Analysis on Sales Data using PySpark

ECE 552 DL1 - Big Data Technologies| Prof. Erton Boci, Ph.D.

George Mason University Fall 2021

TEAM 11

Rishi Varma Dendukuri, Jyoti Mote, Sai Praneeth Ratakonda

**1. Abstract**

The process of integrating, evaluating, and comprehending diverse data connected to sales operations, such as sales, customers, and transaction data, is known as sales analysis. Analyzing sales data is a crucial component as it gives us insights of products which are giving highest/ lowest profits, sales forecasting, revenues of a particular region etc. Analyzing this type of data helps you evaluate the performance of your company against its sales goals. This also helps us to find patterns in the data which enables us to optimize our sales process. This research helps us to make data driven decisions instead of following our gut, we can be aware of the trends in the current market. We can also make the customer service experience better and expand the company’s reach with the help of this analysis. Regular sales analysis establishes responsibility, discloses information about your customers, identifies the characteristics of top-performing sales agents, and a variety of other factors that will boost your bottom line. I've demonstrated the many sorts of sales analysis methodologies and provided you with a step-by-step plan for doing your first sales analysis. (Efti, 2020) (finereport, 2020) The data we selected has 1.5 million records with 13 columns. This data consists of spatial data such as region and country, the date on which order is created and received, the revenue, cost price, selling price and profit of each item, item type and the type of order.

**2. Introduction**

Apache Spark is a large-scale data processing open-source unified analytics engine. Apache Spark is a more contemporary framework that combines a methodology for creating programs on top of an engine for distributing programs over clusters of computers. It is intended to meet the demands of the data scientist community, particularly in support of the Read-Evaluate-Print Loop (REPL) technique to interactively engaging with data.

(Penchikala, 2015)

MapReduce's linear scalability and fault tolerance are preserved in Spark, but it is extended in three keyways. First, instead of using a strict map-then-reduce format, its engine may use a more generic directed acyclic graph (DAG) of operators. This means that when MapReduce must write intermediate results to a distributed file system, Spark may transmit them immediately to the next stage in the pipeline. Second, it adds a rich range of transformations to this capability, allowing users to describe computation in a more natural way. Third, unlike MapReduce, Spark offers in-memory processing across a cluster of servers, eliminating the need for storage to store intermediate data. (Z.Milosevic, 2016)

**3. Objectives**

1. Top 10 countries having the highest revenue in Beverages?
2. Total profit for different sales channel.
3. Total profit based on Order priority for North America region.
4. Total profit per order priority per region.
5. Total\_Profit Month wise for North America.

**4. Literature review**

The following are some of the top articles related to this project.

**1. Walmart’s Sales Data Analysis- A Big Data Analytics Perspective:**

The main purpose is to examine and analyze Walmart's publicly available statistics to gain insights and a broad picture of the company. Retail businesses generate money by selling goods. The stores network is made up of various subsidiaries that are in different parts of the world. Because the company's retail network is so broad and dispersed across the country, it would be impossible for it to fully appreciate customer needs and market potential in each location. They used Walmart's obtained store sales information to identify the factors impacting sales in different stores situated in different geographical regions, such as the unemployment rate, gasoline costs, temperature, and holidays, so that resources may be handled intelligently to optimize returns. These insights, for example, can help retailers better understand market conditions and the various factors that influence sales.

(IMMERMAN, 2017) The Easter holiday would boost income, allowing businesses to better manage their resources (supply of goods and human resources). They combined Apache Spark with a Hadoop build that used HDFS as a data storage option. As development tools, they utilized InteliJ Idea Community Edition as a substitute for spark-shell and iPython Notebook. With Apache Spark, all the fundamentals are combined in a single system across many libraries, requiring you to simply call the libraries you require. The dataset was mapped and reduced to a key-value format for comparison using Spark SQL.

