EFFECT OF COVID ON SALES AND E-COMMERCE

PROJECT REPORT (GROUP 4) – BUAN 6346.001

PROF. Antonio Paes

TEAM MEMBERS

- Ram Smaran Pandiri (rxp210024)
- Shubham (sxx210007)
- Keerthana Aravindhan (kxa210051)
- Manas Chourey (mxc200000)
- Abhiram Lankipalli (arl170230)

EFFECT OF COVID ON SALES AND E-COMMERCE

1. INTRODUCTION

In 2019, one of the biggest events of the twenty-first century made an appearance unexpectedly: COVID-19. As early as late 2019, incidental cases of COVID continued to increase exponentially. As the severity increased, more facets of society continued to shut down, and in March, the United States officially went into lockdown. One of the most affected constituents were businesses across the globe. As governmental policies came into play, people stopped shopping for non-essentials, and tried to over-stock on necessities (the well-known toilet paper shortage). This led to shortages in the supply-chain, which ultimately affected the sales of many products.

While everyone is aware that the months following lockdown were difficult for the U.S. economy to sustain, not as much research has been done into the couple months that led up to the lockdown. An analysis of this data would provide better insight into why the supply and demand issues arose the way they did. The data could also provide a better pattern recognition sequence so that experts could identify when a similar trend starts developing and control the problem before it becomes too big (prevention). In addition, the analysis of this data could better inform business owners how to become more cost-effective during exceptional circumstances.

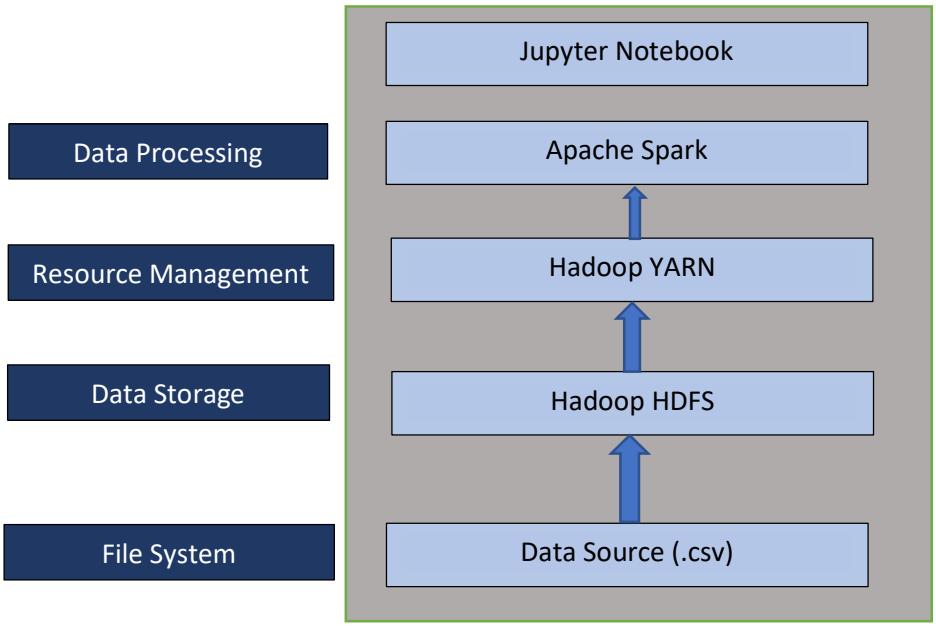
Our group hypothesized that an increase in the number of COVID-19 cases would increase sales. While most businesses would have suffering sales, our group would analyze an e-commerce business structure. Since most people would hope to avoid exposure to COVID-19, we would likely see a surge in the number of online customers. Therefore, as more and more people became sick from October 2019-February 2020, real-world business sales would drop, and as a result there would be an increased shift into online sales.

BUSINESS QUESTIONS

1. Does increasing covid cases affect cosmetic sales?
2. What are the most affected brands?
3. Does increasing covid cases affect jewelry sales?
4. Does increasing covid cases affect jewelry price?

2. FRAMEWORK USED

We used Hadoop + Spark for our project.



Why we chose this framework?

Jupyter notebook on Spark is a very nice combination for big data processes and data analytics. It provides a good user interface and combines multiple works into a single file. We could do everything from data processing and analysis to data visualization.

3. DATASET

We selected our dataset from Kaggle,

1. **eCommerce in Cosmetic Shop:** https://www.kaggle.com/datasets/mkechinov/ecommerce-events-history-in-cosmetics-shop
2. **eCommerce in Jewelry Shop:** https://www.kaggle.com/datasets/mkechinov/ecommerce-purchase-history-from-jewelry-store
3. **Covid Dataset:** https://www.kaggle.com/datasets/gauravduttakiit/covid-19

eCommerce (Cosmetic) dataset consists of data from Oct 2019 – Feb 2020. eCommerce (Jewelry) dataset consists of data from Dec 2018 – Dec 2021. The covid dataset consists of data from Jan 2020 – Apr 2022. Total size of the datasets is around 3 GB.

4. DATA STRUCTURE

E-commerce (Cosmetic):

PROPERTY	DESCRIPTION
event_time	Time when event happened at (in UTC).
event_type	Only one kind of event: purchase.
product_id	ID of a product
category_id	Product's category ID
category_code	Product's category taxonomy (code name) if it was possible to make it.
Brand	Downcased string of brand name
Price	Float price of a product.
user_id	Permanent user ID.
user_session	Temporary user's session ID. Same for each user's session.

E-commerce (Jewelry):

PROPERTY	DESCRIPTION
OrderDate	The date when the order is placed
Order_ID	ID of the order
Product_ID	ID of a product
Quantity	The number of items purchased
Category_ID	ID of a category
Category_Alias	Name of the category
Brand_ID	ID of brand
Price	Price of the product

User_ID	ID of user
Gender	Gender of the person
Color	Color of the product
Metal	Type of metal in the product
Gem	Type of gem in the product

Covid:

PROPERTY	DESCRIPTION
Date	Date when the case was reported
Country	Country from where the case was reported
Confirmed	Total number of cases reported
Recovered	Total number patients recovered
Deaths	Total number of deaths reported

5. LOADING DATA INTO HADOOP

- Starting Hadoop Daemon

```
shubham@shubham-VirtualBox:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as shubham in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [shubham-VirtualBox]
Starting resourcemanager
Starting nodemanagers
```

- Making a directory for the files

```
shubham@shubham-VirtualBox:~$ hadoop fs -mkdir project
```

- Checking the directory is created

```
shubham@shubham-VirtualBox:~$ hadoop fs -ls
Found 2 items
drwxr-xr-x   - shubham supergroup          0 2022-10-20 18:38 .sparkStaging
drwxr-xr-x   - shubham supergroup          0 2022-11-30 23:10 project
```

- Adding all ".csv" files to HDFS

```
shubham@shubham-VirtualBox:~/Downloads/Datasets$ hadoop fs -put /home/shubham/Downloads/Datasets/*.csv project
shubham@shubham-VirtualBox:~/Downloads/Datasets$ hadoop fs -ls
```

- Checking the files are added

```
shubham@shubham-VirtualBox:~/Downloads/Datasets$ hadoop fs -ls project
Found 7 items
-rw-r--r--   1 shubham supergroup    5512931 2022-11-30 23:27 project/covid_world_Data.csv
-rw-r--r--   1 shubham supergroup  415302972 2022-11-30 23:27 project/ecom_dec_2019.csv
-rw-r--r--   1 shubham supergroup  488799986 2022-11-30 23:28 project/ecom_feb_2020.csv
-rw-r--r--   1 shubham supergroup  501792804 2022-11-30 23:28 project/ecom_jan_2020.csv
-rw-r--r--   1 shubham supergroup  545839412 2022-11-30 23:28 project/ecom_nov_2019.csv
-rw-r--r--   1 shubham supergroup  482542278 2022-11-30 23:28 project/ecom_oct_2019.csv
-rw-r--r--   1 shubham supergroup   11428830 2022-11-30 23:28 project/jewelry.csv
```

