Shan Super Market

Data Analysis Report

Prepared by

Zeeshan Ali

SM. Raahim Taqvi

Saqib Sayab Khan

alizeeshan001@hotmail.com

December 28 2020

Introduction

Data science is a multifaceted field used to gain insights from complex data. The aim of this thesis is to discover how to analyses data using Python and Excel with the different data sets. The proposal of this thesis to analyses datasets of Shan super mart by using Python´s libraries also this report analyses the excel datasets and illustrate the comparative study between them. The data analytics is the investigation of analyzing unrefined data to resolve on selections about the information where data science is an interdisciplinary field that utilizes logical techniques, procedures, calculations and frameworks to separate information and bits of knowledge from organized and unstructured information which is identified with data mining and big data.

This report comprises analysis, data analysis, normalization, formatting, binning and exploratory data analysis. Additionally, practical part exemplifies the data analysis of super mart that how it improved it's rating among all branches which is completed by Python based data analysis beside this it contains Excel based data analysis of excel datasheets and comparative study of two data sets by using two types of data analysis: Python-based data analysis and Excel-based data analysis. Finally, this report concludes the preeminent data analysis method by describing its features and recommend for individuals who want to develop career in data science as well as refers to future of data analysis.

Key Words:

Pandas, Matplotlib, Seaborn, CSV, OS, NumPy,

Concept Definition:

NumPy: Numeric Python

CSA: comma-separated values

Matplotlib: Plotting library for python

OS:

Seaborn: Python data visualization library based on matplotlib, it provides a high-level interface for drawing attractive and informative statistical graphs.

Pandas: For Data Manipulation

- ## *Data Science & It’s Relation to Statistics*

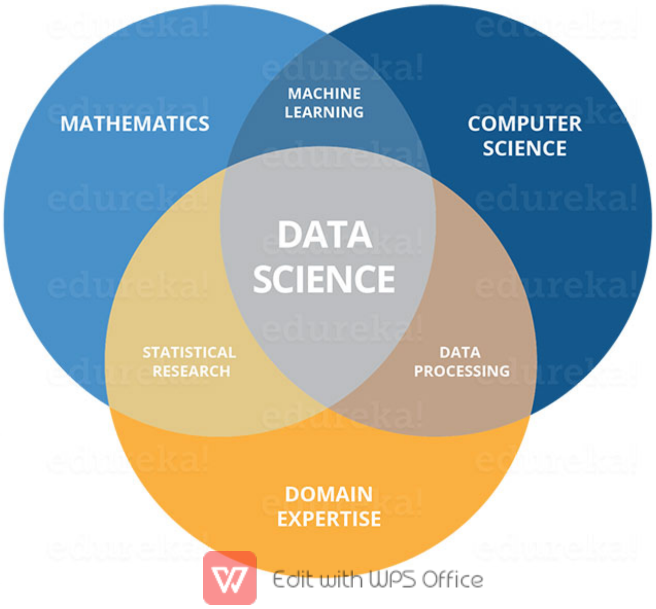
Data science is an interdisciplinary field focused on extracting knowledge from data sets, which are typically large (see big data). The field encompasses analysis, preparing data for analysis, and presenting findings to inform high-level decisions in an organization. As such, it incorporates skills from computer science, mathematics, statistics, information visualization, graphic design, complex systems, communication and business. Statistician Nathan Yau, drawing on Ben Fry, also links data science to human-computer interaction: users should be able to intuitively control and explore data. In 2015, the American Statistical Association identified database management, statistics and machine learning, and distributed and parallel systems as the three emerging foundational professional communities.

❖ *Relationship to statistic*

Many statisticians, including Nate Silver, have argued that data science is not a new field, but rather another name for statistics. Others argue that data science is distinct from statistics because it focuses on problems and techniques unique to digital data. Vasant Dhar writes that statistics emphasizes quantitative data and description. In contrast, data science deals with quantitative and qualitative data (e.g. images) and emphasizes prediction and action. Andrew Gelman of Columbia University and data scientist Vincent Granville have described statistics as a nonessential part of data science. Stanford professor David Donoho writes that data science is not distinguished from statistics by the size of datasets or use of computing, and that many graduate programs misleadingly advertise their analytics and statistics training as the essence of a data science program. He describes data science as an applied field growing out of traditional statistics, in summary, data science can be therefore described as an applied branch of statistics.

❖ *Importance of Data Science*

Data is one of the important features of every organization because it helps business leaders to make decisions based on facts, statistical numbers and trends. Due to this growing scope of data, data science came into picture which is a multidisciplinary field. It uses scientific approaches, procedure, algorithms, and framework to extract the knowledge and insight



from a huge amount of data. The extracted data can be either structured or unstructured. Data science is a concept to bring together ideas, data examination, Machine Learning, and their related strategies to comprehend and dissect genuine phenomena with data. Data science is an extension of various data analysis fields such as data mining, statistics, predictive analysis and many more. Data Science is a huge field that uses a lot of methods and concepts which belongs to other fields like information science, statistics, mathematics, and computer science. Some of the techniques utilized in Data Science encompasses machine learning, visualization, pattern recognition, probability model, data engineering, signal processing, etc.

The developments of plenty of data have given enormous importance to many features of Data science particularly big data. But data science is not limited to big data alone as big data solutions concentrated more on organizing and pre-processing the data instead of analyzing them. Also, due to Machine Learning, the importance and growth of data science has been improved.

❖ *Role of Python in Data Science*

Python is a flexible programming language that is primarily favoured by software engineers and tech organizations around the globe, from new businesses to the bigger ones. Data Scientists use it widely for data analysis and knowledge generation, while many organizations pick it for its convenience, extensibility, coherence, transparency, and the completeness of its standard library.

Python programming skills are in great demand and learning it can open ways to endless opportunities in Data Science, Machine Learning, Artificial Intelligence, web development, and much more. The Python for Data Science is applied to-

- Candidates with no background of coding or want to explore their career in Data Science
- Candidates who are currently or planning to enroll in Data Science field
- Candidates who are currently or planning to enroll in Machine Learning industry
- Candidates who are currently or planning to enroll in Artificial

Intelligence industry

Python for Data Science is used to build strong skills in the basic concepts that are mandatory for Data Science. It includes data handling, feature engineering, statistical analysis, and Python programming. When you start building for the foundation in Python for Data Science, you build and implement predictive analytics algorithms (regression and forecasting), and classification and segmentation Machine Learning algorithms (K Means and Random Forest) using Python. Python with Data Science allows flexibility and makes integration of programming easy with complex systems. This is all because of the Python programming which is very dynamic and portable and due to this, the use of Python is wide in the industries.