(Manpreet Singh)

**2. Data Analysis and Price Prediction of Black Friday Sales using Machine Learning Techniques:**

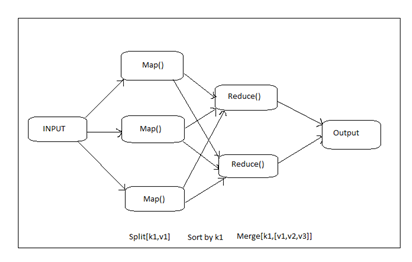
The prediction model developed in this study will aid in the analysis of the relationships between various variables. For training and prediction, the Black Friday Sale Dataset is used. The Black Friday Sales Dataset is the largest online dataset, and it is accepted by a variety of e-commerce companies. Retailers would be able to study and customize offers for more customers' favorite items based on the purchase predictions produced. (Serhat Ata, 2021)

The predictor variable will be the Purchase Variable. The Purchase Variable will forecast how much money a consumer will spend during Black Friday discounts. Linear regression, Ridge Regression [19], Lasso Regression [19], Decision Tree Regressor, and RandomForest Regressor are the methods utilized to create the system. 5 fold cross-validation is used to train the models [4][12]. Mean Squared Error is the performance evaluation metric employed (MSE). With an MSE score of 3062.719, the Random Forest Regressor outperforms the other algorithms.

(Amruta Aher, 2021)

**3. Youtube Data Analysis Using Hadoop Framework:**

The main purpose of this project is to assess real-time and informed judgments utilizing YouTube data by demonstrating Apache Hadoop framework features. Hadoop is a framework for storing and managing enormous volumes of data. During the writing of this paper, they gathered YouTube data from a variety of channels. The document output from the console application is then fed into Mapper from an HDFS file. HDFS is Hadoop's main program, and it may be accessed directly via Hadoop's shell-like commands. After that, we'll use a mapper to shuffle the data and a reducer to aggregate the meaningful results, which may be finished as a reducer. (Pathak, 2020)

 **Fig 1: Analysis of YouTube data using Hadoop MapReduce framework**

Based on the findings, it was determined that Apache Hive may be used to easily retrieve large amounts of data. Hive pulls YouTube data by modeling an API for channel users. MapReduce allows you to examine large amounts of data in several phases. Due to the large space consumption of each task, implementing iterative map reduce jobs is costly. Hive, which produces efficient results, overcomes the drawback of map reduction. (Ashwini T, 2021)

**5. System Architecture**

The sales data, which is in csv format, is first imported onto an Azure/ AWS virtual machine. To increase query performance, we imported the csv into a spark data frame and converted it to a paraquet file before working on the data. The data is then saved in Apache Parquet, an open-source column-based data storage format that is part of the Apache Hadoop ecosystem. In the third phase, Apache Spark does data cleaning and pre-processing, which includes eliminating any null values, spaces, and special unrequired characters. Using Jupyter and zeppelin, we construct data visualization and code in the following stage.

Logo, company name

Description automatically generated

**Fig 2: System Architecture**

The above figure displays the architectural design of the system for the analysis on sales using Apache Spark.

**Hardware specifications:**

* Operating system: Windows 10, 64 bit
* CPU: 4 core Intel64 Family 6 Model 85
* Memory: 16 GB
* Hard Disk:128GB

**Software’s used:**

* Apache Spark: version 2.4.7
* Apache Parquet is used for data
* transformation.
* Jupyter Notebook for operating queries and performing visualizations.

**6. About Dataset:**

The dataset we have taken to perform this analysis has 1.5million records with 13 features. It consists of geographical data which is depicted by the attribute’s country and region, Date at which order is created and shipping date. It also consists of a primary key which is Order ID. All the records are unique in this data. “Item Type”, “Sales Channel”, “Order Priority”, “Units sold”,

“Unit cost”, “Total Revenue”, “Total cost”, “Total profit” are the other attributes in this dataset. (Excel, 2017)

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Attribute** | **Data Type** |
| ID | Varchar | Country | Varchar |
| Region | Varchar | Item type | Varchar |
| Sales Channel | Varchar | Order  Priority | Varchar |
| Date | Date | Shipping Date | Date |
| Unit Sold | Integer | Unit Cost | Float |
| Total Revenue | Float | Total Cost | Float |
| Total Profit | Float |  |  |

**Table 1: Attributes and data types**

**7. Higher Level Framework**

The below designed framework gives us a clear over-view of the process in a step-by-step manner throughout this research. To clean, pre-process, wrangle, explore, analyze and visualize the data tools like pyspark API and SparkSQL is used. The primary tool used to clean and wrangle the data is pySpark API. After cleaning and wrangling sql and jupyter are used for exploratory analysis and visualizations.

