.019822
          Amount     281.211687
          dtype: float64

In [193]: df = df.drop('index', axis = 1)
```

Drop Categorical Variables with Zero Variance:

Dropping Categorical Variables with Zero variance

```
In [195]: Categorical_Variables = list(df.select_dtypes(object).columns)
          Categorical_Variables

Out[195]: ['Order ID',
           'Date',
           'Status',
           'Fulfilment',
           'Sales Channel ',
           'ship-service-level',
           'Style',
           'SKU',
           'Category',
           'Size',
           'ASIN',
           'Courier Status',
           'currency',
           'ship-city',
           'ship-state',
           'ship-postal-code',
           'ship-country',
           'promotion-ids',
           'B2B',
           'fulfilled-by',
           'Unnamed: 22']

In [196]: len(Categorical_Variables)

Out[196]: 21

In [197]: zero_cardinality = []

          for i in Categorical_Variables:
              if len(df[i].value_counts().index) == 1:
                  zero_cardinality.append(i)
          zero_cardinality

Out[197]: ['currency', 'ship-country', 'fulfilled-by', 'Unnamed: 22']
```

Note: We will Not drop the 'ship-country' variable, since we will need it for visualization on tableau

```
In [198]: df = df.drop(['currency', 'fulfilled-by', 'Unnamed: 22'], axis =1)
```

Dropping Categorical Variables with Many Levels:

Dropping Categorical Variables with Many Levels

In [199]:
```
Categorical_Variables = list(df.select_dtypes(object).columns)
Categorical_Variables
```

Out[199]:
```
['Order ID',
 'Date',
 'Status',
 'Fulfilment',
 'Sales Channel ',
 'ship-service-level',
 'Style',
 'SKU',
 'Category',
 'Size',
 'ASIN',
 'Courier Status',
 'ship-city',
 'ship-state',
 'ship-postal-code',
 'ship-country',
 'promotion-ids',
 'B2B']
```

In [200]:
```
len(Categorical_Variables)
```

Out[200]:
```
18
```

In [201]:
```
high_cardinality = []

for i in Categorical_Variables:
    if len(df[i].value_counts().index) > 200:
        high_cardinality.append(i)