- Importing numpy, pandas, matplotlib and pyspark for analysis
- Commands used to import spark into the Jupyter Notebook and initialize a Spark session

```
import sys

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
import findspark
findspark.init()
```

```
import pyspark
from pyspark.sql import SparkSession
```

```
spark = SparkSession.builder.getOrCreate()
```

6. DATA PREPARATION AND CLEANING

1. Loaded the e-com dataset as a spark data frame in jupyter notebook.
2. Converted the datatype of the variables as desired.
3. Combined five datasets (500MB each) into single data frame. This is our e-Commerce Cosmetic dataset.
4. Loaded covid dataset as a spark data frame in jupyter notebook.
5. Converted the datatype of the variables as desired.
6. We processed the covid dataset to get covid cases on each day.
7. Finally, we are also using eCom – Jewelry dataset to find effect of covid on gold sales
8. Now we have two data frames: 1. eCom for Cosmetic 2. Covid dataset 3. eCom for Jewelry

7. HOW OUR DATASET LOOKS NOW?

ECOM DATSET:

```
ecom_dataframe.show(5)
```

```
+----------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
|event_time|event_type|product_id|        category_id|category_code| brand|price|  user_id|        user_session|
+----------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
|2019-10-01|      cart|   5773203|1487580005134238553|         null|runail| 2.62|463240011|26dd6e6e-4dac-477...|
|2019-10-01|      cart|   5773353|1487580005134238553|         null|runail| 2.62|463240011|26dd6e6e-4dac-477...|
|2019-10-01|      cart|   5881589|2151191071051219817|         null|lovely|13.48|429681830|49e8d843-adf3-428...|
|2019-10-01|      cart|   5723490|1487580005134238553|         null|runail| 2.62|463240011|26dd6e6e-4dac-477...|
|2019-10-01|      cart|   5881449|1487580013522845895|         null|lovely| 0.56|429681830|49e8d843-adf3-428...|
+----------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
only showing top 5 rows
```

EFFECT OF COVID ON SALES AND E-COMMERCE

COVID DATASET:

```
covid_df = covid_df_2.select(col('Date'),col('Confirmed')).groupby('Date').agg(sum('Confirmed').alias('Total_cases'))
covid_df.show(5)
```

```
+----------+-----------+
|      Date|Total_cases|
+----------+-----------+
|2020-01-22|        557|
|2020-01-23|        657|
|2020-01-24|        944|
|2020-01-25|       1437|
|2020-01-26|       2120|
+----------+-----------+
only showing top 5 rows
```

ECOM JEWELRY DATASET:

```
ecom_jewel_df_1.show(5)
```

```
+-------------+-------------------+-------------------+--------+-------------------+---------------+--------+----
--+-------------------+------+------+-----+--------+
|Order_datetime|           Order_ID|Purchased_Product_ID|Quantity|        Category_ID|  Category_alias|Brand_ID| Pri
ce|            User_ID|Gender| Color|Metal|     Gem|
+-------------+-------------------+-------------------+--------+-------------------+---------------+--------+----
--+-------------------+------+------+-----+--------+
|    2018-12-01|1924719191579951782| 1842195256808833386|       1|1806829201890738522| jewelry.earring|       0|561.
51|1515915625207851155|  null|   red| gold| diamond|
|    2018-12-01|1924899396621697920| 1806829193678291446|       1|1806829201848795479|            null|    null|212.
14|1515915625071969944|  null|yellow| gold|    null|
|    2018-12-02|1925511016616034733| 1842214461889315556|       1|1806829201915904347| jewelry.pendant|       1| 54.
66|1515915625048493557|     f| white| gold|sapphire|
|    2018-12-02|1925626951238681511| 1835566849434059453|       1|1806829201915904347| jewelry.pendant|       0|  8
8.9|1515915625207630915|     f|   red| gold| diamond|
|    2018-12-02|1925740842841014667| 1873936840742928865|       1|1806829201924292956|jewelry.necklace|       0|417.
67|1515915625175329378|  null|   red| gold|amethyst|
+-------------+-------------------+-------------------+--------+-------------------+---------------+--------+----
--+-------------------+------+------+-----+--------+
only showing top 5 rows
```

8. QUERIES AND RESULTS

1. WHAT IS THE TOP 5 PRODUCTS SOLD EACH MONTH?

OCTOBER 2019

```
#Top 5 products sold on October
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==10).where(ecom_datafr
```

```
[Stage 221:=================================================>  (20 + 1) / 21]
```

```
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5877454|11116.790229797363|
|   5560754| 8555.360107421875|
|   5560756| 7693.780090332031|
|   5856186| 7087.600143432617|
|   5751422| 7051.799877166748|
+----------+------------------+
only showing top 5 rows
```

NOVEMBER 2019

```
#Top 5 products sold on November
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==11).where(ecom_datafram
```

```
[Stage 226:>                                                        (0 + 1) / 1]
```

```
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5560754|23116.720428466797|
|   5809910|10911.229706764221|
|   5751422| 9902.159860610962|
|     89343| 8859.299926757812|
|   5751383|  8068.58988571167|
+----------+------------------+
only showing top 5 rows
```

DECEMBER 2019

```
#Top 5 products sold on December
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==12).where(ecom_datafram
```

```
[Stage 227:=============================================================>  (20 + 1) / 21]
```

```
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5850281|10471.279907226562|
|   5560754|10110.880126953125|
|   5809910| 8693.159620285034|
|   5751422| 6000.599895477295|
|   5751383| 4489.199867248535|
+----------+------------------+
only showing top 5 rows
```

JANUARY 2020

```
#Top 5 products sold on January
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==1).where(ecom_dataframe
```

```
[Stage 230:=============================================================>  (20 + 1) / 21]
```

```
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5560754| 16527.40020751953|
|   5809910|10631.959535598755|
|   5751422|  8738.09984779358|
|   5849033| 6976.319793701172|
|   5560756| 6862.020080566406|
+----------+------------------+
only showing top 5 rows
```

FEBRUARY 2020

```
#Top 5 products sold on February
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==2).where(ecom_dataframe
```

```
[Stage 233:=============================================================>  (20 + 1) / 21]
```

```
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5560754|13416.360168457031|
|   5850281| 8542.359924316406|
|   5809910| 7336.479698181152|
|     89343| 6895.629943847656|
|   5751422| 6537.149886131287|
+----------+------------------+
only showing top 5 rows
```

From the above tables, we could see that the products ‘5560754’, ‘5751422’, ‘5809910’ are one among the top 5 each month. The performance of these products is consistent

throughout the five months. From this pattern, we couldn't find much difference on sales before and during covid.

2. WHAT ARE THE TOTAL SALES EACH MONTH?

```
#Total Sales Each Month
ecom_dataframe.where(ecom_dataframe.event_type=='purchase').groupby(month(ecom_dataframe.event_time).alias('Month')).
```

```
[Stage 52:=============================================================>  (20 + 1) / 21]
```

```
+-----+-----------+
|Month|Total_Sales|
+-----+-----------+
|   10|     245624|
|   11|     322417|
|   12|     213176|
|    1|     263797|
|    2|     241993|
+-----+-----------+
```

From the above report, we could see that the sales are more in November and less in December. From this, we don't find the effect of covid on sales.

3. WHAT ARE THE SALES ON EACH DAY OF THE MONTH?

```
total_sales.show(31)
```

```
+---+--------+--------+--------+--------+--------+
|Day|oct_sale|nov_sale|dec_sale|jan_sale|feb_sale|
+---+--------+--------+--------+--------+--------+
|  1|    8476|    7761|    7236|    3269|    7435|
|  2|    9100|    7422|    9079|    4875|    8715|
|  3|    8865|    7798|    8217|    6031|    9998|
|  4|    7562|    8053|    8371|    6602|    8937|
|  5|    5940|    8887|    7767|    7227|    9503|
|  6|    7265|    8710|    7383|    6368|    9209|
|  7|    9376|   16489|    6023|    7564|    7880|
|  8|    8604|   16628|    6595|    7602|    7216|
|  9|    8464|    5442|    9294|    8774|    8642|
| 10|    8117|    6208|    9794|    8575|    8981|
| 11|    6922|    8921|    9390|    7470|    9440|
| 12|    6431|    8184|    9396|    8147|    9563|
| 13|    7088|    8434|    7796|   10138|    9088|
| 14|    8004|    7632|    6038|   10316|    6947|
| 15|    8299|    6860|    7019|    9962|    6035|
| 16|    9096|    7029|    9226|    9354|    7404|
| 17|    9236|    7937|    8425|    9265|    9037|
| 18|    7920|    9593|    7818|    6197|    8691|
```

From analyzing each day sales, we could observe spikes of sales on each month, except February. Also, we observe declining sales in February, which will be clearer in visualization. These declining sales may be the effect of Covid.

