Submitted By: Manash Bhele (220433)

Analysis and prediction of Customer Clusters based on Various attributes

Using PySpark and Tableau

Softwarica College of IT and E-Commerce

STW7082CEM Big Data Management and Data Visualisation

Contents

[Introduction 2](#_Toc144469925)

[Implementation 3](#_Toc144469926)

[1. PySpark Configuration and Setup 3](#_Toc144469927)

[a. PySpark Dataframes 3](#_Toc144469928)

[b. Machine Learning (MLlib) 3](#_Toc144469929)

[2. Dataset Description and Analysis Goal 6](#_Toc144469930)

[3. Importing Necessary Libraries and Initializing Spark Context 8](#_Toc144469931)

[4. Data Preprocessing 8](#_Toc144469932)

[Discussion of Findings 9](#_Toc144469933)

[Conclusion 10](#_Toc144469934)

[References 11](#_Toc144469935)

[Appendix 12](#_Toc144469936)

# Introduction

Customers are the most important part of any businesses. How a company deals with its customers plays a huge role in determining whether the company grows or falls apart. Analysing the customer base, taking into consideration various customer attributes and patterns play a significant part in helping the company understand their customers well. The more the company understands their customers and their needs, the better the company can work on developing and delivering customer driven products which inturn helps in the growth of the company. (Hannu Kärkkäinen, 2001)

This project focuses on analysing a variety of personal and buying behaviors of customers and find out insights regarding the customers as well as predict how customers react to various campaigns led by the company. For finding out valuable insights related to customers, a Clustering analysis has been performed. Furthermore, a classification analysis has also been implemented to find out whether these attributes can be used to determine whether customer attributes can be used to predict if they accept the campaigns led by the company.

The dataset has been taken from Kaggle and all the related links has been provided below in their respective sections. For the part of analysis and prediction, tools such as Tableau and PySpark has been used. The coding has been done in Python using PySpark, which is a Python API for Apache Spark. Making use of the PySpark Mllib library, different machine learning analysis has been done on the data. Whereas the data exploration and analysis of the results using visualizations have been created in Tableau.

All the details concerning the tasks carried out, the dataset, packages used and the results attained, have been discussed in detail in their respective sections below.

# Implementation

This section breaks down the technical aspects of the project’s implementation part. First, the configuration and installation process for PySpark is discussed.

## PySpark Configuration and Setup

Apache Spark is an analytics engine which is mainly used for large-scale data processing on single node mahcines or in clusters. It is available in many languages including Scala, Java, R, Python, etc.

PySpark is a Python API for Apache Spark using Python. The main features of PySpark which is used in the project are:

### PySpark Dataframes

* PySpark Dataframes help to read, write, transform and analyze data efficiently using Python and SQL.

### Machine Learning (MLlib)

* MLlib is a machine learning library build on top of Spark which allows users to create and train machine learning models.

The installation process of Apache Spark requires some components to be installed properly to run. The following steps were followed in order to install Apache Spark and run pyspark on Windows 10.

1. Install Python 3.11.
2. Install Java 8 from Oracle Website.
   1. Set the Environment Variables and Path.
      * 1. JAVA\_HOME = C:\Program Files\Java\jdk1.8
        2. PATH as C:\Program Files\Java\jdk1.8.0\_201\bin

(Use paths as per your installation directory)

1. Install Apache Spark. Download from the link,

(<https://spark.apache.org/downloads.html>)

* 1. Choose the right version (I chose Spark release: 3.2.4 (Apr 13 2023) & package type: Pre-built for Apache Hadoop 3.2 and later)
  2. Extract and copy the downloaded tar file to the directory of your choice.
  3. Set the Environment variable and Path carefully. (In my case)
     1. SPARK\_HOME = C:\apps\spark-3.0.0-bin-hadoop2.7
     2. HADOOP\_HOME = C:\apps\spark-3.0.0-bin-hadoop2.7
     3. Add PATH as C:\apps\spark-3.0.0-bin-hadoop2.7\bin