Exploratory Analysis

Importing Data from the source

Visualizations

Pre-processing the Data

**Fig 3: High-Level Architecture**

**8. Data Cleaning**

We use PySpark for cleaning, wrangling and transforming the data. It is an application programming interface which helps in using python combined with the spark framework. Some of the columns has no value in this dataset other than representing each record. One of those attributes is “ID”. Our dataset consists of very few missing values and inconsistent data. Some of the column names have gaps. Those space have been replaced with underscores to improve consistency.

The special characters have been removed to smoothly store it in parquet file format. Many of the attributes in the dataset have long variability. Many of the attributes consists of numerical values and the rest are VarChar. The following commands are used to clean the data.

*df.select([count(when(isnan(c) | col(c).isNull(), c)).alias(c) for c in df.columns]).show()*

*df = df.select([col(c).alias(re.sub("[^0-9a-zA-Z$]+","\_",c)) for c in df.columns])*

**9. Importing the Libraries**

We have imported PySpark library. PySpark is a Python interface to Apache Spark. It includes the PySpark shell for interactively examining data in a distributed environment, as well as the ability to develop Spark applications using Python APIs. Most Spark technologies, such as Spark SQL, DataFrame, Streaming, MLlib (Machine Learning), and Spark Core, are supported by PySpark. The following commands are used to import this library. (apache.org, n.d.)

*import findspark*

*findspark.init()*

*import pyspark*

From Pyspark we have imported sparkcontext, SQLContext and sparksession.

SparkContext is the basic foundation for all spark functionalities. When you run a Spark application, a driver application with the main function runs, and your SparkContext is established here. The operations are then carried out by the driver programme inside the executors on worker nodes. (tutorialspoint.com, n.d.). SQL context where you'll find all of the Spark SQL features. A SQLContext can be used to build data Frames, register Data Frames as tables, execute SQL over tables, cache tables, and read parquet files, among other things. . Sparksession creates data Frames, register data Frames as tables, execute SQL over tables, cache tables, and read parquet files using a SparkSession. (apache.org, n.d.) . The following commands are use to import the above mentioned libraries.

*from pyspark import SparkContext*

*from pyspark import SQLContext*

*from pyspark.sql import SparkSession*

*sc = SparkContext.getOrCreate()*

*spark = SparkSession.builder.getOrCreate()*

Some SQl functions have also been imported to clean the data.

**10. Importing the Data**

The original dataset we have extracted is in excel format. We have read that dataset file into a new data frame using the following commands.

*df = spark.read.options(header='True', inferSchema='True') \.csv("O:/MS/ECE552/Final\_Project/1500000 Sales Records/1500000 Sales Records.csv")*

**11. Data Transformation**

The data is initially in Comma Separated values format. But the CSV format will be effective when we deal with data sets of small size. When we are dealing with large datasets, csv is not the ideal format. In this case, we use paraquet or Avro. These two can deal with datasets containing millions of records. This is the reason why we have converted our dataset into paraquet format. This enables optimized columnar storage. The following commands are used to transform the data into the paraquet format.

*df.write.parquet("O:/MS/ECE552/Final\_Project/1500000 Sales Records/1500000 Sales Records.parquet")*

*par\_df= spark.read.parquet("O:/MS/ECE552/Final\_Project/1500000 Sales Records/1500000 Sales Records.parquet")*

**12. Data Querying**

We have used SparkSQL for querying. Spark SQL is a new Apache Spark module that combines relational processing with the functional programming API of Spark.

Spark SQL allows Spark programmers to take advantage of the advantages of relational processing (for example, declarative queries and optimal storage) while also allowing SQL users to utilize advanced analytics libraries in Spark (e.g., machine learning).