print(high_cardinality)
```

```
['Order ID', 'Style', 'SKU', 'ASIN', 'ship-city', 'ship-postal-code', 'promotion-ids']
```

Note: We will Not drop the 'Order ID', 'ship-city', 'ship-postal-code', 'promotion-ids' variable, since we will need it for visualization on tableau

In [202]:
```
df = df.drop(['Style', 'SKU', 'ASIN'], axis = 1)
```

Data Imputation for numerical variable:

Data Imputation

Numerical Variables

In [204]:
```
df.isnull().sum()
```

Out[204]:
```
Order ID                  0
Date                      0
Status                    0
Fulfilment                0
Sales Channel             0
ship-service-level        0
Category                  0
Size                      0
Courier Status         6872
Qty                       0
Amount                 7795
ship-city                33
ship-state               33
ship-postal-code         33
ship-country             33
promotion-ids         49153
B2B                       0
dtype: int64
```

In [205]:
```
Numerical_Variables = list(df.select_dtypes(exclude = object).columns)
Numerical_Variables
```

Out[205]:
```
['Qty', 'Amount']
```

In [206]:
```
df['Amount'].median()
```

Out[206]:
```
605.0
```

In [207]:
```
df['Amount'] = df['Amount'].fillna(df['Amount'].median(), inplace = False)
```

Data Imputation for categorical variable:

Categorial Variable

In [208]:
```
df.isnull().sum()
```

Out[208]:
```
Order ID                  0
Date                      0
Status                    0
Fulfilment                0
Sales Channel             0
ship-service-level        0
Category                  0
Size                      0
Courier Status         6872
Qty                       0
Amount                    0
ship-city                33
ship-state               33
ship-postal-code         33
ship-country             33
promotion-ids         49153
B2B                       0
dtype: int64
```

In [209]:
```
Categorical_Variables = list(df.select_dtypes(object).columns)
Categorical_Variables
```

Out[209]:
```
['Order ID',
 'Date',
 'Status',
 'Fulfilment',
 'Sales Channel ',
 'ship-service-level',
 'Category',
 'Size',
 'Courier Status',
 'ship-city',
 'ship-state',
 'ship-postal-code',
 'ship-country',
 'promotion-ids',
 'B2B']
```

In [212]:
```
df[Categorical_Variables].mode()
```

Out[212]:

	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	ship-city	ship-state	ship-postal-code	ship-country	promotion-ids	B2B
0	171-5057375-2831560	05-03-22	Shipped	Amazon	Amazon.in	Expedited	Set	M	Shipped	BENGALURU	MAHARASHTRA	201301.000000	IN	No Promotion	False

In [215]:
```
df = df.fillna({'Courier Status' : 'Shipped','promotion-ids': 'No Promotion', 'ship-country': 'IN' })
```

As we know our data, we cannot impute the location of the orders with the mode, since a state may not match with the city or zip code, for these reasons we can delete the rows or filter the data to not consider missing values for map graphs.

For our 'promotion-ids' variable, we cannot impute the data with its mode since orders with this missing value mean that they do not apply for promotion ids.

Dropping rows with missing values:

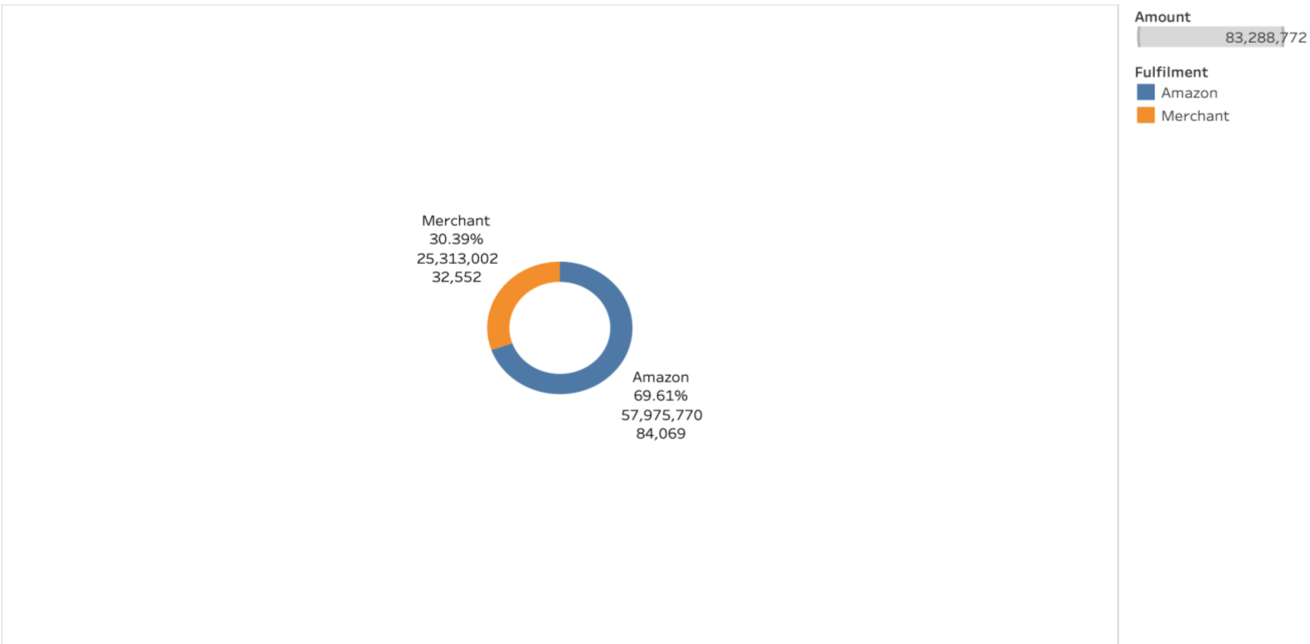
We considered that deleting rows that have these types of missing values was the best way to handle this data set since we consider that computing location may interfere with the veracity of the information.

```
In [216]: df.isnull().sum()

Out[216]: Order ID               0
          Date                   0
          Status                 0
          Fulfilment             0
          Sales Channel          0
          ship-service-level     0
          Category               0
          Size                   0
          Courier Status         0
          Qty                    0
          Amount                 0
          ship-city             33
          ship-state            33
          ship-postal-code      33
          ship-country           0
          promotion-ids          0
          B2B                    0
          dtype: int64

In [217]: df = df.dropna()

In [218]: df.isnull().sum()