1. Download ‘winutils.exe’ file from [https://github.com/cdarlint/winutils](https://github.com/cdarlint/winutils%20) and choose the right version. Paste the winutils.exe file inside the directory in which the tar file was pasted. (Example: Paste it in the directory “C:\apps\spark-3.0.0-bin-hadoop2.7\bin”)
2. If installed successfully, when you type ‘pyspark’ on command, something like this will be shown:

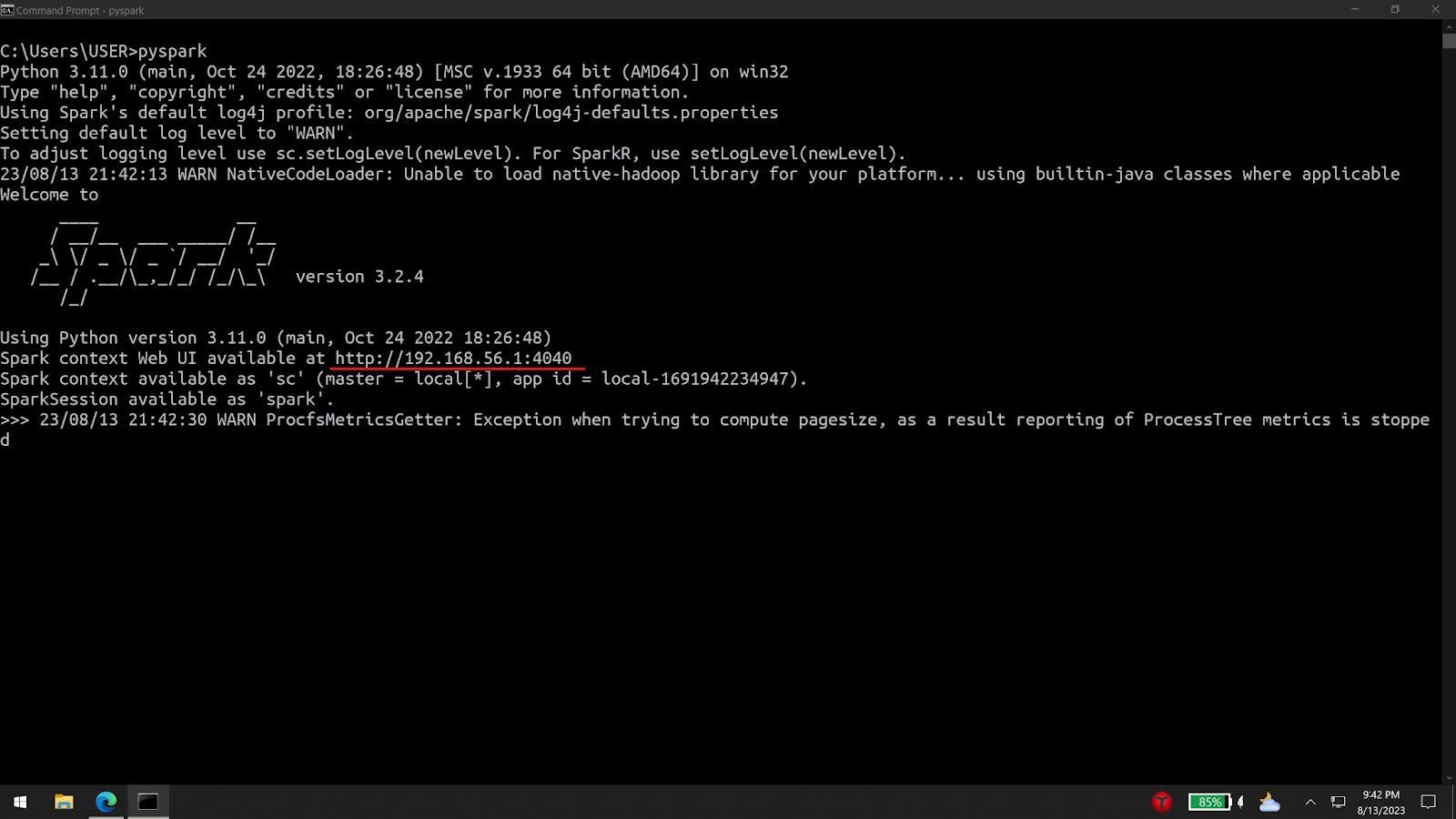


Figure 1 Checking if Spark installed Succesfully

For the coding task, Jupyter Notebook was used. To use pyspark in jupyter notebook, pip (a package installed for python) was used to install pyspark. To make sure no issue arises, the Apache Spark installation version and pyspark installation version must be the same. (3.2.4 in this case). When using pip to install pyspark, the specific version can be specified and the pyspark version can be checked as shown in the picture.