4. WHAT ARE THE MOST AFFECTED BRANDS IN JANUARY AND FEBRUARY?

```
#Most Affected Brand IN JANUARY
ecom_dataframe.select(col('brand'),col('price')).filter(ecom_dataframe.brand.isNotNull()).where(ecom_dataframe.event_
```

```
[Stage 59:=============================================================>  (20 + 1) / 21]
```

```
+------------+------------------+
|       brand|             Sales|
+------------+------------------+
|rocknailstar| 2.380000114440918|
|       ovale| 2.859999895095825|
|      weaver|3.4100000858306885|
+------------+------------------+
only showing top 3 rows
```

```
#Most Affected Brand IN FEBRUARY
ecom_dataframe.select(col('brand'),col('price')).filter(ecom_dataframe.brand.isNotNull()).where(ecom_dataframe.event_
```

```
[Stage 62:=============================================================>  (20 + 1) / 21]
```

```
+------+------------------+
| brand|             Sales|
+------+------------------+
| rorec| 1.399999976158142|
| tazol|               2.0|
|weaver|3.4100000858306885|
+------+------------------+
only showing top 3 rows
```

5. WHAT ARE THE GOLD SALES EACH MONTH OF 2019, 2020, 2021?

```
#Gold sales every month for 2019, 2020, 2021
gold_sales_2019.join(gold_sales_2020,gold_sales_2019.month == gold_sales_2020.month,"left").join(gold_sales_2021,gold
```

```
+-----+----------+----------+----------+
|month|Count_2019|Count_2020|Count_2021|
+-----+----------+----------+----------+
|    1|       535|      1794|      3501|
|    2|       759|      1783|      3381|
|    3|       630|      1624|      4515|
|    4|       664|       556|      3765|
|    5|       658|       586|      3846|
|    6|       651|      1216|      3765|
|    7|       963|      1982|      4261|
|    8|      1045|      2830|      6594|
|    9|       757|      2222|      4616|
|   10|      1502|      2309|      3909|
|   11|      1812|      2394|      8580|
|   12|      1829|      6997|       141|
+-----+----------+----------+----------+
```

From the above table, we could observe that the gold sales keep increasing from 2019 to 2021. It is not just a slight change but could see a drastic increase of sales during covid which is unbelievable. Because of this increase in gold sales, we thought of analyzing the gold price during these periods, that might be the influencing factor.

6. HOW DOES PRICE OF GOLD PRODUCTS CHANGES EACH YEAR FROM 2019 TO 2021?

```
#Gold Price by productID for 2019, 2021
gold_price_2019.join(gold_price_2021,gold_price_2019.Purchased_Product_ID == gold_price_2021.Purchased_Product_ID,"in
```

```
+-------------------+----------+----------+
|Purchased_Product_ID|Price_2019|Price_2021|
+-------------------+----------+----------+
| 1956663848253522808|    142.33|    142.33|
| 1956663836408808274|     488.9|     488.9|
| 1956663840334676577|    623.15|    623.15|
| 1515966222670193008|    187.53|    187.53|
| 1956663846382863131|    239.59|    239.59|
| 1515966223085693366|    794.38|    794.38|
| 1956663846491915163|    291.79|    291.79|
| 1956663836803072373|    205.34|    205.34|
| 1948597376289603702|    287.53|    287.53|
| 1515966223158436127|    575.07|    575.07|
| 1956663846147981803|    136.85|    136.85|
| 1956663831090430685|     513.7|     513.7|
| 1956663831098819307|    445.07|    445.07|
| 1515966222962731904|     73.84|     73.84|
| 1886420208604676355|    150.55|    150.55|
| 1956663831300145191|    183.42|    183.42|
| 1956663840275956246|    142.33|    142.33|
| 1956663848387739700|    138.22|    138.22|
| 1956663847691485626|    236.85|    236.85|
| 1806829191128154472|    128.63|    128.63|
```

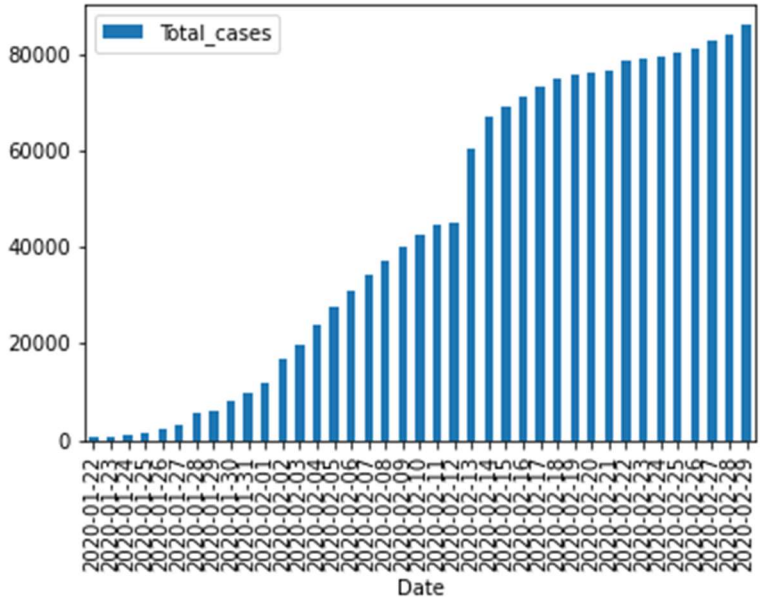
From the above table, we could observe that price of the product remaining the same for each product from 2019 to 2021.

9. VISUALIZATION

1. HOW COVID INCREASES EVERYDAY?

```
covid_df_pd.plot(x="Date", y="Total_cases", kind = "bar")
```

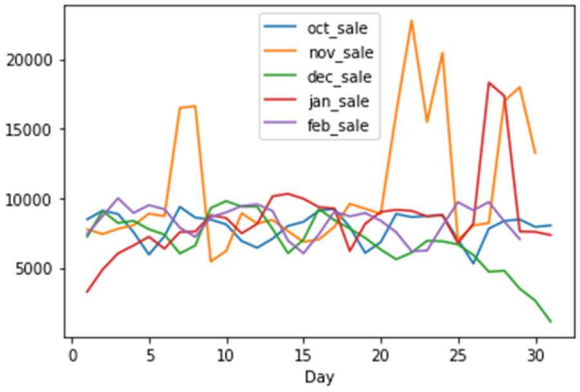
```
<AxesSubplot:xlabel='Date'>
```



2. WHAT ARE THE COSMETIC SALES EACH MONTH?

```
total_sales_pd = total_sales.select("*").toPandas()
total_sales_pd.plot(x="Day", y=["oct_sale","nov_sale","dec_sale","jan_sale","feb_sale"])
```

```
<AxesSubplot:xlabel='Day'>
```

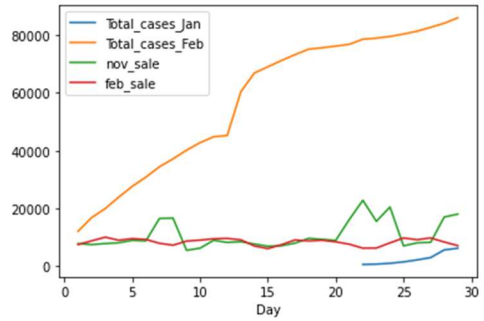



```
#Path Sales and covid
```

3. HOW COVID AFFECTS SALES IN FEBRUARY?

```
covid_sales_combined_pd = covid_sales_combined.select("*").toPandas()
covid_sales_combined_pd.plot(x="Day", y=["Total_cases_Jan","Total_cases_Feb","nov_sale","feb_sale"])
```

```
<AxesSubplot:xlabel='Day'>
```



10. FINDINGS

From the first portion of queries (Top 5 products sold each month), we see that three of the products consistently remained in the top 5 items sold each month. This indicates that even through the initial development of COVID-19, these products remained an essential necessity. While there is only a **productID** and no name/description of the product, we know that it falls under the cosmetics category. Therefore, at least one of the items may have been an item such as soap or body wash, which indicates that even in difficult economic circumstances, a company can depend on these products to continue to generate a profit. Therefore, a focus on the production of cosmetic essentials such as these during an economic downturn could be a savvy business move.

