*Customer Segmentation*

**ABSTRACT:**

Consider a firm owning multiple branches across a country. The firm decides to analyze its customer in order to find most suitable place to open a new branch and understand the customer behavior. It will not only help in aiming maximum profit but the firm can serve the customers in more efficient ways. The research was conducted to analyze the effect of customers' level of experience towards their satisfaction, image and loyalty in a product of a company. Data collection had dataset of 200 people. The data gathered were then analyzed using Descriptive Analysis, Correlation, T-test, Mean Value Analysis, and Multiple Regressions. The result indicates that there was positive relationship between satisfaction, image and loyalty. New customers were proven to have different views towards satisfaction, image and loyalty compared to experienced customers.

Key words: *Data set, Customer, Spending Score, Income, Visualization*

**INTRODUCTION:**

Consumer analysis is the process where information about the consumer is found out from market research like the needs of the consumer, the target market and the relevant demographics so that this information can be used in market segmentation for further steps of market research.

Bar charts help in representing the categorical data using rectangular bars. They are used to compare the items between different groups over time. Pie charts are graphs that represent the data in the circular graph, the slices of pie show the relative size of the data. Pie chart is said to be the pictorial representation of data. Box plot is a measure of how well distributed the data is in a data set. It divides the data set into three quartiles and represents the minimum, maximum and median, first quartile and third quartile in the data set.

This project tries to visualize these important factors that determine the customer’s interest in any product, the decision-making unit, time and frequency of purchase, how the consumer makes the purchase and the method of payment. The former is called demographic analysis and the latter is called behavioral analysis.

**OBJECTIVES**

• To visualize the data set to maximum extent

• To learn the development of a dashboard using the html, css, javascript. • To apply various visualizing techniques learnt in class

• To use the django framework in the pycharm IDE

• To analyze the players performance and win the match strategically.

**REVIEW OF LITERATURE**

Consumer value begins to emerge in the 1990s as an issue of growing interest to business and, in particular, to marketing, at both the academic and practitioner levels. This concept is considered to be one of the most significant factors in the success of an organization and it has been pointed to as an important source of competitive advantage for the firm (Mizik and Jacobson 2003; Spiteri and Dion 2004; Woodruff 1997). Consumer value has been recognized as the fundamental basis in every marketing activity (Holbrook 1994, 1999), and it has been envisioned as a critical strategic weapon in attracting and retaining customers (Lee and Overby 2004; Wang, Lo, Chi, and Yang 2004).

Recognition of the relevance of this concept has generated important research focused on the study of its composition and its relationship with other concepts of interest to marketers such as satisfaction, trust, and loyalty. However, even though there is a significant body of knowledge about the concept of consumer value, this research is rather fragmented. The extent and heterogeneity of the various studies have created a dispersed, sometimes confusing and still-inconclusive base of knowledge about consumer value. As Wang et al. (2004) contend, different points of view about the meaning of value are advocated in the literature, with no widely accepted way of pulling views together. In this same sense, Ulaga (2001, p. 318) regards that "the fundamental question of how to conceptualize value still merits further investigation." Moreover, relevant studies have not yet yielded any unambiguous

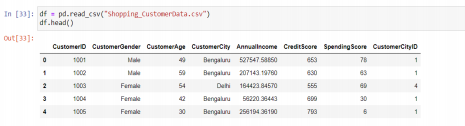
interpretations of the nature of customer value. Inconsistency pervades the terminology used, confuses the meaning of the concept, and thus its conceptual component parts.

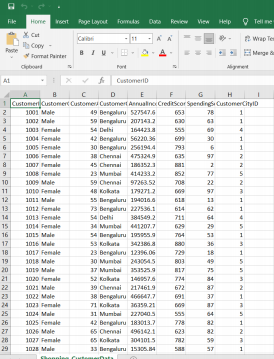
For these reasons, the objective of this article is to develop an integrative framework that clarifies the confusion surrounding this very important concept. Accordingly, we analyze the variety of terms and meanings found in the literature. We also classify and provide in-depth commentary on the conceptual approaches available, and identify a series of common and divergent elements among the various definitions. As a consequence of this review, a conceptual framework on consumer value is outlined, the main characterizing features of this construct are highlighted, and we propose a global definition. Finally, conclusions are drawn and future research directions are discussed.