Figure 2 Installing and Checking Version of PySpark

## Dataset Description and Analysis Goal

The dataset was download from Kaggle. It can be found in Kaggle’s website with name “Customer Personality Analysis” or can be downloaded using the link <https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis>. The dataset is a collection of various attributes of different customers of a company. Initially it contains 2240 rows and 29 columns or features.

The table below provides detail about the various features with their data types.

|  |  |  |
| --- | --- | --- |
| Attribute | Description | Data Type |
| ID | Customer's unique identifier | Integer |
| Year\_Birth | Customer's birth year | Integer |
| Education | Customer's education level | String |
| Marital\_Status | Customer's marital status | Integer |
| Income | Customer's yearly household income | Double |
| Kidhome | Number of children in customer's household | Integer |
| Teenhome | Number of teenagers in customer's household | Integer |
| Dt\_Customer | Date of customer's enrollment with the company | String |
| Recency | Number of days since customer's last purchase | Integer |
| Complain | 1 if the customer complained in the last 2 years 0 otherwise | Integer |
| MntWines | Amount spent on wine in the last 2 years | Integer |
| MntFruits | Amount spent on fruits in the last 2 years | Integer |
| MntMeatProducts | Amount spent on meat in the last 2 years | Integer |
| MntFishProducts | Amount spent on fish in the last 2 years | Integer |
| MntSweetProducts | Amount spent on sweets in the last 2 years | Integer |
| MntGoldProds | Amount spent on gold in the last 2 years | Integer |
| NumDealsPurchases | Number of purchases made with a discount | Integer |
| AcceptedCmp1 | 1 if customer accepted the offer in the 1st campaign 0 otherwise | Integer |
| AcceptedCmp2 | 1 if customer accepted the offer in the 2nd campaign 0 otherwise | Integer |
| AcceptedCmp3 | 1 if customer accepted the offer in the 3rd campaign 0 otherwise | Integer |
| AcceptedCmp4 | 1 if customer accepted the offer in the 4th campaign 0 otherwise | Integer |
| AcceptedCmp5 | 1 if customer accepted the offer in the 5th campaign 0 otherwise | Integer |
| Response | 1 if customer accepted the offer in the last campaign 0 otherwise | Integer |
| NumWebPurchases | Number of purchases made through the company’s website | Integer |
| NumCatalogPurchases | Number of purchases made using a catalogue | Integer |
| NumStorePurchases | Number of purchases made directly in stores | Integer |
| NumWebVisitsMonth | Number of visits to company’s website in the last month | Integer |

The dataset consists of integers, doubles and strings. Some features have been assigned data types different from what it really is. For example, the ‘Dt\_Customer’ which should originally be a Date datatype has been assigned as string. Also, the ‘Year\_Birth’ feature has been assigned a string datatype. These kinds of incosistencies will be addressed when preprocessing the data which will be discussed and explained in the latter sections.

The analysis of this dataset can provide many valuable insights on the customers as well as spread light on the parts which the company needs to focus on. The features consist of both personal descriptions of the customers as well as their behaviours regarding how and when they buy goods. Using these features, this project aims on performing a cluster analysis of the customers which might help unfold different clusters of customers based on different attributes which would help the company in mnay ways. Also, another goal of this analysis is to create a predictive model which would help in predicting if a customer would accept the promotional or other campaigns led by the company. Following sections will provide the findings of exploratory analysis of the dataset.

## Importing Necessary Libraries and Initializing Spark Context

This project uses PySpark and its functions solely for all data preprocessing, analysis and model creating tasks. The following image provides a code snippet showing the libraries that needs to be imported.