Understanding the use of Python in Data Science

Consistently, around the United States, in excess of 36,000 climate estimates are given covering 800 unique locales and urban areas. If it starts raining in the middle of your outing but it was assumed to be a sunny day! Such forecasts around the country predict the usage of forecast models. The people at Forecastwatch.com did! Consistently, they accumulate every one of the 36,000 estimates, put them in a database, and compare them with the actual conditions experienced in that area on that day. Forecasters around the country at that point utilize the outcomes to improve their forecast models for the next round. All this collection of data, analysis, and reporting takes a lot of time but Forecastwatch.com does it with the use of a single programming language: *Python*.

Not only Forecastwatch.com but according to the industry analyst survey of O' Reilly, 40% of the data scientists make use of Python for their day-to-day work. *Trusted organizations such as Google, NASA, and CERN use Python for almost every programming purpose under the sun… including, in increasing measures of data science.*

Python: the new life of Data Science

Algorithms or predictions made in Data Science are very tricky and mind-boggling. If the programming language will also be tricky then a data scientist would use his maximum time in coding and syntax only. Even Java can be used for Data Science. But if the comparison between two programming Languages is done then Python will be a winner. Because of the small code snippets, Python is the most loved language for Data Science and this is the primary reason for utilizing Python for Data Science.

❖ *IDE that I used to analyse data(Jupyter Notebook)*

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning.

1 Libraries That I used in My Project

i. NumPy

In Python programming language, NumPy is a library containing assistance for with a massive assortment of higher-level numerical capabilities to operate on these constellations and multi-dimensional exhibitions and networks. The predecessor of NumPy, Numeric, was initially produced by Jim Hugunin with commitments from a few different designers. In 2005, Travis Oliphant made NumPy by consolidating highlights of the contending Num-array into Numeric with wide changes. NumPy is open-source programming language and has numerous benefactors NumPy (Numerical Python) is the basic bundle for numerical calculation in Python; it contains a groundbreaking N-dimensional cluster object. It has around 18,000 remarks on GitHub and a functioning network of 700 givers. It is a universally useful cluster preparing bundle that gives superior multidimensional articles called exhibits and devices for working with them. NumPy likewise addresses the gradualness issue halfway by giving these multidimensional exhibits just as giving capacities and administrators that work proficiently on these clusters. The main features of the NumPy is to give quick, precompiled capacities for numerical schedules. Similarly, it exhibits situated registering for better proficiency along with it establishes an article situated methodology and conservative and quicker calculations with factorization.

ii. Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged SciPy makes use of Matplotlib.Matplotlib was originally written by John D. Hunter, since then it has an active development community,and is distributed under a BSD-style license. Michael Droettboom was nominated as matplotlib's lead developer shortly before John Hunter's death in August 2012, and further joined by Thomas Caswell

iii. Sea-born

Sea born is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Sea born helps you explore and understand your data. Its plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them. Sea born is the only library we need to import for this simple example. By convention, it is imported with the shorthand sns. Behind the scenes, sea born uses matplotlib to draw its plots. For interactive work, it’s recommended to use a Jupyter/IPython.

iv. Pandas

In computer programming, for data control and examination in Python programming language a product library composes which is known as Pandas. Specifically, it extends data structure and activities for regulating mathematical tables and time arrangement. It is free programming language released under the three-provision BSD permit. The name is grown from the expression "panel data", an econometrics term for informational collections that incorporate perceptions over various timeframes for similar people. Pandas (Python data analysis) is an indisputable requirement in the Data Science life cycle. It is the most famous and broadly utilized Python library for Data Science, alongside NumPy in matplotlib. With around 17, 00 remarks on GitHub and a functioning network of 1,200 benefactors, it is actively utilized for data analysis and cleaning. Pandas give quick, adaptable information structures, for example, information outline CDs, which are intended to work with organized data rapidly and instinctively.

v. OS Module

The OS module in python provides functions for interacting with the operating system. OS, comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality. The *os* and *os.path* modules include many functions to interact with the file system.

2. Some Basic Steps to Analyze Data

➤ Processing Data

A data mining method that involves editing raw data within a reasonable range is known as data preprocessing. In particularly, by conflicting and painful practices or deviations and verifiable data is often insufficient and may contain many errors. Data preprocessing is an effective strategy to solve these problems. Data preprocessing can prepare raw data for other management. Database-based applications (such as customer relationships, executives, and rules-based applications (such as nervous systems) use data preprocessing. Data preprocessing is a significant advance in the data mining process. The expression "trash in, trash out" is especially relevant to data mining and AI ventures. Data gathering strategies are regularly almost controlled, arriving in out-of-extend values (e.g., Income: -100), unthinkable information blends (e.g., Sex:

Male, Pregnant: Yes) and missing qualities. Examining data that has not been deliberately screened for such issues can create deceiving results. In this way, the portrayal and nature of data are as a matter of first importance before running an examination. Regularly, data preprocessing is the most significant period of an AI project, particularly in computational science.

➤ *Missing Values*

Most datasets contain missing data, mistakenly encoded data, or other data that cannot be utilized for demonstrating. Occasionally missing data is only that — missing. There is no genuine incentive in each field, for instance, an unfilled string in a CSV record. Different occasion's data is encoded with an extraordinary watchword or a string. Some basic encodings are NA, N/A, None, and - 1 Before utilizing data with missing data fields, it must change those fields so they can be utilized for examination and display. There are AI calculations and bundles that can naturally distinguish and manage missing data, however, it is however a reasonable practice to change that data physically. (Tunguz 2018.) In the same way as other different parts of Data Science, there is a considerable several craftsmanship and expertise associated with how to manage missing data, and Data Science may really be more workmanship than science. Understanding the data and the area from which it comes from is significant. For example, by computing mean bodes well for specific highlights and areas than for other people. Having missing qualities in data is not really a mishap. Essentially, as a rule one can gather a ton of helpful data from missing qualities, and they can be utilized for the reasons for highlight designing. Individual must be cautious, however: on the off chance that there is an element in the dataset that has a truly elevated level of missing qualities. At that point which element much the same as some other low changes include, ought to be dropped.