Out[218]: Order ID              0
          Date                  0
          Status                0
          Fulfilment            0
          Sales Channel         0
          ship-service-level    0
          Category              0
          Size                  0
          Courier Status        0
          Qty                   0
          Amount                0
          ship-city             0
          ship-state            0
          ship-postal-code      0
          ship-country          0
          promotion-ids         0
          B2B                   0
          dtype: int64
```

Data Visualization

Market Share

Market Propotion by sale amount and quantity.

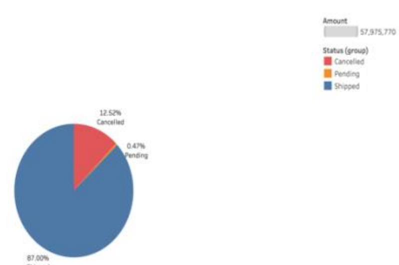


Minimum of Qty and minimum of Qty. For pane Minimum of Qty: Color shows details about Fulfilment. Size shows sum of Amount. The marks are labeled by Fulfilment, % de Ventas, sum of Amount and sum of Qty. The data is filtered on Status (group), Tooltip (Fulfilment,Ship-Country,Ship-State) and Tooltip (Ship-Country,Ship-State). The Status (group) filter keeps Cancelled, Pending and Shipped. The Tooltip (Fulfilment,Ship-Country,Ship-State) filter keeps 122 members. The Tooltip (Ship-Country,Ship-State) filter keeps 69 members. The view is filtered on Fulfilment, which keeps Amazon and Merchant.

Status Merchant

% de Ventas and Status (group). Color shows details about Status (group). Size shows sum of Amount. The marks are labeled by % de Ventas and Status (group). The data is filtered on Tooltip (Fulfilment) and Fulfilment. The Tooltip (Fulfilment) filter keeps 2 members. The Fulfilment filter keeps Merchant. The view is filtered on Status (group), which keeps Cancelled, Pending and Shipped.

Status Amazon

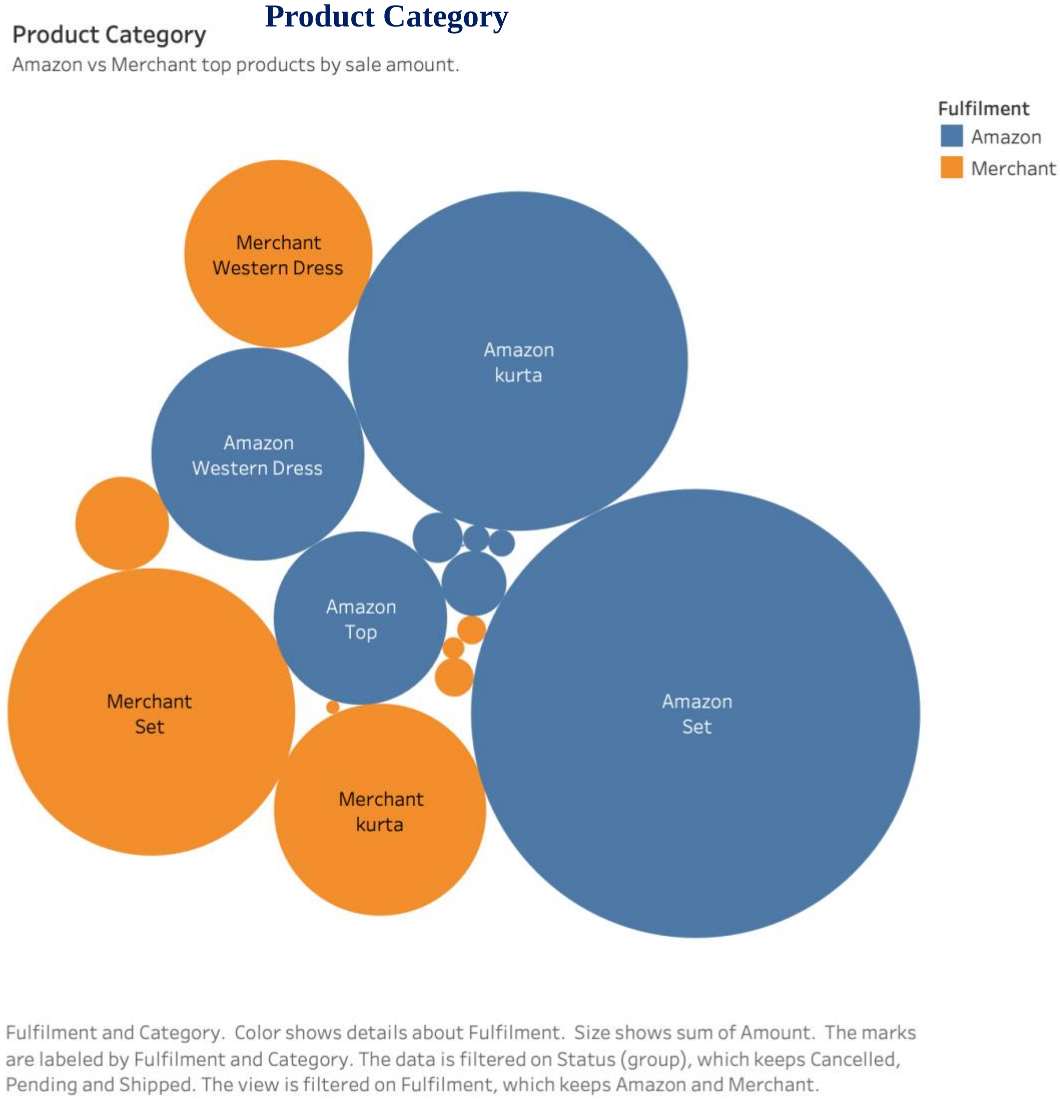


% de Ventas and Status (group). Color shows details about Status (group). Size shows sum of Amount. The marks are labeled by % de Ventas and Status (group). The data is filtered on Tooltip (Fulfilment) and Fulfilment. The Tooltip (Fulfilment) filter keeps 2 members. The Fulfilment filter keeps Amazon. The view is filtered on Status (group), which keeps Cancelled, Pending and Shipped.