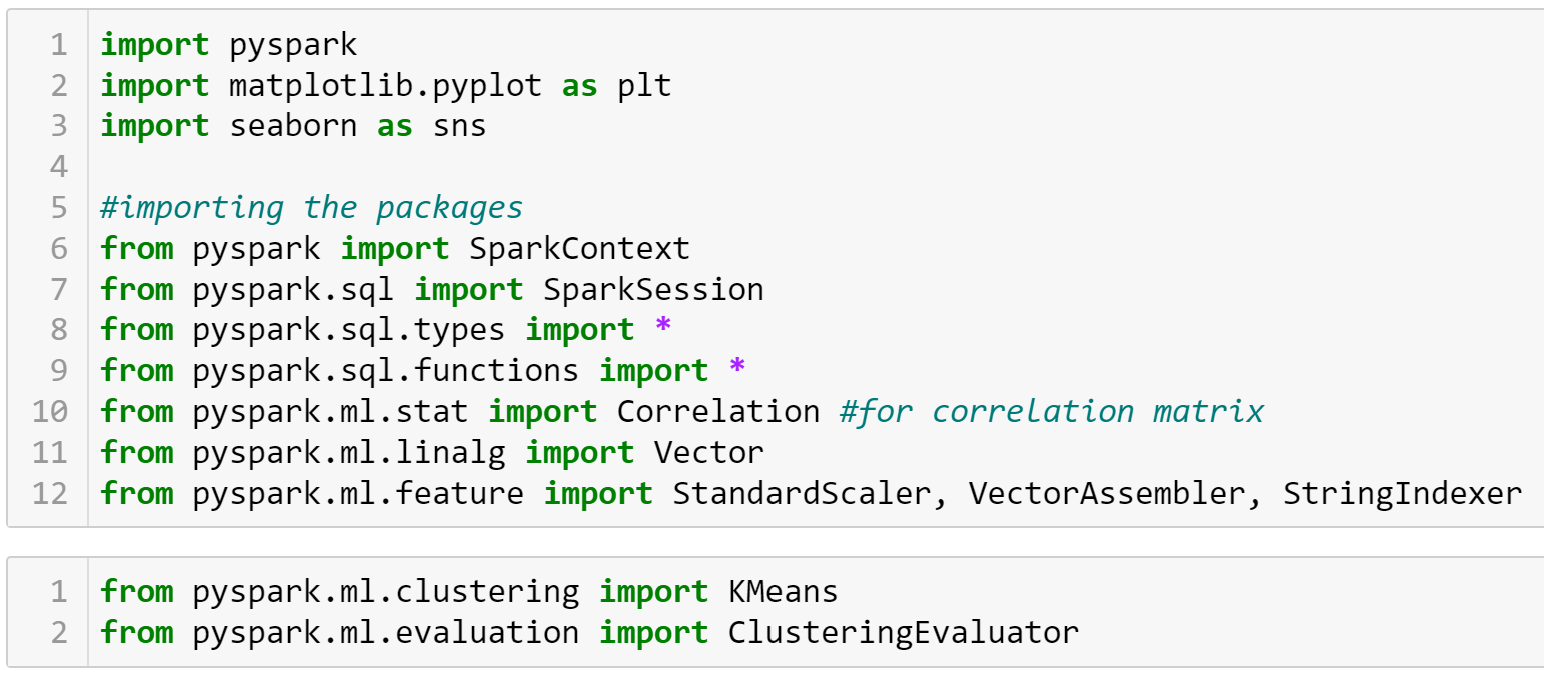


Figure 3 Importing required libraries and functions

To interact with Apache Spark, either a Spark Context or a Spark Session can be created as it is the entry point to connect with Spark. Here, the spark session has been used to interact with Apache Spark. ((NNK), 2023)