➤ *Data Normalization*

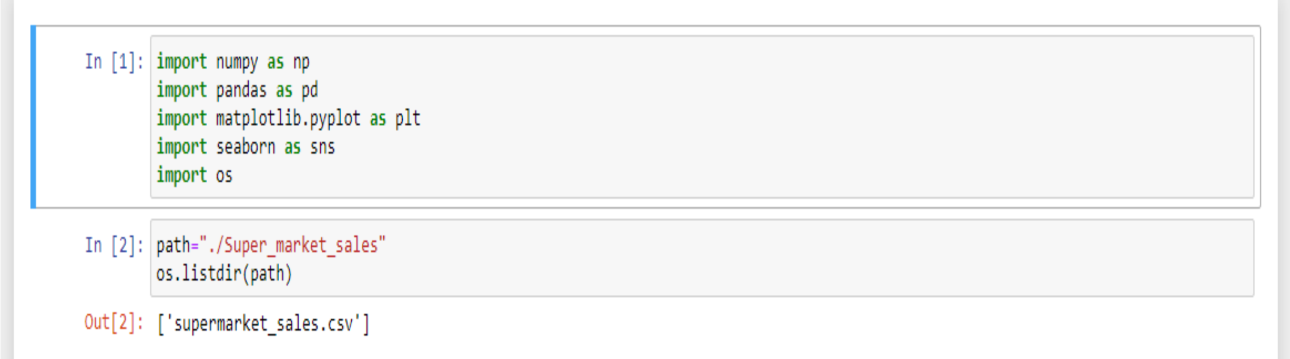
The definition of data normalization is not straightforward although selecting one might be a little unstable. With all the various descriptions in mind, data normalization is basically a process where the data in the database is updated so that the customer can use the database appropriately for further questions and inquiries. When embarking on a data standardization process, there are some goals are at top priority. The first is to delete all replica data that can be viewed in the information index. This tests the database and eliminates any redundancy that may occur. Also, downsizing can adversely affect data analysis because that constitute unnecessary values. By removing from the database helps to reorder the data for inspection. Correspondingly, another goal is to accumulate data intelligently that requires the data that can identify each other that will happen in a data-standardized database. If the data have been linked, it should be close to the information index.

Sometimes the dataset will contain competing data, hence data normalization aims to resolve this conflict and reveal it before continuing. The third step is to organize the data. This creates the data and becomes it a protocol that can be done more preparation and analysis. Finally, data normalization will consolidate information and merge it into a substantially more ordered structure.

➤ *Analysing Data for Shaan Super Market*

Step 01

In the first I import libraries for analyse data and then I'll open the selected data file.

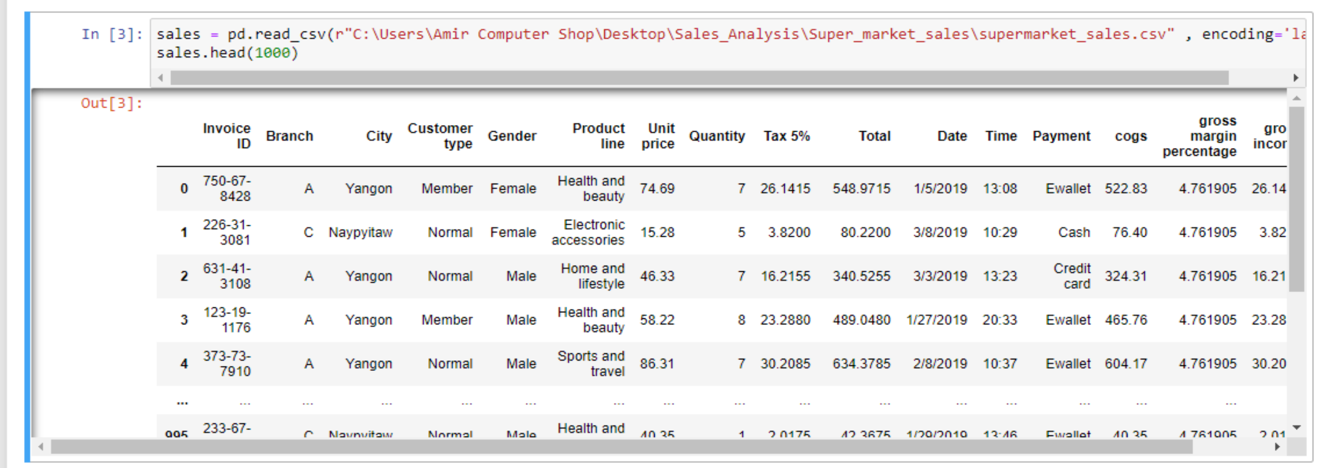


```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        import os

In [2]: path="./Super_market_sales"
        os.listdir(path)

Out[2]: ['supermarket_sales.csv']
```

These are the libraries that I'll work on and this is the code from I opened the file in my jupyter notebook.

Step 02

Now ill read my file and find the NaN Values from the file and delete those values from my file.

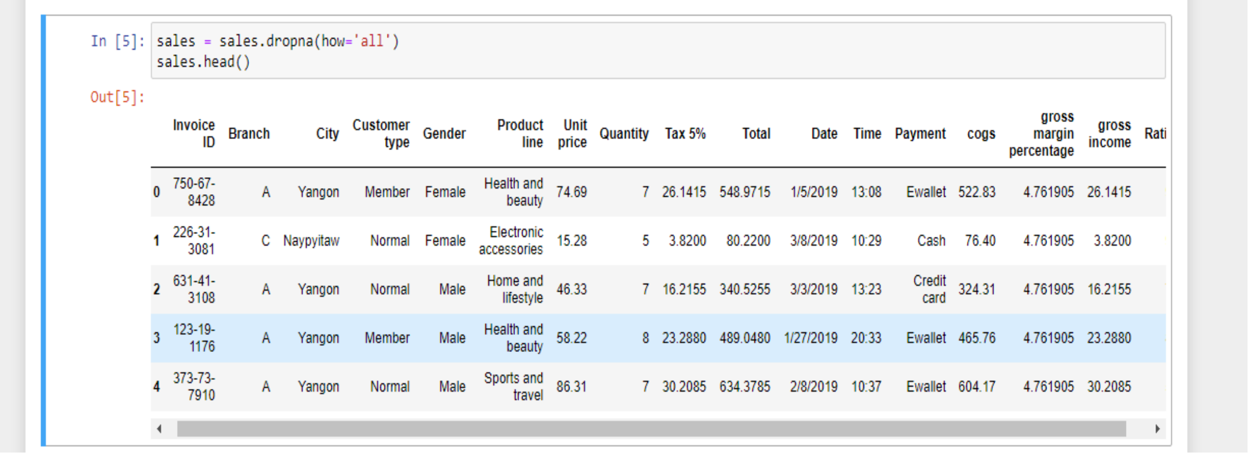


```
In [3]: sales = pd.read_csv(r"C:\Users\Amir Computer Shop\Desktop\Sales_Analysis\Super_market_sales\supermarket_sales.csv" , encoding='la
        sales.head(1000)
```

Out[3]:

	Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity	Tax 5%	Total	Date	Time	Payment	cogs	gross margin percentage	gro incor
0	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7	26.1415	548.9715	1/5/2019	13:08	Ewallet	522.83	4.761905	26.14
1	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5	3.8200	80.2200	3/8/2019	10:29	Cash	76.40	4.761905	3.82
2	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2155	340.5255	3/3/2019	13:23	Credit card	324.31	4.761905	16.21
3	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8	23.2880	489.0480	1/27/2019	20:33	Ewallet	465.76	4.761905	23.28
4	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7	30.2085	634.3785	2/8/2019	10:37	Ewallet	604.17	4.761905	30.20
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
995	233-67-	C	Naypyitaw	Normal	Male	Health and	40.35	1	2.0175	42.3675	1/29/2019	13:46	Ewallet	40.35	4.761905	2.01