**13. Analysis and Visualizations**

**Top 10 countries having the highest revenue in Beverages**

Table

Description automatically generated

Fig (4.1)

Chart, bar chart

Description automatically generated

Fig (4.2)

The above fig (4.2) represents a bar plot, it shows the trends of top 10 countries having the highest revenue in Beverages. We can observe that United Kingdom records the highest revenue, there is not much difference in the revenue between Serbia and Namibia and Ethiopia records the least.

In order to visualize this chart, we have created a data frame with columns Country and Total\_Revenue. The column Total\_Revenue is derived from aggregating all the revenues and grouped by country, as we are calculating the revenue for beverages, a filter has been added where item\_type would only be beverages.

The Revenue recorded is in the units of USD. The highest revenue is over 175 million dollars, and the least revenue is over 170 million dollars.

**Total profit for different sales channel**

Table

Description automatically generated

Fig (5.1)

Chart, pie chart

Description automatically generated

Fig (5.2)

Chart, scatter chart

Description automatically generated

Fig (5.3)

The above fig (5.2) and fig (5.3) represents a pie chart and a dot chart, it shows the trends of Profit made by the mode of Sales Channel, pie chart represents the online sales and dot chart represents the offline sales.

We can observe that Cosmetics records the highest profit, followed by household items for both online sales and offline sales and fruits records the least profit for both online and offline mode.

In order to visualize these charts, we have created two data frames for online and offline with columns Sales\_Channel, Item\_Type and Total\_Profit. The column Profit is derived from aggregating all the profits grouped by Sales\_Channel and Item\_Type.

The Profit recorded is in the units of USD. The highest profit is over 1.73 million dollars and the least profit is over 2.4 thousand of dollars.

**Total profit based on Order priority for North America region**

Table

Description automatically generated

Fig (6.1)

Chart, bar chart

Description automatically generated

Fig (6.2)

The above fig (6.2) represents a bar graph, it shows the trends of Total profit based on Order priority for North America region. We can observe that High priority orders recorded the maximum profit followed by critical orders and the least profit is recorded by the medium priority orders.

In order to visualize this chart, we have created a data frame with columns Order\_Priority and Total\_Profit. The column Total\_Profit is derived from aggregating all the profits grouped by Order\_Priority.

The Profit recorded is in the units of USD. The highest profit is over 3266 million dollars, and the least profit is over 3104 million dollars.

**Total profit per order priority per region**

Table

Description automatically generated

Fig (7.1)

Chart, bar chart, box and whisker chart

Description automatically generated

Fig (7.2)

The stacked bar plot from the above figure represents the profit recorded per each region and the profit for each order priority grouped by that region. We can observe that Europe and Sub-Saharan Africa records highest profits among all other regions. We can also observe that each order priority has contributed equally to the total profit. The region of North America records the least profit when compare with all the other regions.

In order to visualize this chart, we have created a data frame with columns Region, Order\_Priority and total\_profit. The column Total\_Profit is derived from aggregating all the profits grouped by that region.

The Profit made by Europe and Sub-Saharan Africa is ranging in between 140000 million dollars to 160000 million dollars. The value of the profit is in USD. There is a minute difference in profits for each order priority.

**Total\_Profit Month wise for North America**

Table

Description automatically generated

Fig (8.1)

Chart, line chart

Description automatically generated

Fig (8.2)

The above plot is a line plot indicating gross profit recorded in each month in ascending order. From the visualization we can observe that the gross profit is least in the month of February and highest in the month of May. There is a major drop of gross profits from January to February. The profit again will rapidly increase from February to March and slightly decrease from march to April and increase from April to May and starts to decline from May to July.

The profits recorded is in the units thousands of USD. The highest profit recorded is over 6601 million dollars and the least profit recorded is just over 5916 million dollars.

This plot is visualized by creating a data frame which has two columns named month and Total\_Profit. The total\_Profit is calculated by aggregating the profit of each record grouped by the month which is extracted from the date of order.