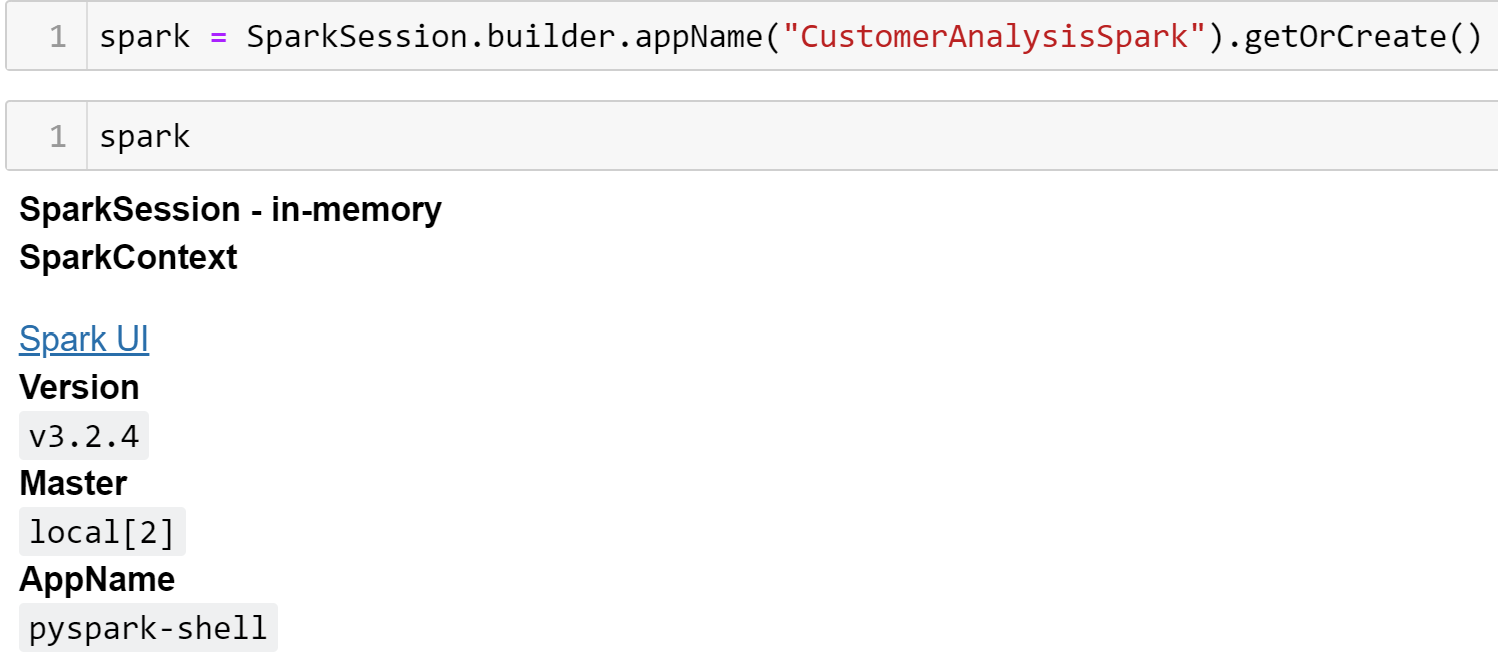


Figure 4 Spark Session Creation

After creating the spark session, the UI of the spark session can be accessed using the ‘Spark UI’ link. The created session is then used to load the dataset as follows:



Figure 5 Loading the Dataset using spark

## Data Preprocessing

For any data related task, data preprocessing is the most crucial step. Data Preprocessing is essential before using it. The data is preprocessed in order to check for any missing values, noisy data, and other inconsistencies before executing it to any kind of algorithms. (Pushpa Singh, 2021) Computational intelligence needs clean, numeric, homogeneous, well organized, and normalized data. Data quality is a major concern in big data processing and knowledge management systems. (Khalid K. Al-jabery, 2020)

Upon looking at the dataset, it is seen that it has 30 columns, where one column is an additional index column added by the dataframe. It is dropped for the sake of simplicity. The number of rows in the dataset is 2240. The following image shows the results of printing the schema of the DataFrame.