Here we considered the market share based on sales. It is clear from the pie-chart that almost two-third percentage of the market share is captured by

FBA. Even though, merchant's market share is smaller than FBA, they experienced a higher percentage of order cancellations and pending.

Product Category



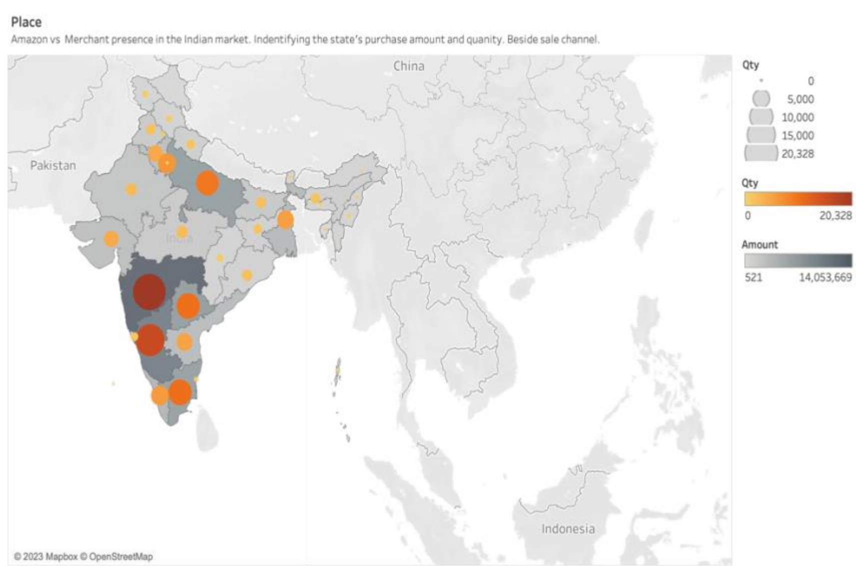
If we notice the western dress catalog, we will see that Merchants hold a 6% market share whereas FBA is 1% above but which should be 3 times higher than Merchants if we consider the overall market share. However, the popularity was acceptable for other categories based on the overall market share.

Price

The plot of count of Amount for Amount (bin). Color shows details about Fulfilment. The data is filtered on Status (group), which keeps Cancelled, Pending and Shipped. The view is filtered on Fulfilment, which keeps Amazon and Merchant.

In this graph we can identify that the price range of sale are mostly between $300-$800 either for Merchant or FBA .

Place



Map based on Longitude (generated) and Longitude (generated) and Latitude (generated). Details are shown for Ship-Country and Ship-State. For pane Longitude (generated): Color shows sum of Amount. For pane Longitude (generated) (2): Color shows sum of Qty. Size shows sum of Qty. The data is filtered on Fulfilment and Status (group). The Fulfilment filter keeps Amazon and Merchant. The Status (group) filter keeps Cancelled, Pending and Shipped. The view is filtered on Latitude (generated) and Longitude (generated). The Latitude (generated) filter keeps non-Null values only. The Longitude (generated) filter keeps non-Null values only.

Top State

Ship-State	
KARNATAKA	11,044,974
MAHARASHTRA	14,053,669
TELANGANA	7,335,881

Sum of Amount broken down by Ship-State. The data is filtered on Ship-State Set, Tooltip (Category,Fulfilment), Fulfilment and Status (group). The Ship-State Set filter keeps 3 members. The Tooltip (Category,Fulfilment) filter keeps 17 members. The Fulfilment filter keeps Amazon and Merchant. The Status (group) filter keeps Cancelled, Pending and Shipped.

Among all states we ranked top 3 states for both, and we also were able to identify in each state the amount of market share.