From the total sales per month portion, we see that the sales are uniform, except for the month of November. However, this may be due to the culture of the holiday season (Thanksgiving), as well as heavily discounted days such as Black Friday and Cyber Monday where there is a large yearly increase in sales. In addition, the cyclic nature of the end of the year in general will see more shopping. Therefore, it would be difficult to conclude that COVID affected the sales.

The increased sale of Gold between 2019-2021 is seen as the number of COVID cases increased. This is most likely tied in with the Fed's attempt at quantitative easing, where the stimulus of money increased the purchasing power of most individuals. Furthermore, during times of high volatility, most people seek to purchase assets that maintain their value during rapid price fluctuations. After quantitative easing, inflation rates tend to be high, which devalues most assets. Therefore, this period is followed by a Fed tapering, where inflation rates are increased every month. An asset like gold is much more resistant to these inflation and interest rates, which help people protect their wealth. This is most likely why there is such a large increase in the purchase of gold during this time period.

11. CHALLENGES FACED AND PROCESS ATTEMPTED

1. **Kafka implementation** – We tried implementing Kafka to process the data as streaming data. We started facing space issues with a very big dataset and very small storage in the ubuntu machine.

2. **AWS EMR + Spark + Jupyter**– To overcome the above issue, we thought to use cloud data storage and integrate with Spark and Jupyter. But with the limited free tier limit, we were unable to keep the service running and had to terminate the service.
3. **Hadoop + Spark + Jupyter** – Finally, we decided to use Hadoop + Spark + Jupyter framework, which we found to be an easy and powerful combination for our project needs.

12. LIMITATIONS

Dataset: The main limitation is that we were unable to find the e-commerce dataset for Jan 2019, our dataset contains records starting from Oct 2019 to Feb 2020. To accurately find the sales and covid relationship, we needed the dataset for Jan 2019. In that case, we would have been clearly able to compare and correlate the sales in Jan 2019 and Jan 2020.

13. FUTURE PLANS

We want to implement the same model using AWS Cloud and Kafka streaming.

To do the same analysis on 2019 E-commerce data to find better correlation between the Sales and Covid cases.

To analyze other areas of e-Commerce other than cosmetics and jewelry.

14. BUSINESS RECOMMENDATIONS

Since there was a clear increase in the buying of gold assets during COVID, the cosmetics industry should focus on sourcing and producing more gold and precious metal jewelry during the next economic downturn(recession) to generate more revenue.

During the month of November, e-commerce business models should spend more on advertising to increase their exposure to the market, since there is a large spike in sales (almost 50 percent) as well as increase their inventory to meet that demand. This could potentially offset loses from reduced sales incurred later during the slightly decreased sales of January-February.

A final business recommendation could be to increase the production of goods that see consistently high sales throughout the year. While obvious, it is important because in this case excess inventory is not an actual worry. Any excess inventory will be cleared indue time given that demand for the product is sustained.

EFFECT OF COVID ON SALES AND E-COMMERCE

QUERIES USED IN THIS PROJECT:

```
#Ecommerce October 2019 Data
ecom_oct_df=spark.read.load("project/ecom_oct_2019.csv",format="csv",header="true",escape="\"")
ecom_oct_df.printSchema()
ecom_oct_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_oct_df = ecom_oct_df.withColumn("product_id", ecom_oct_df["product_id"].cast(IntegerType())).withColumn("price", ecom_oct_df["price"].cast(FloatType()))
ecom_oct_df.printSchema()

from pyspark.sql.functions import *
ecom_oct_df_1 = ecom_oct_df.withColumn("event_time",to_date("event_time"))
ecom_oct_df_1.show(5)
```

```
root
 |-- event_time: string (nullable = true)
 |-- event_type: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- category_id: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- price: string (nullable = true)
 |-- user_id: string (nullable = true)
 |-- user_session: string (nullable = true)

+-------------------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
|         event_time|event_type|product_id|        category_id|category_code| brand|price|  user_id|        user_session|
+-------------------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
|2019-10-01 00:00:...|      cart|   5773203|1487580005134238553|         null|runail|  2.62|463240011|26dd6e6e-4dac-477...|
|2019-10-01 00:00:...|      cart|   5773353|1487580005134238553|         null|runail|  2.62|463240011|26dd6e6e-4dac-477...|
|2019-10-01 00:00:...|      cart|   5881589|2151191071051219817|         null|lovely| 13.48|429681830|49e8d843-adf3-428...|
|2019-10-01 00:00:...|      cart|   5723490|1487580005134238553|         null|runail|  2.62|463240011|26dd6e6e-4dac-477...|
|2019-10-01 00:00:...|      cart|   5881449|1487580013522845895|         null|lovely|  0.56|429681830|49e8d843-adf3-428...|
+-------------------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
only showing top 5 rows
```

EFFECT OF COVID ON SALES AND E-COMMERCE

```
#Ecommerce December 2019 Data
ecom_dec_df=spark.read.load("project/ecom_dec_2019.csv",format="csv",header="true",escape="\"")
ecom_dec_df.printSchema()
ecom_dec_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_dec_df = ecom_dec_df.withColumn("product_id", ecom_dec_df["product_id"].cast(IntegerType())).withColumn("price", ecom_dec_df["price"].cast(FloatType()))
ecom_dec_df.printSchema()

from pyspark.sql.functions import *
ecom_dec_df_1 = ecom_dec_df.withColumn("event_time",to_date("event_time"))
ecom_dec_df_1.show(5)
```

```
root
 |-- event_time: string (nullable = true)
 |-- event_type: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- category_id: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- price: string (nullable = true)
 |-- user_id: string (nullable = true)
 |-- user_session: string (nullable = true)

+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|          event_time|     event_type|product_id|        category_id|category_code|   brand|price|  user_id|        user_session|
+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|2019-12-01 00:00:...|remove_from_cart|   5712790|1487580005268456287|         null|   f.o.x| 6.27|576802932|51d85cb0-897f-48d...|
|2019-12-01 00:00:...|           view|   5764655|1487580005411062629|         null|     cnd|29.05|412120092|8adff31e-2051-489...|
|2019-12-01 00:00:...|           cart|      4958|1487580009471148064|         null|  runail| 1.19|494077766|c99a50e8-2fac-4c4...|
|2019-12-01 00:00:...|           view|   5848413|1487580007675986893|         null|freedecor| 0.79|348405118|722ffea5-73c0-492...|
|2019-12-01 00:00:...|           view|   5824148|1487580005511725929|         null|    null| 5.56|576005683|28172809-7e4a-45c...|
+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
only showing top 5 rows
```

```
#Ecommerce November 2019 Data
ecom_nov_df=spark.read.load("project/ecom_nov_2019.csv",format="csv",header="true",escape="\"")
ecom_nov_df.printSchema()
ecom_nov_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_nov_df = ecom_nov_df.withColumn("product_id", ecom_nov_df["product_id"].cast(IntegerType())).withColumn("price", ecom_nov_df["price"].cast(FloatType()))
ecom_nov_df.printSchema()

from pyspark.sql.functions import *
ecom_nov_df_1 = ecom_nov_df.withColumn("event_time",to_date("event_time"))
ecom_nov_df_1.show(5)
```

```
root
 |-- event_time: string (nullable = true)
 |-- event_type: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- category_id: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- price: string (nullable = true)
 |-- user_id: string (nullable = true)
 |-- user_session: string (nullable = true)

+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|          event_time|     event_type|product_id|        category_id|category_code|   brand|price|  user_id|        user_session|
+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|2019-11-01 00:00:...|           view|   5802432|1487580009286598681|         null|    null| 0.32|562076640|09fafd6c-6c99-46b...|
|2019-11-01 00:00:...|           cart|   5844397|1487580006317032337|         null|    null| 2.38|553329724|2067216c-31b5-455...|
|2019-11-01 00:00:...|           view|   5837166|1783999064103190764|         null|     pnb|22.22|556138645|57ed222e-a54a-490...|
|2019-11-01 00:00:...|           cart|   5876812|1487580010100293687|         null|jessnail| 3.16|564506666|186c1951-8052-4b3...|
```

```
#Ecommerce December 2019 Data
ecom_dec_df=spark.read.load("project/ecom_dec_2019.csv",format="csv",header="true",escape="\"")
ecom_dec_df.printSchema()
ecom_dec_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_dec_df = ecom_dec_df.withColumn("product_id", ecom_dec_df["product_id"].cast(IntegerType())).withColumn("price", ecom_dec_df["price"].cast(FloatType()))
ecom_dec_df.printSchema()

from pyspark.sql.functions import *
ecom_dec_df_1 = ecom_dec_df.withColumn("event_time",to_date("event_time"))
ecom_dec_df_1.show(5)
```

```
root
 |-- event_time: string (nullable = true)
 |-- event_type: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- category_id: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- price: string (nullable = true)
 |-- user_id: string (nullable = true)
 |-- user_session: string (nullable = true)

+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|          event_time|     event_type|product_id|        category_id|category_code|   brand|price|  user_id|        user_session|
+--------------------+---------------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|2019-12-01 00:00:...|remove_from_cart|   5712790|1487580005268456287|         null|   f.o.x| 6.27|576802932|51d85cb0-897f-48d...|
|2019-12-01 00:00:...|           view|   5764655|1487580005411062629|         null|     cnd|29.05|412120092|8adff31e-2051-489...|
|2019-12-01 00:00:...|           cart|      4958|1487580009471148064|         null|  runail| 1.19|494077766|c99a50e8-2fac-4c4...|
|2019-12-01 00:00:...|           view|   5848413|1487580007675986893|         null|freedecor| 0.79|348405118|722ffea5-73c0-492...|
|2019-12-01 00:00:...|           view|   5824148|1487580005511725929|         null|    null| 5.56|576005683|28172809-7e4a-45c...|
```