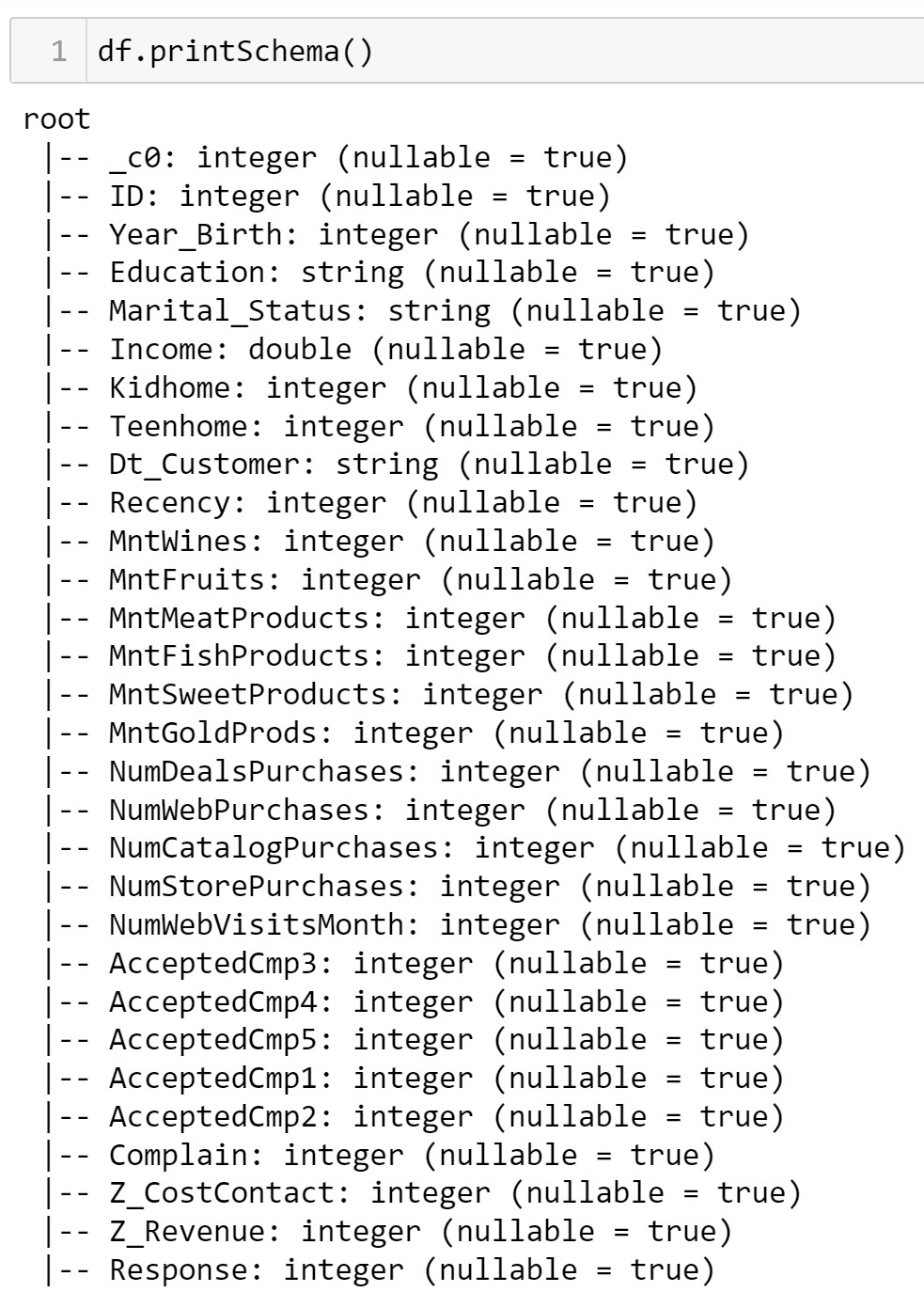


Figure 6 Schema of the DataFrame

As discussed earlier, there are many inconsistencies in the data types of the dataset which needs to be addressed. And also there are columns which needs to changed from one type to another which would be helpful for later part to create the machine learning models.

The ‘Dt\_Customer’ column has the dates when the customer got engaged with the company. As it is a date column but has been assigned as a string type, it needs to be changed to date type. It can be done by using the “df.withColumn()” method.