Others Finding

Shipping services:

Ship-Services

Fulfilment	Ship-Servic..	% Sale
Amazon	Expedited	69.44%
	Standard	0.17%
Merchant	Standard	30.39%

% de Ventas broken down by Measure Names vs. Fulfilment and Ship-Service-Level. The data is filtered on Status (group), which keeps Cancelled, Pending and Shipped. The view is filtered on Fulfilment, which keeps Amazon and Merchant.

We can observe that Merchant only has one mode for shipping. While FBA has two different types of shipment where most of their sale required expedited ship-services. We can see that merchant can explore to expand their ship-services.

Type of purchase:

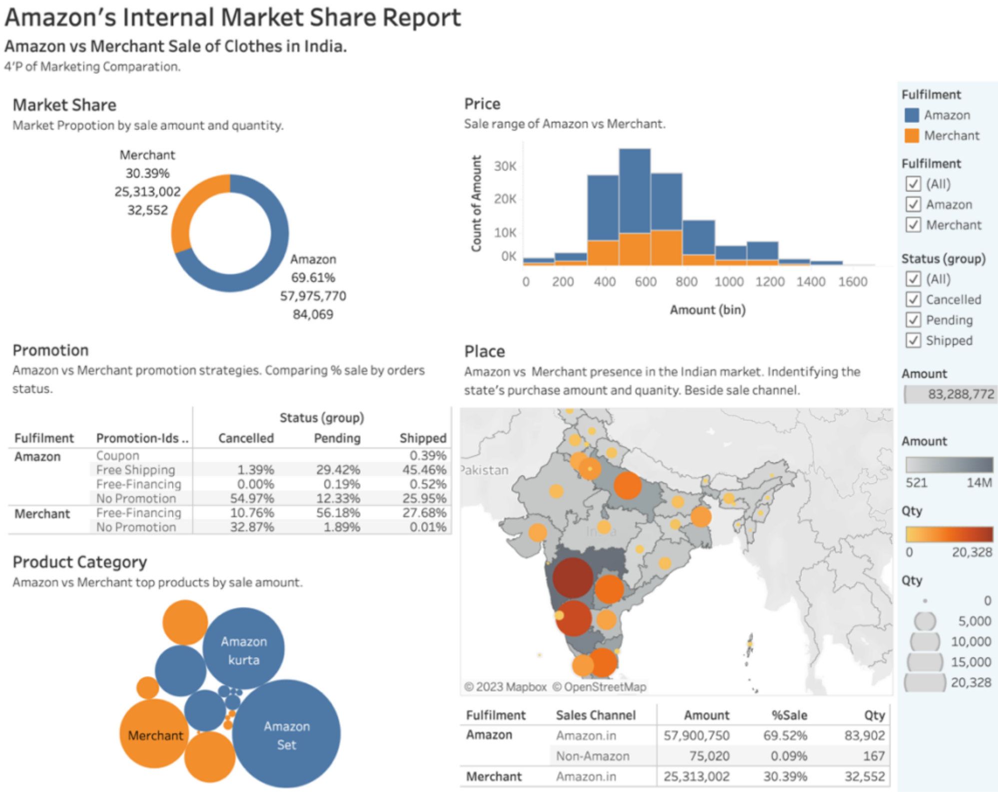
B2B

Fulfilment	B2B	% Sale
Amazon	False	69.13%
	True	0.48%
Merchant	False	30.14%
	True	0.25%

% de Ventas broken down by Measure Names vs. Fulfilment and B2B.

For this table we can conclude that lest the 1% of the purchase are for B2B. It can be inference that the products sale by Amazon are for personal use.

Analysis & Results



Our main intention of the project was to find out the areas where we will increase and 4 Marketing propositions (Price, Product, Place, Promotion).

To conduct the whole project properly we cleaned the data with python code and based on the clean data we visualized our requirements.

From the snapshot of the 4Ps, it was clear to our team that the 2/3 market share was captured by the FBA. We noticed Merchants always suffering from shipment, delivery, increasing sales, and ranking products. Merchants shipped their product directly to customers without quality checks and with the help of the 3rd party shipping company.

But Amazon always would use its own shipping service and must keep the products in its own warehouse for quality checks and fast shipping. That is why customers always trusted FBA service.

Conclusion

To sum up, it is clear to our team we must focus on Merchants operational performance to increase the market share and acceptance to customers. Now it is the time to develop a 4Ps marketing proposition strategy.

Public Tableau Link:

https://public.tableau.com/app/profile/mariana2012/viz/ProjectAmazonv1/4PofMarketingComparationAmazonvsMerchant?publish=yes