Now I'll find Nan values and remove.



```
In [5]: sales = sales.dropna(how='all')
        sales.head()
```

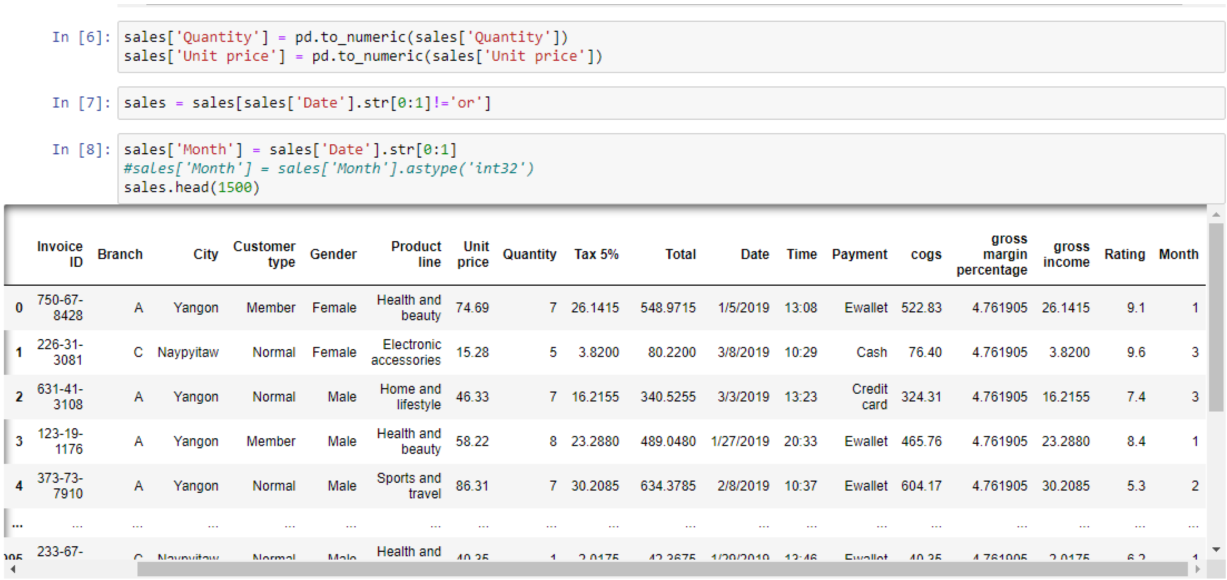
Out[5]:

	Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity	Tax 5%	Total	Date	Time	Payment	cogs	gross margin percentage	gross income	Rati
0	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7	26.1415	548.9715	1/5/2019	13:08	Ewallet	522.83	4.761905	26.1415	
1	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5	3.8200	80.2200	3/8/2019	10:29	Cash	76.40	4.761905	3.8200	
2	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2155	340.5255	3/3/2019	13:23	Credit card	324.31	4.761905	16.2155	
3	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8	23.2880	489.0480	1/27/2019	20:33	Ewallet	465.76	4.761905	23.2880	
4	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7	30.2085	634.3785	2/8/2019	10:37	Ewallet	604.17	4.761905	30.2085	

This file has no NaN values and also showing nothing so my file is clear now for analyzing.

STEP 03

Now I'll convert "Quantity" and "Unit Price" columns into Numeric to find Sales and also make a separate column for month to find in which month stare has sold the most items.



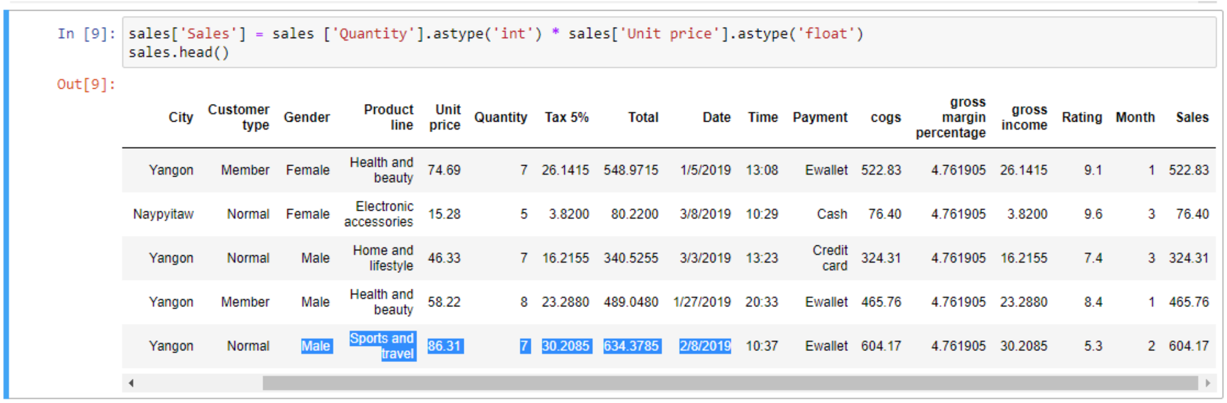
```
In [6]: sales['Quantity'] = pd.to_numeric(sales['Quantity'])
        sales['Unit price'] = pd.to_numeric(sales['Unit price'])

In [7]: sales = sales[sales['Date'].str[0:1]!='or']

In [8]: sales['Month'] = sales['Date'].str[0:1]
        #sales['Month'] = sales['Month'].astype('int32')
        sales.head(1500)
```

	Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity	Tax 5%	Total	Date	Time	Payment	cogs	gross margin percentage	gross income	Rating	Month
0	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7	26.1415	548.9715	1/5/2019	13:08	Ewallet	522.83	4.761905	26.1415	9.1	1
1	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5	3.8200	80.2200	3/8/2019	10:29	Cash	76.40	4.761905	3.8200	9.6	3
2	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2155	340.5255	3/3/2019	13:23	Credit card	324.31	4.761905	16.2155	7.4	3
3	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8	23.2880	489.0480	1/27/2019	20:33	Ewallet	465.76	4.761905	23.2880	8.4	1
4	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7	30.2085	634.3785	2/8/2019	10:37	Ewallet	604.17	4.761905	30.2085	5.3	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	233-67-	C	Naypyitaw	Normal	Male	Health and	40.35	1	2.0175	42.3675	1/29/2019	13:46	Ewallet	40.35	4.761905	2.0175	6.2	1

In this figure I've made separated column for month and two columns converted into numeric for find sales.