**METHODOLOGY**

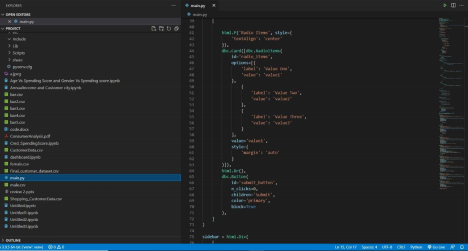
The following dataset have been taken for the project:

Shopping\_Customer.csv in notebook

Shopping\_Customer.csv in excel sheet



We have used the pycharm IDE and set up the necessary environment for the project. The project diary is shown below.



• Django is a framework that connects the front end with the back end. we have used the Django framework in pycharm IDE

• The libraries imported for the project are pandas, numpy, matplotlib and seaborn. Matplotlib and seaborn have been imported to visualize the boxplots. SciPy is a collection of open-source code libraries for maths, science and engineering. NumPy, pandas and Matplotlib are libraries that fall under this umbrella of SciPy.

• Pandas is an open-source Python library providing efficient, easy to use data structure and data analysis tools.

• Matplotlib is a python library that is specifically designers for the development of graphs, charts in order to provide interactive data visualization. It is inspired from MATLAB software and reproduces many of its features.

• The front end has been developed using html, css and JavaScript. • Google charts has been used on the front end for better visualization.

**SOURCE CODE:**

**BACKEND:**

import dash

import pandas

import dash\_bootstrap\_components as dbc import dash\_core\_components as dcc

import dash\_html\_components as html

from dash.dependencies import Input, Output, State import plotly.express as px

# the style arguments for the sidebar. SIDEBAR\_STYLE = {

'position': 'fixed',

'top': 0,

'left': 0,

'bottom': 0,

'width': '20%',

'padding': '20px 10px',

'background-color': '#f8f9fa'

}

# the style arguments for the main content page. CONTENT\_STYLE = {

'margin-left': '25%',

'margin-right': '5%',

'padding': '20px 10p'

}

TEXT\_STYLE = {

'textAlign': 'center',

'color': '#191970'

}

CARD\_TEXT\_STYLE = {

'textAlign': 'center',

'color': '#0074D9'

}

controls = dbc.FormGroup(

[

html.P('Radio Items', style={ 'textAlign': 'center'

}),

dbc.Card([dbc.RadioItems(

id='radio\_items',

options=[{

'label': 'Value One', 'value': 'value1'

},

{

'label': 'Value Two', 'value': 'value2' },

{

'label': 'Value Three', 'value': 'value3' }

],

value='value1',

style={

'margin': 'auto'

}

)]),

html.Br(),

dbc.Button(

id='submit\_button',

n\_clicks=0,

children='Submit',

color='primary',

block=True

),

]

)

sidebar = html.Div(

[

html.H2('Graph types', style=TEXT\_STYLE), html.Hr(),

controls

],

style=SIDEBAR\_STYLE,

)

content\_first\_row = dbc.Row([

dbc.Col(

dbc.Card(

[

dbc.CardBody(

[

html.H4(id='card\_title\_1', children=['GAURAV'], className ='card-title',

style=CARD\_TEXT\_STYLE),

]

)

]

),

md=3

),

dbc.Col(

dbc.Card(

[

dbc.CardBody(

[

html.H4('UDBHAV', className='card

title', style=CARD\_TEXT\_STYLE),

]

),

]

),

md=3

),

dbc.Col(

dbc.Card(

[

dbc.CardBody(

[

html.H4('HARSHINI', className='card

title', style=CARD\_TEXT\_STYLE),

]

),

]

),

md=3

),

dbc.Col(

dbc.Card(

[

dbc.CardBody(

[

html.H4('ARYA', className='card title', style=CARD\_TEXT\_STYLE),

]

),

]

),

md=3

)

])

content\_second\_row = dbc.Row(

[

dbc.Col(

dcc.Graph(id='graph\_1'), md=4

),

dbc.Col(

dcc.Graph(id='graph\_2'), md=4

),

dbc.Col(

dcc.Graph(id='graph\_3'), md=4

)