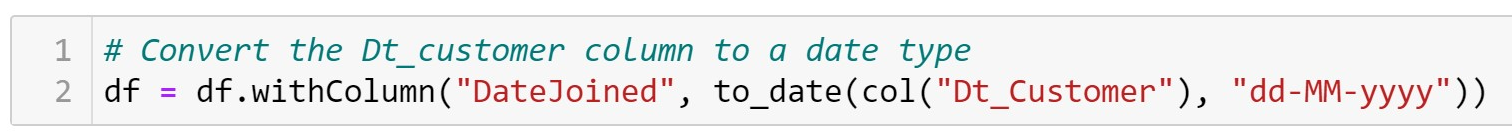


Figure 7 Changing Column Dt\_Customer to Date type and changing name to DateJoined

Along with changing the date type, the name of the column has also been changed to ‘DateJoined’. Furthermore, another column ‘YearJoined’ and ‘DaysSinceJoined’ has also been derived from the feature which would help in visualisation and model building aspects.

One main point to be noted with this dataset is that the data was collected back in 2015 and many columns have values that would only make sense if the reference date for present is taken as 2015. A feature which would benefit from this assumption is the ‘Year\_Birth’ feature which is the year of birth of the customers. The ages of the customers have been calcuated for the year 2015 as shown in the image below.

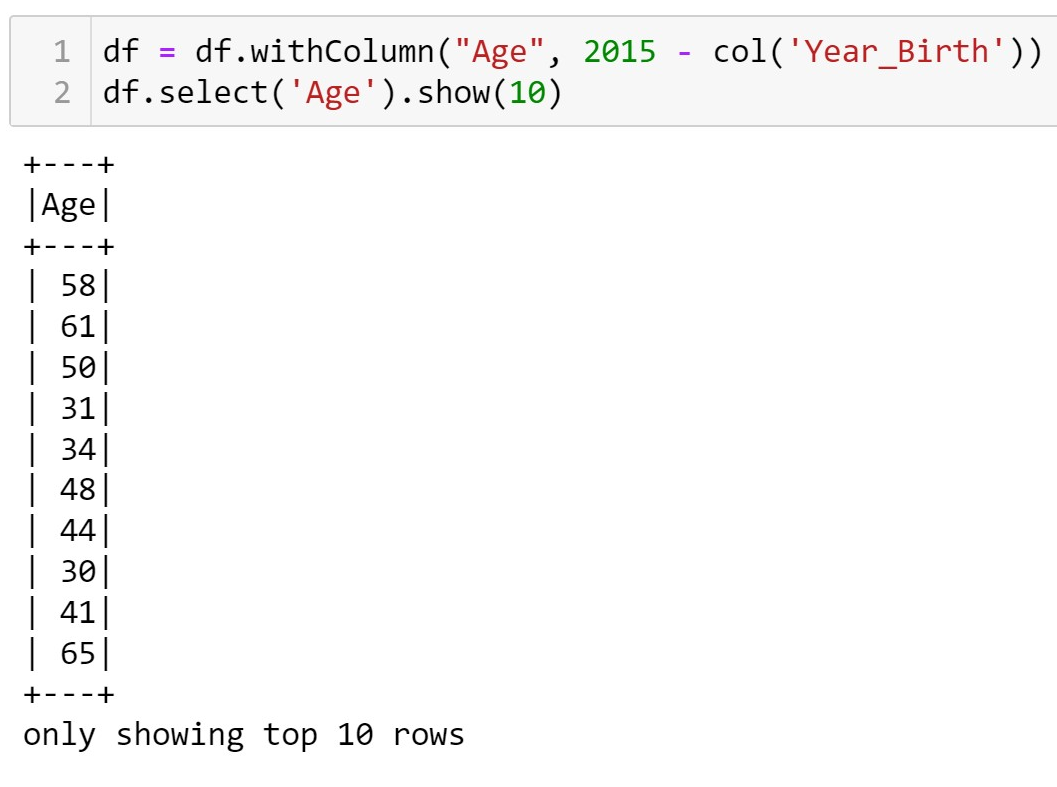


Figure 8 Creating Age attribute form Year\_Birth

Features like ‘ID’ which would not provide much value to the dataset was also dropped from the dataset. To deal with the repetitive columns that were created earlier and for simplicity of model building the created columns ‘Year\_Birth’, ‘YearJoined’, and ‘DateJoined’ were also removed. The resulting dataframe then had 28 columns with 2 non numerical columns and 26 numerical columns.