**14. Challenges**

* It is always difficult to deal with large datasets. Especially any dataset with more than 1 million records.
* When importing the data, we have faced some issues with the process as there are some issues loading that volume of data.
* While cleaning the data, there are some uncertainties like removing special characters for instance, we have to look at the dataset are there any other additional characters than what we have specified.
* The commands took more time until we have converted it into paraquet file.
* Some of the data frames we have created requires highly complex querying.

**15. Conclusion:**

Based on the sales statistics, we may deduce that high priority orders in North America make the most money, whereas medium priority orders make the least. The United Kingdom has the greatest revenue in the beverage industry. The profit margins between online and offline sales are statistically identical. When compared to all other regions, Europe and Sub-Saharan Africa make the most money, while North America makes the least. The lease profits are recorded in February and the highest profits are recorded in the month of May. The highest increase in profits is recorded in between the months of February and March.

**16. Future Scope**

We can use predictive modelling techniques to predict the following:

* Number of items required based on the month, so the inventory is not out of stock.
* Forecasting the profit for upcoming years on different demographic regions.
* We can research more the items sold region wise to yield maximum profits using deeper exploratory analysis.

# Works Cited

1. Amruta Aher, D. K. (2021, July). Data Analysis and Price Prediction of Black Friday Sales using Machine Learning Techniques. *International Journal of Engineering Research & Technology (IJERT), 10*(7), 621-627.
2. apache.org. (n.d.). *PySpark Documentation*. Retrieved from apache.org: <https://spark.apache.org/docs/latest/api/python/>
3. apache.org. (n.d.). *pyspark.sql module* . Retrieved from apache.org: <https://spark.apache.org/docs/1.6.2/api/python/pyspark.sql.html>
4. Ashwini T, S. L. (2021). YOUTUBE DATA ANALYSIS USING HADOOP FRAMEWORK. *International Journal of Engineering Applied Sciences and Technology, 5*(11), 305-313.
5. Efti, S. (2020). *How to perform a sales analysis (step-by-step with methods & metrics)*. Retrieved Dec 2021, from blog.close.com: <https://blog.close.com/sales-analysis/>
6. Excel. (2017, August 26). *E for Excel* . Retrieved from eforexcel: <https://eforexcel.com/wp/downloads-18-sample-csv-files-data-sets-for-testing-sales/>
7. finereport. (2020, June 24). *A Beginner’s Guide to Sales Analysis*. Retrieved dec 2021, from finereport.com: <https://www.finereport.com/en/data-analysis/sales-analysis.html>
8. IMMERMAN, G. (2017, Sept 06). *WALMART BIG DATA CASE STUDY*. Retrieved from Machine Metrics: <https://www.machinemetrics.com/blog/walmart-big-data-case-study>
9. Manpreet Singh, B. G. (n.d.). Walmart's Sales Data Analysis - A Big Data Analytics Perspective. *2017 4th Asia-Pacific World Congress on Computer Science and Engineering.* APWC on CSE.
10. Pathak, R. (2020, Aug 02). *YouTube Data Analysis Using Hadoop Tools*. Retrieved from Start it up: <https://medium.com/swlh/youtube-data-analysis-using-hadoop-tools-ff47601d6b12>
11. Penchikala, S. (2015, Jan 30). *Big Data Processing with Apache Spark – Part 1: Introduction*. Retrieved from InfoQ: <https://www.infoq.com/articles/apache-spark-introduction/>
12. Serhat Ata, A. B. (2021, March). *An Empirical Study to Determine the Impact of Black Friday Days on Consumer Purchasing Behavior*. Retrieved from Research Gate: <https://www.researchgate.net/publication/349733441_An_Empirical_Study_to_Determine_the_Impact_of_Black_Friday_Days_on_Consumer_Purchasing_Behavior>
13. tutorialspoint.com. (n.d.). *PySpark - SparkContext*. Retrieved from tutorialspoint.com: <https://www.tutorialspoint.com/pyspark/pyspark_sparkcontext.html>
14. Z.Milosevic, W. (2016). Real-Time Analytics. In *Big Data* (pp. 39-61). Elsevier Inc. Retrieved from <https://www.sciencedirect.com/topics/computer-science/apache-spark>