EFFECT OF COVID ON SALES AND E-COMMERCE

```
#Ecommerce January 2020 Data
ecom_jan_df=spark.read.load("project/ecom_jan_2020.csv",format="csv",header="true",escape="\"")
ecom_jan_df.printSchema()
ecom_jan_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_jan_df = ecom_jan_df.withColumn("product_id", ecom_jan_df["product_id"].cast(IntegerType())).withColumn("price", ecom_jan_df["price"].cast(FloatType()))
ecom_jan_df.printSchema()

from pyspark.sql.functions import *
ecom_jan_df_1 = ecom_jan_df.withColumn("event_time",to_date("event_time"))
ecom_jan_df_1.show(5)
```

```
root
 |-- event_time: string (nullable = true)
 |-- event_type: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- category_id: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- price: string (nullable = true)
 |-- user_id: string (nullable = true)
 |-- user_session: string (nullable = true)

+--------------------+----------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|          event_time|event_type|product_id|        category_id|category_code|   brand|price|  user_id|        user_session|
+--------------------+----------+----------+-------------------+-------------+--------+-----+---------+--------------------+
|2020-01-01 00:00:...|      view|   5809910|1602943681873052386|         null| grattol| 5.24|595414620|4adb70bb-edbd-498...|
|2020-01-01 00:00:...|      view|   5812943|1487580012121948301|         null|kinetics| 3.97|595414640|c8c5205d-be43-4f1...|
|2020-01-01 00:00:...|      view|   5798924|1783999068867920626|         null|  zinger| 3.97|595412617|46a5010f-bd69-4fb...|
|2020-01-01 00:00:...|      view|   5793052|1487580005754995573|         null|    null| 4.92|420652863|546f6af3-a517-475...|
|2020-01-01 00:00:...|      view|   5899926|2115334439910245200|         null|    null| 3.92|484071203|cff70ddf-529e-4b0...|
+--------------------+----------+----------+-------------------+-------------+--------+-----+---------+--------------------+
only showing top 5 rows
```

```
#Ecommerce February 2020 Data
ecom_feb_20_df=spark.read.load("project/ecom_feb_2020.csv",format="csv",header="true",escape="\"")
ecom_feb_20_df.printSchema()
ecom_feb_20_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_feb_20_df = ecom_feb_20_df.withColumn("product_id", ecom_feb_20_df["product_id"].cast(IntegerType())).withColumn("price", ecom_feb_20_df["price"].cast(FloatType()))
ecom_feb_20_df.printSchema()

from pyspark.sql.functions import *
ecom_feb_df_1 = ecom_feb_20_df.withColumn("event_time",to_date("event_time"))
ecom_feb_df_1.show(5)
```

```
root
 |-- event_time: string (nullable = true)
 |-- event_type: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- category_id: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- price: string (nullable = true)
 |-- user_id: string (nullable = true)
 |-- user_session: string (nullable = true)

+--------------------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
|          event_time|event_type|product_id|        category_id|category_code| brand|price|  user_id|        user_session|
+--------------------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
|2020-02-01 00:00:...|      cart|   5844305|1487580006317032337|         null|  null| 2.14|485174092|4be9643a-420b-4c6...|
|2020-02-01 00:00:...|      view|   5769925|1487580013841613016|         null|kapous| 4.22|594621622|a88baf11-9cd0-436...|
|2020-02-01 00:00:...|      view|   5817765|1487580008246412266|         null|zeitun|11.03|495404942|3a569c8d-d848-4f0...|
|2020-02-01 00:00:...|      view|   5877033|1487580010100293687|         null|  milv| 3.49|564814969|7feb39e5-bb7b-4b2...|
|2020-02-01 00:00:...|      cart|   5814871|1487580008112194531|         null|zinger| 2.54|551205603|106a7c7f-7fa1-446...|
+--------------------+----------+----------+-------------------+-------------+------+-----+---------+--------------------+
only showing top 5 rows

root
 |-- event_time: string (nullable = true)
```

EFFECT OF COVID ON SALES AND E-COMMERCE

```
covid_df=spark.read.load("project/covid_world_Data.csv",format="csv",header="true",escape="\"")
covid_df.printSchema()
covid_df.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
from pyspark.sql.functions import *
covid_df_1 = covid_df.withColumn("Date",to_date("Date"))

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
covid_df_1 = covid_df_1.withColumn("Confirmed", covid_df_1["Confirmed"].cast(IntegerType())).withColumn("Recovered", covid_df_1["Recovered"].cast(IntegerType())).withColum
covid_df_1.printSchema()
covid_df_1.show(5)
```

```
root
 |-- Date: string (nullable = true)
 |-- Country: string (nullable = true)
 |-- Confirmed: string (nullable = true)
 |-- Recovered: string (nullable = true)
 |-- Deaths: string (nullable = true)

+----------+-----------+---------+---------+------+
|      Date|    Country|Confirmed|Recovered|Deaths|
+----------+-----------+---------+---------+------+
|2020-01-22|Afghanistan|        0|        0|     0|
|2020-01-23|Afghanistan|        0|        0|     0|
|2020-01-24|Afghanistan|        0|        0|     0|
|2020-01-25|Afghanistan|        0|        0|     0|
|2020-01-26|Afghanistan|        0|        0|     0|
+----------+-----------+---------+---------+------+
only showing top 5 rows

root
 |-- Date: date (nullable = true)
 |-- Country: string (nullable = true)
```

```
#Ecommerce Jewelry Data
ecom_jewel_df=spark.read.load("project/jewelry.csv",format="csv",header="true",escape="\"")
ecom_jewel_df.printSchema()
ecom_jewel_df.show(5)

from pyspark.sql.functions import *
ecom_jewel_df_1 = ecom_jewel_df.withColumn("Order_datetime",to_date("Order_datetime"))
ecom_jewel_df_1.show(5)

from pyspark.sql.types import StringType, DateType, FloatType, IntegerType
ecom_jewel_df_1 = ecom_jewel_df_1.withColumn("Quantity", ecom_jewel_df_1["Quantity"].cast(IntegerType())).withColumn("Price", ecom_jewel_df_1["price"].cast(FloatType()))
ecom_jewel_df_1.printSchema()
```

```
root
 |-- Order_datetime: string (nullable = true)
 |-- Order_ID: string (nullable = true)
 |-- Purchased_Product_ID: string (nullable = true)
 |-- Quantity: string (nullable = true)
 |-- Category_ID: string (nullable = true)
 |-- Category_alias: string (nullable = true)
 |-- Brand_ID: string (nullable = true)
 |-- Price: string (nullable = true)
 |-- User_ID: string (nullable = true)
 |-- Gender: string (nullable = true)
 |-- Color: string (nullable = true)
 |-- Metal: string (nullable = true)
 |-- Gem: string (nullable = true)

+--------------------+------------------+--------------------+--------+-------------------+---------------+--------+------+------------------+------+------+-----+-----
---+
|      Order_datetime|          Order_ID|Purchased_Product_ID|Quantity|        Category_ID| Category_alias|Brand_ID| Price|           User_ID|Gender| Color|Metal|
Gem|
+--------------------+------------------+--------------------+--------+-------------------+---------------+--------+------+------------------+------+------+-----+-----
---+
|2018-12-01 11:40:...|1924719191579951782| 1842195256808833386|       1|1806829201890738522| jewelry.earring|       0|561.51|1515915625207851155|  null|   red| gold| diam
ond|
|2018-12-01 17:38:...|1924899396621697920| 1806829193678291446|       1|1806829201848795479|           null|    null|212.14|1515915625071969944|  null|yellow| gold|    n
ull|
|2018-12-02 13:53:...|1925511016616034733| 1842214461889315556|       1|1806829201915904347| jewelry.pendant|       1| 54.66|1515915625048493557|     f| white| gold|sapph
```