STEP 04

In this step I'll find sales and make another separate column for sales.



```
In [9]: sales['Sales'] = sales ['Quantity'].astype('int') * sales['Unit price'].astype('float')
        sales.head()
```

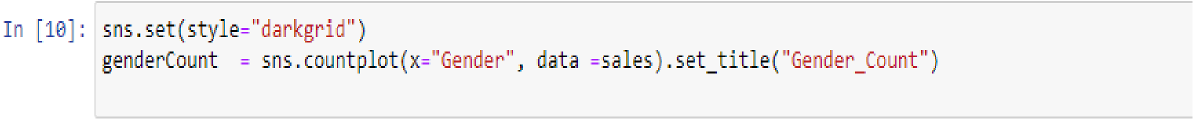
Out[9]:

City	Customer type	Gender	Product line	Unit price	Quantity	Tax 5%	Total	Date	Time	Payment	cogs	gross margin percentage	gross income	Rating	Month	Sales
Yangon	Member	Female	Health and beauty	74.69	7	26.1415	548.9715	1/5/2019	13:08	Ewallet	522.83	4.761905	26.1415	9.1	1	522.83
Naypyitaw	Normal	Female	Electronic accessories	15.28	5	3.8200	80.2200	3/8/2019	10:29	Cash	76.40	4.761905	3.8200	9.6	3	76.40
Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2155	340.5255	3/3/2019	13:23	Credit card	324.31	4.761905	16.2155	7.4	3	324.31
Yangon	Member	Male	Health and beauty	58.22	8	23.2880	489.0480	1/27/2019	20:33	Ewallet	465.76	4.761905	23.2880	8.4	1	465.76
Yangon	Normal	Male	Sports and travel	86.31	7	30.2085	634.3785	2/8/2019	10:37	Ewallet	604.17	4.761905	30.2085	5.3	2	604.17

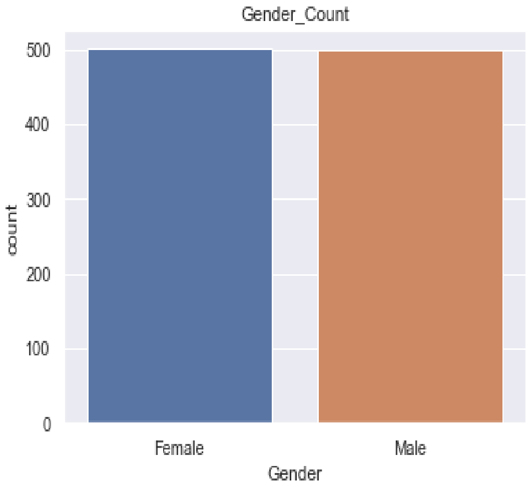
This figure shows the calculation of all months sales and it's separate column as well.

STEP 05

Now I'll find the gender of customers graphically for checking that which gander's products has sold most from the store then we'll increase the production and give discount as well on the products.



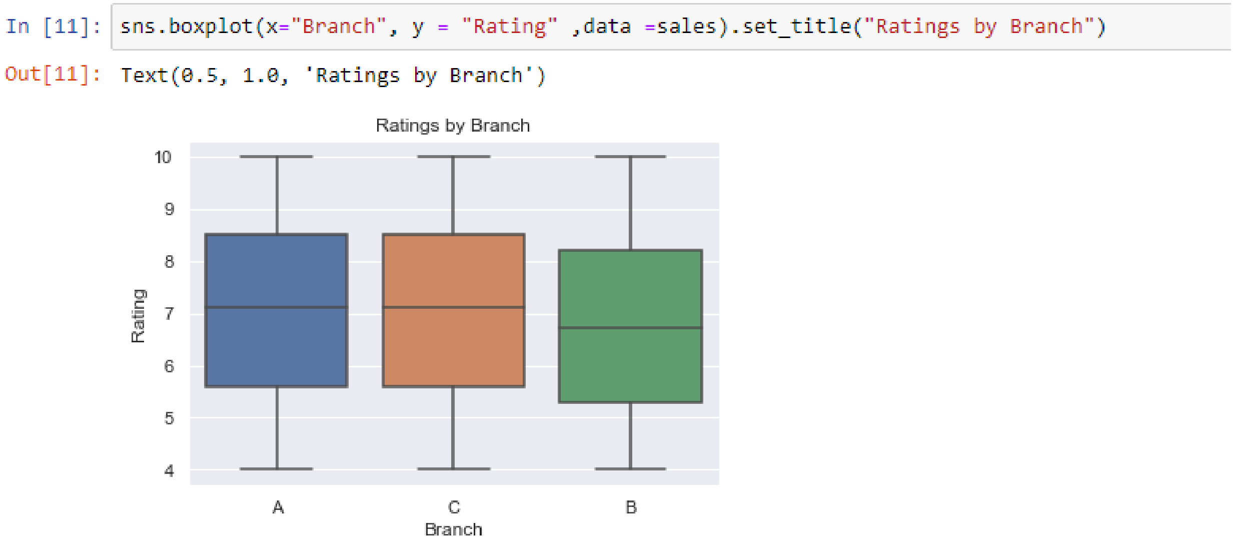
```
In [10]: sns.set(style="darkgrid")
         genderCount  = sns.countplot(x="Gender", data =sales).set_title("Gender_Count")
```



In this graph I've found the equal customers as gander, and I used sea-born library for plotting.

STEP 06

In this step we'll analyse the Rating between three different Branches for upcoming year's planning the where should we've to increase our sales and where should we have to give discounts and improve out management.

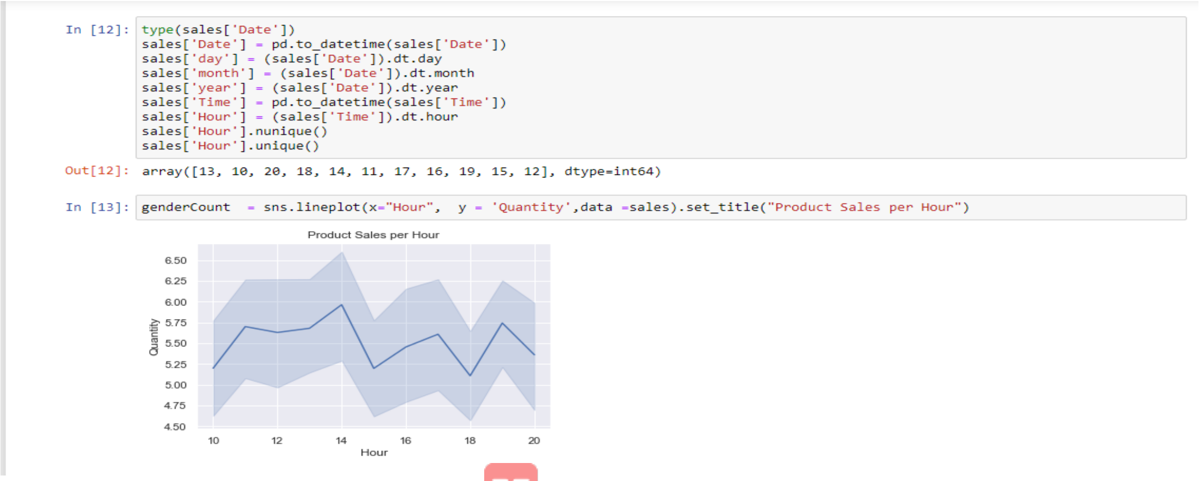


```
In [11]: sns.boxplot(x="Branch", y = "Rating" ,data =sales).set_title("Ratings by Branch")

Out[11]: Text(0.5, 1.0, 'Ratings by Branch')
```

So B branch has low rating then the other A and C Branch now we've to look that why this branch has low rating what kind of improvements needed for Branch C.