When checked for null values, the only column with null values was ‘Income’ with 24 null values. They were dropped using the “df.na.drop()” function which brough the number of rows down to 2216.

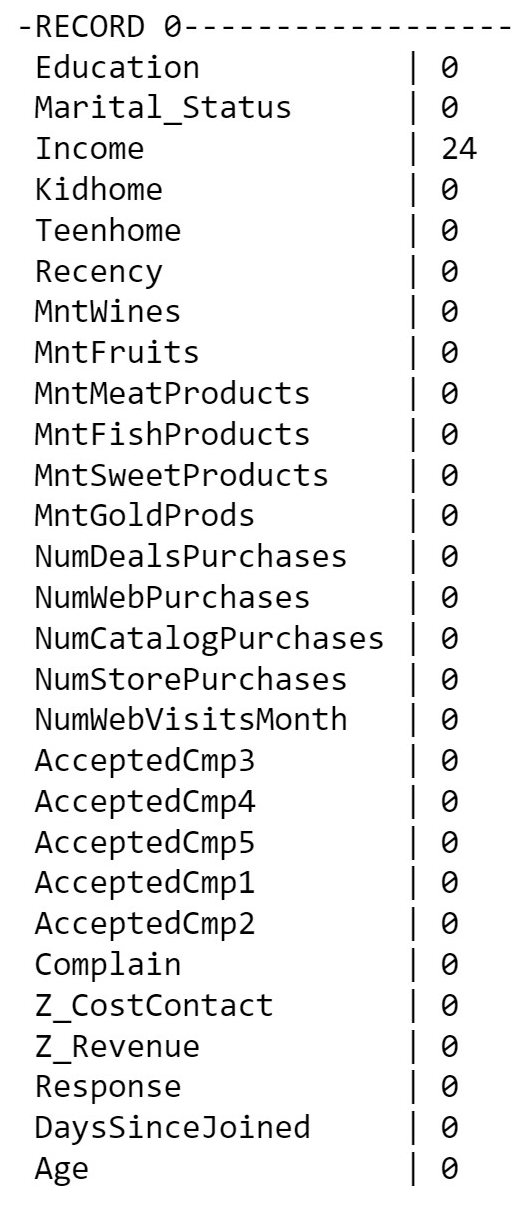
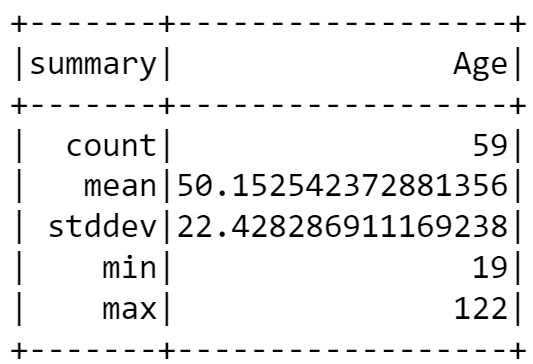


Figure 9 Null values in Income

The ‘Age’ feature of the dataset provides the age of the customers. Upon looking at the counts of unique ages of the customers it was found that there are customers of age more than 100. These values seemed unreal so they were also dropped.



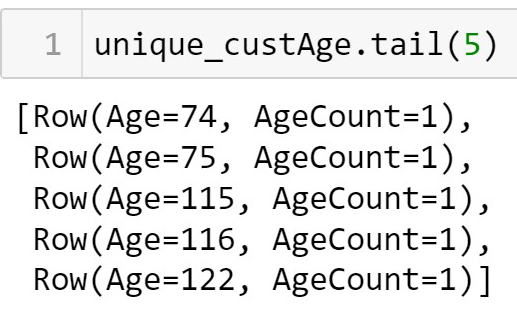


Figure 10 Ages more than 100 Figure 11 Summary of Age

# Discussion of Findings

# Conclusion

# References

# Appendix