EFFECT OF COVID ON SALES AND E-COMMERCE

```
covid_df_1.columns
```

```
['Date', 'Country', 'Confirmed', 'Recovered', 'Deaths']
```

```
#Looking for covid cases in January and Febrauary 2020 and loading it to different dataframe
covid_df_2 = covid_df_1.where(year(covid_df_1.Date) == 2020).where(month(covid_df_1.Date) >= 1).where(month(covid_df_1.Date) <= 2).sort(covid_df_1.Date, ascending=True)

covid_df_2.show(5)
```

```
[Stage 25:>                                                 (0 + 1) / 1]
+----------+-----------+---------+---------+------+
|      Date|    Country|Confirmed|Recovered|Deaths|
+----------+-----------+---------+---------+------+
|2020-01-22|    Albania|        0|        0|     0|
|2020-01-22|    Algeria|        0|        0|     0|
|2020-01-22|Afghanistan|        0|        0|     0|
|2020-01-22| Antarctica|        0|        0|     0|
|2020-01-22|    Andorra|        0|        0|     0|
+----------+-----------+---------+---------+------+
only showing top 5 rows
```

```
covid_df = covid_df_2.select(col('Date'),col('Confirmed')).groupby('Date').agg(sum('Confirmed').alias('Total_cases')).sort(['Date'],ascending=True)
covid_df.show(5)
```

```
+----------+-----------+
|      Date|Total_cases|
+----------+-----------+
|2020-01-22|        557|
|2020-01-23|        657|
|2020-01-24|        944|
|2020-01-25|       1437|
|2020-01-26|       2120|
+----------+-----------+
only showing top 5 rows
```

```
covid_jan_2 = covid_df.where(month(covid_df.Date) == 1).sort(['Date'],ascending=True)
covid_jan_2 = covid_jan_2.withColumnRenamed("Total_cases","Total_cases_Jan")

covid_feb_2 = covid_df.where(month(covid_df.Date) == 2).sort(['Date'],ascending=True)
covid_feb_2 = covid_feb_2.withColumnRenamed("Total_cases","Total_cases_Feb")
```

```
covid_jan_2.show()
covid_feb_2.show()
```

```
+----------+---------------+
|      Date|Total_cases_Jan|
+----------+---------------+
|2020-01-22|            557|
|2020-01-23|            657|
|2020-01-24|            944|
|2020-01-25|           1437|
|2020-01-26|           2120|
|2020-01-27|           2929|
|2020-01-28|           5580|
|2020-01-29|           6169|
|2020-01-30|           8237|
|2020-01-31|           9927|
+----------+---------------+

+----------+---------------+
|      Date|Total_cases_Feb|
+----------+---------------+
|2020-02-01|          12038|
|2020-02-02|          16787|
|2020-02-03|          19887|
|2020-02-04|          23899|
|2020-02-05|          27644|
|2020-02-06|          30806|
|2020-02-07|          34400|
|2020-02-08|          37131|
|2020-02-09|          40162|
|2020-02-10|          42771|
|2020-02-11|          44814|
```

EFFECT OF COVID ON SALES AND E-COMMERCE

```
covid_jan_1 = covid_jan_2.withColumn("Day",date_format(col('Date'),"d"))
covid_feb_1 = covid_feb_2.withColumn("Day",date_format(col('Date'),"d"))

covid_jan = covid_jan_1.withColumn("Day", covid_jan_1["Day"].cast(IntegerType()))
covid_feb = covid_feb_1.withColumn("Day", covid_feb_1["Day"].cast(IntegerType()))
```

```
covid_feb.join(covid_jan,covid_feb.Day == covid_jan.Day,"left").select(covid_feb.Day,covid_jan.Total_cases_Jan,covid_feb.Total_cases_Feb).sort([covid_feb.Day],Ascending=Fa
```

Day	Total_cases_Jan	Total_cases_Feb
1	null	12038
2	null	16787
3	null	19887
4	null	23899
5	null	27644
6	null	30806
7	null	34400
8	null	37131
9	null	40162
10	null	42771
11	null	44814
12	null	45232
13	null	60384
14	null	66912
15	null	69055
16	null	71238
17	null	73273
18	null	75155
19	null	75655
20	null	76216
21	null	76846
22	557	78608
23	657	78990
24	944	79558
25	1437	80412
26	2120	81385

```
#Finding October 2019 Sales
oct_sales = ecom_dataframe.where(year(ecom_dataframe.event_time) == 2019).where(month(ecom_dataframe.event_time) == 10).where(ecom_dataframe.event_type == 'purchase').grou
```

```
#Finding November 2019 Sales
nov_sales = ecom_dataframe.where(year(ecom_dataframe.event_time) == 2019).where(month(ecom_dataframe.event_time) == 11).where(ecom_dataframe.event_type == 'purchase').grou
```

```
#Finding December 2019 Sales
dec_sales = ecom_dataframe.where(year(ecom_dataframe.event_time) == 2019).where(month(ecom_dataframe.event_time) == 12).where(ecom_dataframe.event_type == 'purchase').grou
```

```
#Finding January 2020 Sales
jan_sales = ecom_dataframe.where(year(ecom_dataframe.event_time) == 2020).where(month(ecom_dataframe.event_time) == 1).where(ecom_dataframe.event_type == 'purchase').group
```

```
#Finding February 2020 Sales
feb_sales = ecom_dataframe.where(year(ecom_dataframe.event_time) == 2020).where(month(ecom_dataframe.event_time) == 2).where(ecom_dataframe.event_type == 'purchase').group
feb_sales.show()
```

```
total_sales = oct.join(nov,oct.Day == nov.Day,"left").join(dec, oct.Day == dec.Day,"left").join(jan, oct.Day == jan.Day,"left").join(feb, oct.Day == feb.Day,"left").select
```

```
total_sales.show(31)
```