STEP 07

This is the major step, in this I'm going to find sales per hour graphically and for the all branches, and also convert my data into integer for calculation.

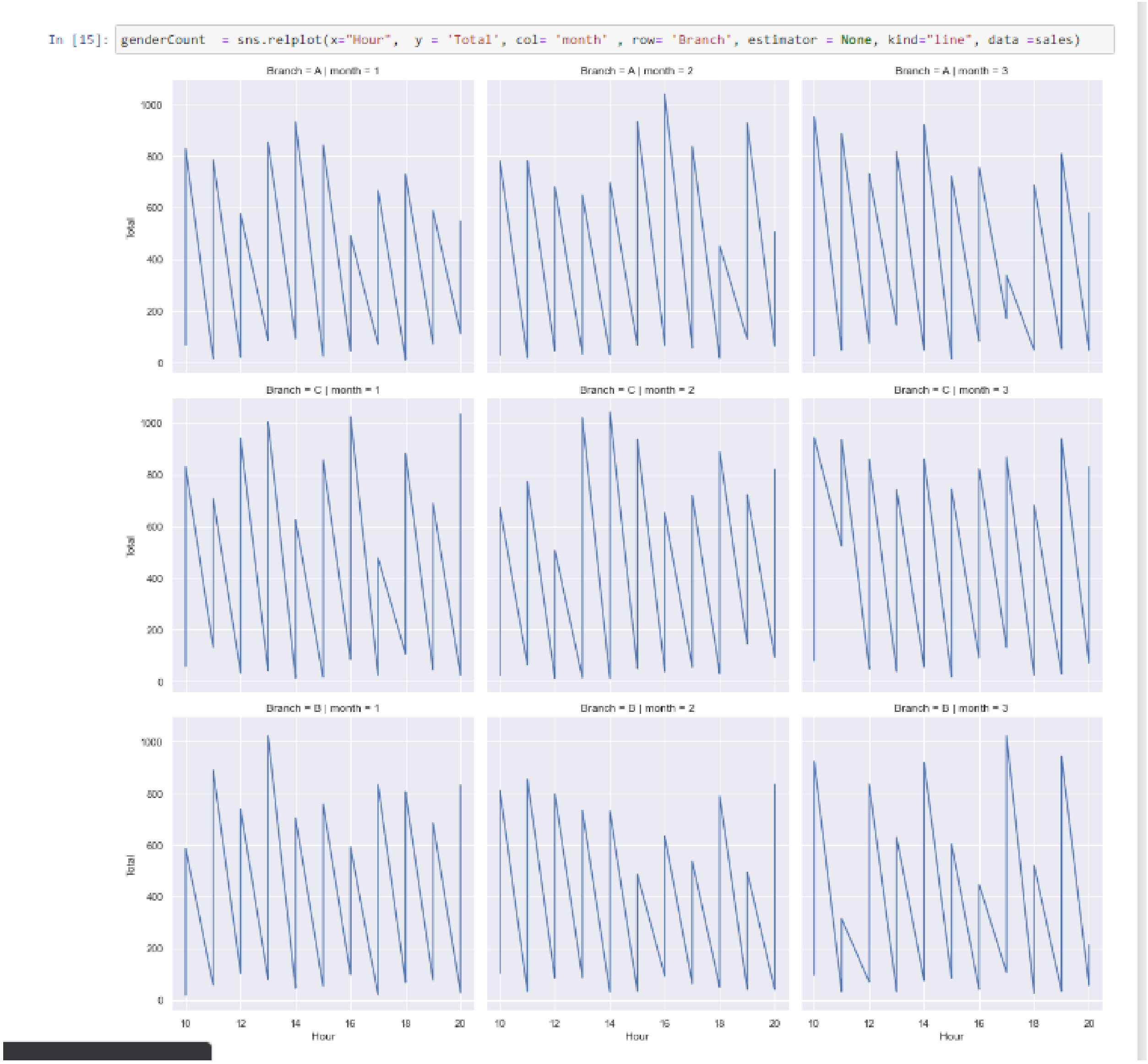


```
In [12]: type(sales['Date'])
         sales['Date'] = pd.to_datetime(sales['Date'])
         sales['day'] = (sales['Date']).dt.day
         sales['month'] = (sales['Date']).dt.month
         sales['year'] = (sales['Date']).dt.year
         sales['Time'] = pd.to_datetime(sales['Time'])
         sales['Hour'] = (sales['Time']).dt.hour
         sales['Hour'].nunique()
         sales['Hour'].unique()

Out[12]: array([13, 10, 20, 18, 14, 11, 17, 16, 19, 15, 12], dtype=int64)

In [13]: genderCount  = sns.lineplot(x="Hour",  y = 'Quantity',data =sales).set_title("Product Sales per Hour")
```

In this graph I've converted all data into integer, and market is looking the sales of these store every hour.

STEP 08

In this step I'm going to analyse the First three months sales of all branches to see where to give discount and in which branch, and also to check that from which branch we got more profit.



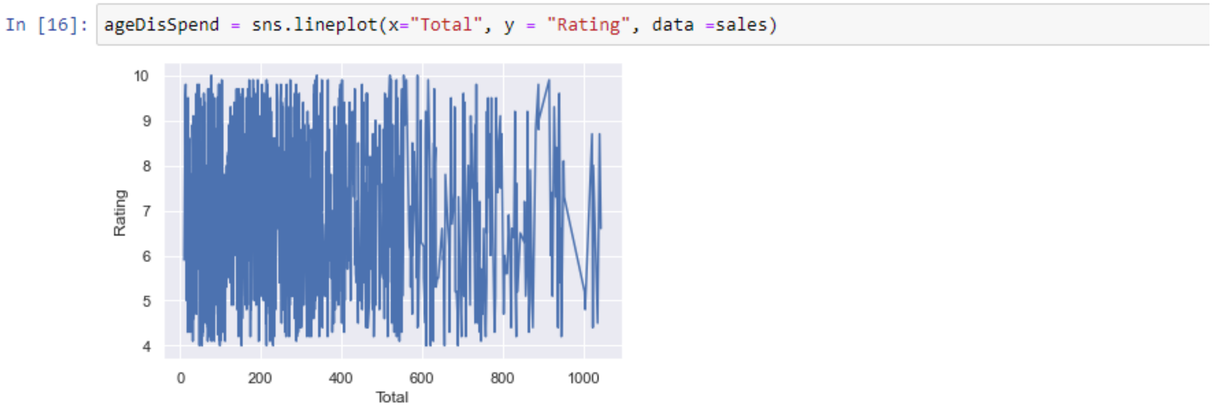
```
In [15]: genderCount  = sns.relplot(x="Hour",  y = 'Total', col= 'month' , row= 'Branch', estimator = None, kind="line", data =sales)
```

This figure shows that branch B has more sales then the other A and C branch in the 1st month of the Year.