]

)

content\_third\_row = dbc.Row(

[

dbc.Col(

dcc.Graph(id='graph\_4'), md=12, )

]

)

content\_fourth\_row = dbc.Row(

[

dbc.Col(

dcc.Graph(id='graph\_5'), md=6

),

dbc.Col(

dcc.Graph(id='graph\_6'), md=6

)

]

)

content = html.Div(

[

html.H2('CUSTOMER SEGMENTATION', style=TEXT\_STYLE),

html.Hr(),

content\_first\_row,

content\_second\_row,

content\_third\_row,

content\_fourth\_row

],

style=CONTENT\_STYLE

)

app = dash.Dash(external\_stylesheets=[dbc.themes.BOOTSTRAP])

app.layout = html.Div([sidebar, content])

@app.callback(

Output('graph\_1', 'figure'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_graph\_1(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_value , radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value)

fig = {

'data': [{

'x': [1, 2, 3],

'y': [3, 4, 5]

}]

}

return fig

@app.callback(

Output('graph\_2', 'figure'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_graph\_2(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_value , radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value)

fig = {

'data': [{

'x': [1, 2, 3],

'y': [3, 4, 5],

'type': 'bar'

}]

}

return fig

@app.callback(

Output('graph\_3', 'figure'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_graph\_3(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_value , radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value)

df = px.data.iris()

fig = px.density\_contour(df, x='sepal\_width', y='sepal\_length') return fig

@app.callback(

Output('graph\_4', 'figure'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_graph\_4(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_value , radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value) # Sample data and figure

df = px.data.gapminder().query('year==2007')

fig = px.scatter\_geo(df, locations='iso\_alpha', color='continent', hover\_name='country', size='pop', projection='natural ea rth')

fig.update\_layout({

'height': 600

})

return fig

@app.callback(

Output('graph\_5', 'figure'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_graph\_5(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_value , radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value) # Sample data and figure

df = px.data.iris()

fig = px.scatter(df, x='sepal\_width', y='sepal\_length')

return fig

@app.callback(

Output('graph\_6', 'figure'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_graph\_6(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_value , radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value) # Sample data and figure

df = px.data.tips()

fig = px.bar(df, x='total\_bill', y='day', orientation='h') return fig

@app.callback(

Output('card\_title\_1', 'children'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_card\_title\_1(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_ value, radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value) # Sample data and figure

return 'Card Tile 1 change by call back'

@app.callback(

Output('card\_text\_1', 'children'),

[Input('submit\_button', 'n\_clicks')],

[State('dropdown', 'value'), State('range\_slider', 'value'), State('check\_lis t', 'value'),

State('radio\_items', 'value')

])

def update\_card\_text\_1(n\_clicks, dropdown\_value, range\_slider\_value, check\_list\_v alue, radio\_items\_value):

print(n\_clicks)

print(dropdown\_value)

print(range\_slider\_value)

print(check\_list\_value)

print(radio\_items\_value) # Sample data and figure

return 'Card text change by call back'

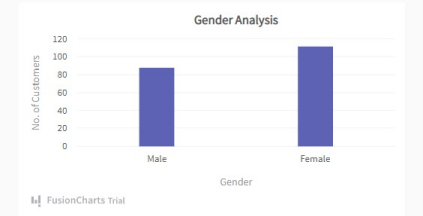
if \_\_name\_\_ == '\_\_main\_\_':

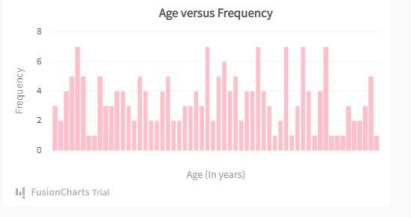
app.run\_server(port = 8080)

RESULTS:



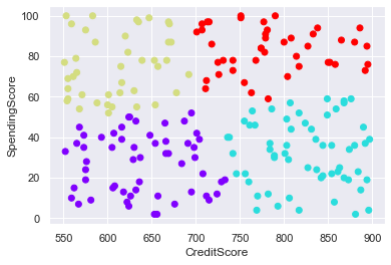
Project Details from Dashboard:



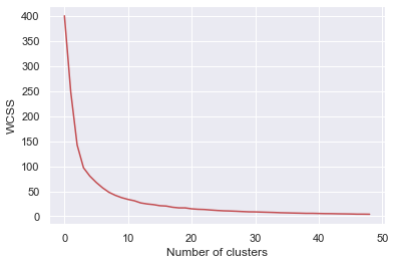


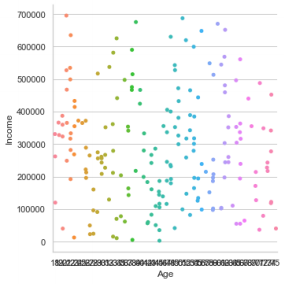
Other important graphs done in Jupyter Notebook:

Spending Score vs Credit Score:

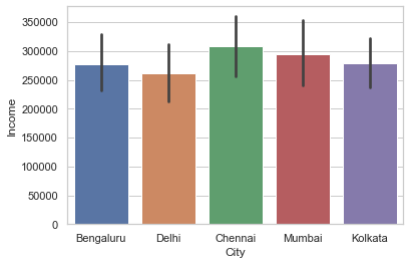


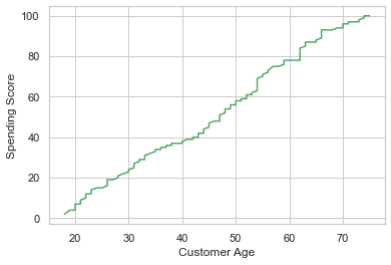
No. Of Clusters vs WWCS:

Age vs Annual Income:

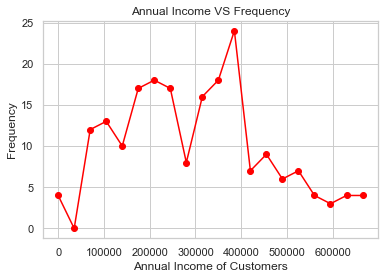


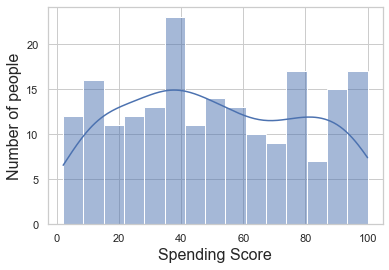
City vs Annual Income:

Spending Score vs Age:

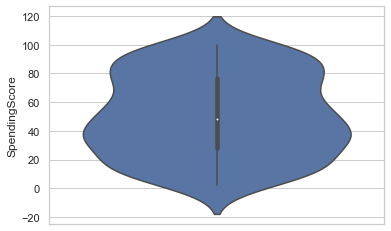


Annual Income vs Frequency:

Number of People vs Spending Score:



Spending Score Analysis:



Project By:



CONCLUSION:

The project has been implemented according to the understanding and requirements of information visualization using the necessary coding, and software and the output is displayed.

REFERENCES:

• R. E. Ziemer and W. H. Tranter, *Principles of Communications: Systems, Modulation*, *and Noise*, 7th ed. Hoboken, NJ: Wiley, 2015.

• J. D. Bellamy *et al.*, *Computer Telephony Integration*, New York: Wiley, 2010. • C. Jacks, *High Rupturing Capacity (HRC) Fuses*, New York: Penguin Random House, 2013, pp. 175–225.

• N. B. Vargafik, J. A. Wiebelt, and J. F. Malloy, "Radiative transfer," in *Convective Heat*. Melbourne: Engineering Education Australia, 2011, ch. 9, pp. 379–398.

• H. C. Hottel and R. Siegel, "Film condensation," in *Handbook of Heat Transfer,* 2nd ed*.* W. C. McAdams, Ed. New York: McGraw-Hill, 2011, ch. 9, pp. 78–99.

• W. M. Rohsenow, "Heat transmission," in *Thermal Radiation Properties,* vol. 3, M. W. Catton and J. P. Hartnett, Eds. New York: Macmillan, 2012, ch. 9, pp. 37–62.