Day	oct_sale	nov_sale	dec_sale	jan_sale	feb_sale
1	8476	7761	7236	3269	7435
2	9100	7422	9079	4875	8715
3	8865	7798	8217	6031	9998
4	7562	8053	8371	6602	8937
5	5940	8887	7767	7227	9503
6	7265	8710	7383	6368	9209
7	9376	16489	6023	7564	7880
8	8604	16628	6595	7602	7216
9	8464	5442	9294	8774	8642
10	8117	6208	9794	8575	8981
11	6922	8921	9390	7470	9440
12	6431	8184	9396	8147	9563
13	7088	8434	7796	10138	9088
14	8004	7632	6038	10316	6947
15	8299	6860	7019	9962	6035
16	9096	7029	9226	9354	7404
17	9236	7937	8425	9265	9037
18	7920	9593	7818	6197	8691
19	6064	9243	7143	8168	8934
20	6836	8873	6308	9014	8382
21	8872	16123	5603	9165	7557
22	8643	22780	6086	9088	6189
23	8714	15483	6939	8700	6231
24	8779	20464	6901	8779	8024
25	7024	6990	6671	6766	9718
26	5295	8040	5914	8182	9126
27	7816	8209	4706	18314	9726
28	8366	16974	4786	17352	8328
29	8472	17992	3493	7605	7057
30	7935	13258	2633	7582	null

EFFECT OF COVID ON SALES AND E-COMMERCE

```
#Both Sales and covid
total_sales.join(covid_jan,total_sales.Day == covid_jan.Day,"left").join(covid_feb,covid_feb.Day == total_sales.Day,"left").select(covid_feb.Day,covid_jan.Total_cases_Jan,
```

Day	Total_cases_Jan	Total_cases_Feb	oct_sale	nov_sale	dec_sale	jan_sale	feb_sale
1	null	12038	8476	7761	7236	3269	7435
2	null	16787	9100	7422	9079	4875	8715
3	null	19887	8865	7798	8217	6031	9998
4	null	23899	7562	8053	8371	6602	8937
5	null	27644	5940	8887	7767	7227	9503
6	null	30806	7265	8710	7383	6368	9209
7	null	34400	9376	16489	6023	7564	7880
8	null	37131	8604	16628	6595	7602	7216
9	null	40162	8464	5442	9294	8774	8642
10	null	42771	8117	6208	9794	8575	8981
11	null	44814	6922	8921	9390	7470	9440
12	null	45232	6431	8184	9396	8147	9563
13	null	60384	7088	8434	7796	10138	9088
14	null	66912	8004	7632	6038	10316	6947
15	null	69055	8299	6860	7019	9962	6035
16	null	71238	9096	7029	9226	9354	7404
17	null	73273	9236	7937	8425	9265	9037
18	null	75155	7920	9593	7818	6197	8691
19	null	75655	6064	9243	7143	8168	8934
20	null	76216	6836	8873	6308	9014	8382
21	null	76846	8872	16123	5603	9165	7557
22	557	78608	8643	22780	6086	9088	6189
23	657	78990	8714	15483	6939	8700	6231
24	944	79558	8779	20464	6901	8779	8024
25	1437	80412	7024	6990	6671	6766	9718
26	2120	81385	5295	8040	5914	8182	9126
27	2929	82730	7816	8209	4706	18314	9726
28	5580	84154	8366	16974	4786	17352	8328
29	6169	86030	8472	17992	3493	7605	7057
null	8237	null	7935	13258	2633	7582	null
null	9927	null	8043	null	1126	7346	null

```
#Gold Sales in first three months of 2019, 2020, and 2021 groupby(month(ecom_dataframe.event_time)).agg(count('event_type').alias('Total_Sales')).show()
gold_sales_2019 = ecom_jewel_df_1.where(year(ecom_jewel_df_1.Order_datetime)==2019).where(ecom_jewel_df_1.Metal == 'gold').groupby(year(ecom_jewel_df_1.Order_datetime),mor
gold_sales_2020 = ecom_jewel_df_1.where(year(ecom_jewel_df_1.Order_datetime)==2020).where(ecom_jewel_df_1.Metal == 'gold').groupby(year(ecom_jewel_df_1.Order_datetime),mor
gold_sales_2021 = ecom_jewel_df_1.where(year(ecom_jewel_df_1.Order_datetime)==2021).where(ecom_jewel_df_1.Metal == 'gold').groupby(year(ecom_jewel_df_1.Order_datetime),mor
```

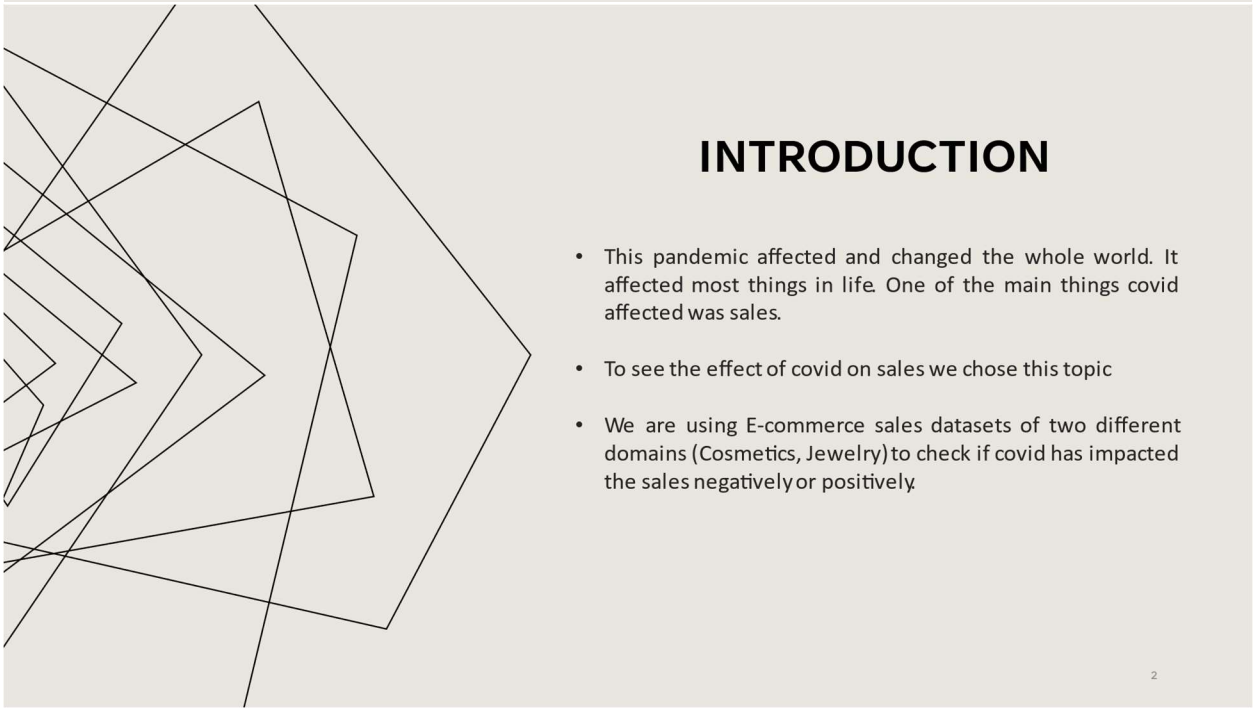
```
# Renaming the columns
gold_sales_2019 = gold_sales_2019.withColumnRenamed("year(Order_datetime)","year").withColumnRenamed("month(Order_datetime)","month")
gold_sales_2020 = gold_sales_2020.withColumnRenamed("year(Order_datetime)","year").withColumnRenamed("month(Order_datetime)","month")
gold_sales_2021 = gold_sales_2021.withColumnRenamed("year(Order_datetime)","year").withColumnRenamed("month(Order_datetime)","month")
```

```
# Changing the dataType
gold_sales_2019 = gold_sales_2019.withColumn("year", gold_sales_2019["year"].cast(IntegerType())).withColumn("month", gold_sales_2019["month"].cast(IntegerType()))
gold_sales_2020 = gold_sales_2020.withColumn("year", gold_sales_2020["year"].cast(IntegerType())).withColumn("month", gold_sales_2020["month"].cast(IntegerType()))
gold_sales_2021 = gold_sales_2021.withColumn("year", gold_sales_2021["year"].cast(IntegerType())).withColumn("month", gold_sales_2021["month"].cast(IntegerType()))
```

```
#Gold sales every month for 2019, 2020, 2021
gold_sales_2019.join(gold_sales_2020,gold_sales_2019.month == gold_sales_2020.month,"left").join(gold_sales_2021,gold_sales_2019.month == gold_sales_2021.month,"left").sel
```

month	Count_2019	Count_2020	Count_2021
1	535	1794	3501
2	759	1783	3381
3	630	1624	4515
4	664	556	3765
5	658	586	3846
6	651	1216	3765
7	963	1982	4261
8	1045	2830	6594
9	757	2222	4616
10	1502	2309	3909
11	1812	2394	8580
12	1829	6997	141

PRESENTATION SLIDES



BUSINESS QUESTIONS

COSMETICS — Does increase in covid cases affect cosmetic sales ?