More over in the 2nd month of the Year Branch A has most Sales, and in 3rd month branch also Branch B is on the top in Sales.

STEP 09

In this we want to see the Whole rating of the Sales from our All Branches.

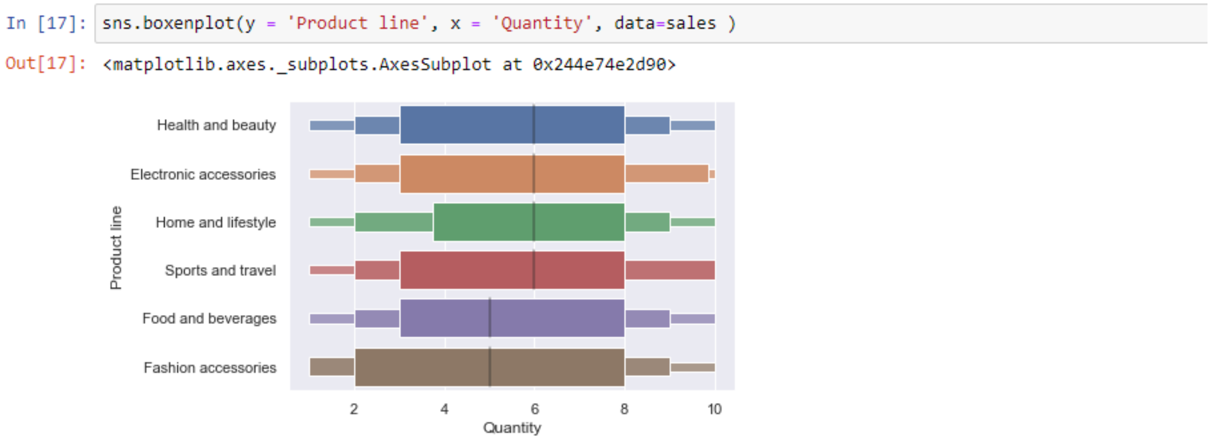


```
In [16]: ageDisSpend = sns.lineplot(x="Total", y = "Rating", data =sales)
```

This graph shows that the Rating of our Sales, and this goes to the positive result and is enough to attract to the investors in the stock market.

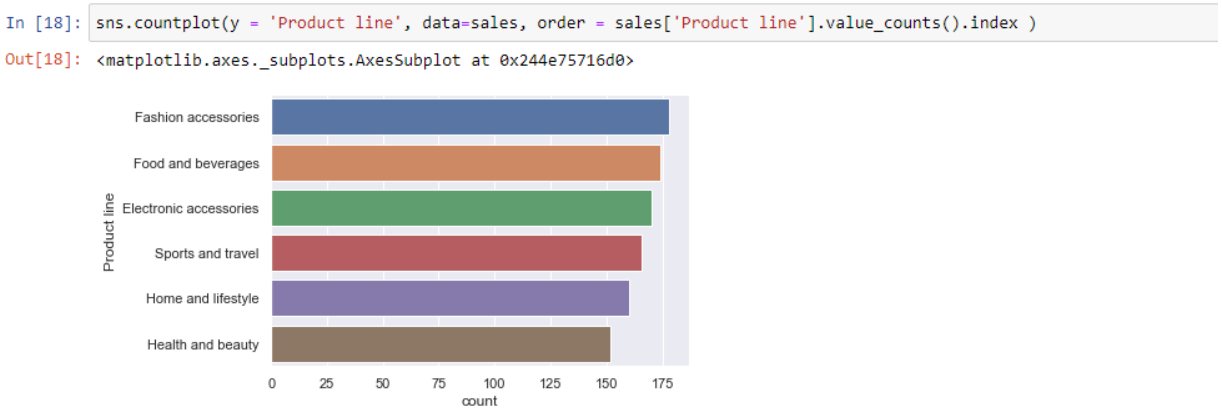
STEP 10

In this step we'll find that which kind of accessories has sold the most in the current period



```
In [17]: sns.boxenplot(y = 'Product line', x = 'Quantity', data=sales )
Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x244e74e2d90>
```

This figure shows something but not clearing the specific kind of accessories which are sold the most so we analyse vie the order we got,

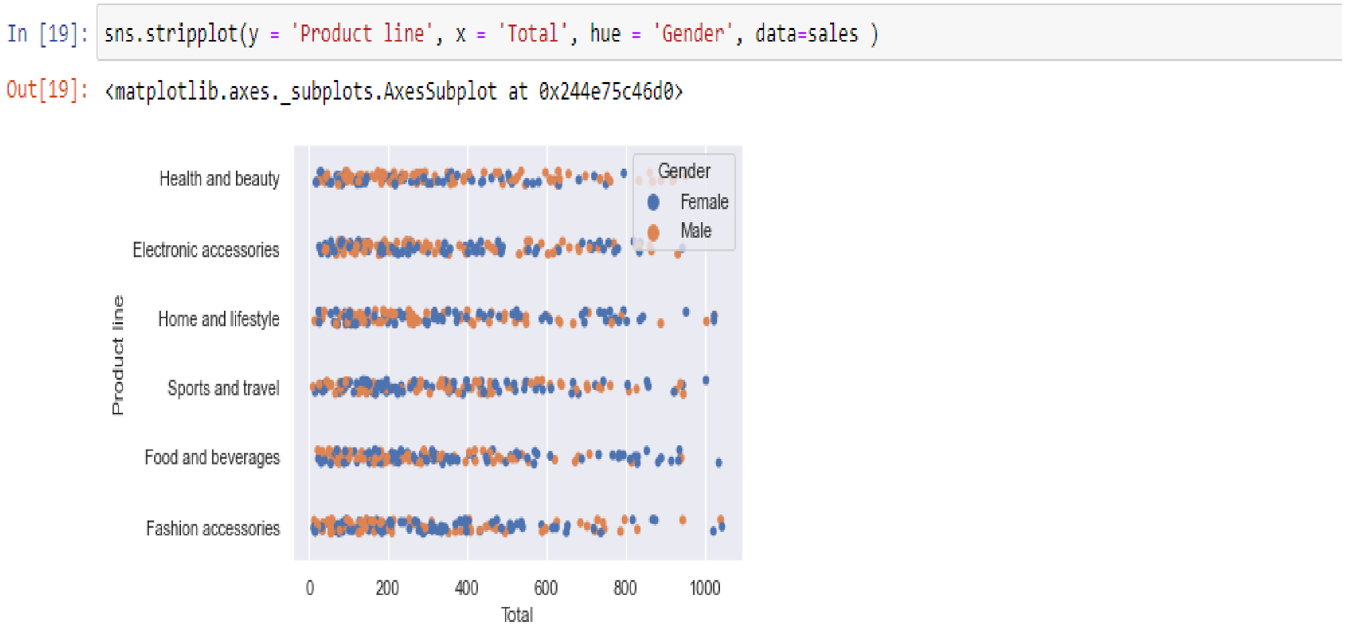


```
In [18]: sns.countplot(y = 'Product line', data=sales, order = sales['Product line'].value_counts().index )
Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x244e75716d0>
```

This figure shows clearly that Fashion accessories sold the most than the other products, so we'll give 10% OFF on these products in the coming year and also give a free little beauty box On the sales of 10000.

STEP 11

In this step we want to see that who bought out items means what gender is more attracted to our services.

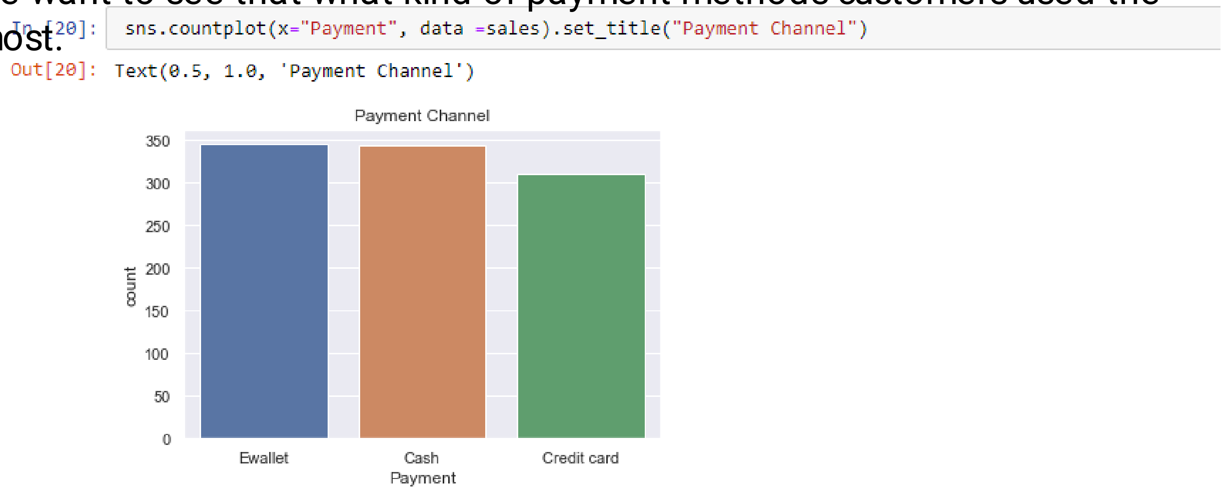


```
In [19]: sns.stripplot(y = 'Product line', x = 'Total', hue = 'Gender', data=sales )
Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x244e75c46d0>
```

This graph clearly shows that Females come to our branches than the Males, so we'll offer 5% OFF on all products to our Female Customers in coming Year.

STEP 12

In this step we want to see that how customers pay in our branches, means we want to see that what kind of payment methods customers used the most.

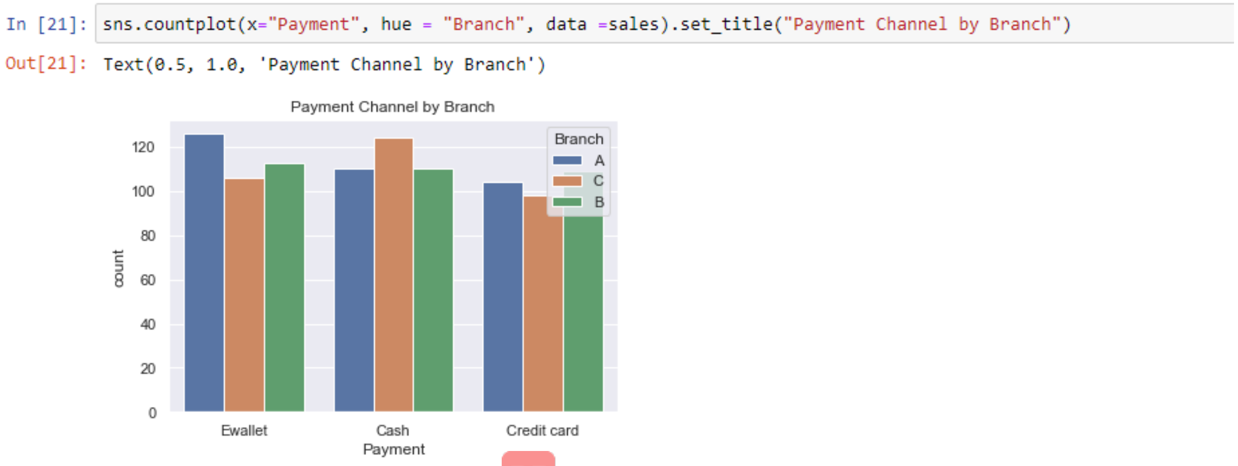


```
In [20]: sns.countplot(x="Payment", data =sales).set_title("Payment Channel")
Out[20]: Text(0.5, 1.0, 'Payment Channel')
```

This graph shows that customers used the Most Cash Payment and E-Wallet, and now we'll plan to increase Credit Payment method so we'll give 2.5% OFF on the every Payment using Credit Card.

STEP 13

This is the final step , so in this I'm going to find Payment Method in our all branches,



```
In [21]: sns.countplot(x="Payment", hue = "Branch", data =sales).set_title("Payment Channel by Branch")
Out[21]: Text(0.5, 1.0, 'Payment Channel by Branch')
```

This graph clearly shows that in our Branch A E-Wallet Used the Most, and in our Branch B Credit Card Used the most and in our Branch C Cash Payment used the most,

So i want to give some benefits to our customers by adding one Cookies Pocket free to whom who paid the Payment vie their Credit Card in our Branch C because in this Branch credit card not used at all.

And in Branch A We'll offer a Gift hamper to those who pay in Cash Payment in the coming year,

And in the last in our Branch B we also offer a mega box costs at least 12000 to those customer which used E-Wallet, and we'll select a lucky winner from them.

These are our future planning to increase our all store Rating , Sales and Marketing.

Scope

The report centralized on the Rating-benefit study on every branch which corresponding to the execution of the marketing costs and strategies from the current year or 2020.

Methodology

The utilized the data and finding that were obtained from the initial Rating-benefit Analysis report conducted 28 December 2020.