JEWELRY — Does increase in covid cases affect jewelry sales ?

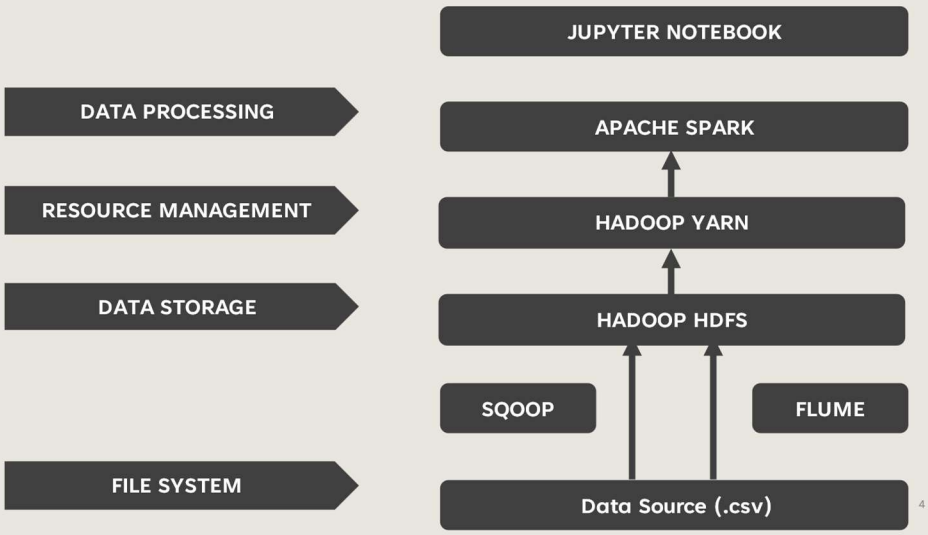
BRANDS — What are the most affected brands ?

JEWELRY PRICE — Does increase in covid cases affect jewelry price ?

3

FRAMEWORKS USED :

We used Hadoop + Spark frameworks for our project.



4

WHY THIS FRAMEWORK?

- Jupyter notebook on spark is a well balanced combination for Big Data processing and Data Analytics.
- Provides good User Interface and combines multiple works into a single file.
- It could perform everything from Data Processing and Analysis to Data Visualization.

5

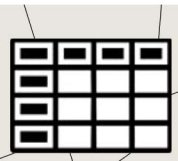
DATASETS

E-COMMERCE IN COSMETICS — https ://www .kaggle .com/datasets/mkechinov/ ecommerce -events -history -in-cosmetics - shop

E-COMMERCE IN JEWELRY — https ://www .kaggle .com/datasets/mkechinov/ ecommerce -purchase -history -from-jewelry - store

COVID — https://www.kaggle.com/datasets/gauravdutt akiit/covid -19

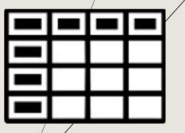
6



DATA STRUCTURE

- Total size of the datasets is around ~2.5 GB.
- E-Commerce (Cosmetic) dataset consists of data from Oct 2019 – Feb 2020.
- E-Commerce (Jewellry) dataset consists of data from Dec 2018 – Dec 2021.
- The covid dataset consists of data from Jan 2020 – Apr 2022.

7

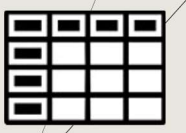


DATA STRUCTURE

- **E-Commerce Cosmetic Dataset:**

PROPERTY	DESCRIPTION
event_time	Time when event happened at (in UTC).
event_type	Only one kind of event: purchase.
product_id	ID of a product
category_id	Product's category ID
category_code	Product's category taxonomy (code name) if it was possible to make it.
brand	Downcased string of brand name
price	Float price of a product.
user_id	Permanent user ID.
user_session	Temporary user's session ID. Same for each user's session.

8

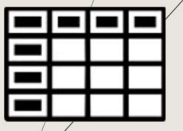


DATA STRUCTURE

- **E-Commerce Jewellery Dataset:**

PROPERTY	DESCRIPTION
OrderDate	The date when the order is placed
Order_ID	ID of the order
Product_ID	ID of a product
Quantity	The number of items purchased
Category_ID	ID of a category
Category_Alias	Name of the category
Brand_ID	ID of brand
Price	Price of the product
User_ID	ID of user
Gender	Gender of the person
Color	Color of the product
Metal	Type of metal in the product
Gem	Type of gem in the product

9



DATA STRUCTURE

- **Covid Dataset:**

PROPERTY	DESCRIPTION
Date	Date when the case was reported
Country	Contry from where the case was reported
Confirmed	Total number of cases reported
Recovered	Total number patients recovered
Deaths	Total number of deaths reported

10

DATA PREPARATION & CLEANING

- Loaded the e-com dataset as a spark data frame in Jupyter notebook
- Converted the datatype of the variables as desired.
- Combined five datasets (500MB each) into single data frame. This is our e-Commerce Cosmetic dataset.
- Loaded covid dataset as a spark data frame in Jupyter notebook.
- Converted the datatype of the variables as desired.
- We processed the covid dataset to get covid cases on each day
- Finally, we are also using E-Commerce – Jewelry dataset to find effect of covid on gold sale
- Now we have three data frames:
  - 1. E-Commerce for Cosmetic
  - 2. Covid dataset
  - 3. E-Commerce for Jewelry

11

QUERIES & RESULTS

1. What are the top 5 products sold each month?

```
#Top 5 products sold on October
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==10).where(ecom_datafr
[Stage 221:==========================================================> (20 + 1) / 21]
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5877454|11116.790229797363|
|   5560754| 8555.360107421875|
|   5560756| 7693.780090332031|
|   5856186| 7087.680143432617|
|   5751422| 7051.799877166748|
+----------+------------------+
only showing top 5 rows
```

October

```
#Top 5 products sold on November
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==11).where(ecom_datafram
[Stage 226:>                                                        (0 + 1) / 1]
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5560754|23116.720428466797|
|   5809910|10911.229706764221|
|   5751422| 9902.159860610962|
|     89343| 8859.299926757812|
|   5751383|  8068.58988571167|
+----------+------------------+
only showing top 5 rows
```

November

```
#Top 5 products sold on December
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==12).where(ecom_datafram
[Stage 227:==========================================================> (20 + 1) / 21]
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5850281|10471.279907226562|
|   5560754|10110.880126953125|
|   5809910| 8693.159628285034|
|   5751422| 6800.598095477295|
|   5751383| 6489.199867248535|
+----------+------------------+
only showing top 5 rows
```

December

- From the results obtained, we could see that the products '5560754', '5751422', '5809910' are one among the top 5 each month. The performance of these products is consistent throughout the five months. From this pattern, we couldn't find much difference on sales before and during covid.

```
#Top 5 products sold on January
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==1).where(ecom_dataframe
[Stage 238:==========================================================> (20 + 1) / 21]
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5560754| 16527.40020751953|
|   5809910|10631.959535598753|
|   5751422|  8738.899847793558|
|   5849033| 6976.319793701172|
|   5560756| 6862.020080566406|
+----------+------------------+
only showing top 5 rows
```

January

```
#Top 5 products sold on February
ecom_dataframe.select(col('product_id'),col('price')).where(month(ecom_dataframe.event_time)==2).where(ecom_dataframe
[Stage 233:==========================================================> (20 + 1) / 21]
+----------+------------------+
|product_id|             sales|
+----------+------------------+
|   5560754|13416.369186457031|
|   5850281| 8542.359924316406|
|   5809910| 7336.479690181152|
|     89343| 6895.629943847656|
|   5751422| 6537.149886131287|
+----------+------------------+
only showing top 5 rows
```

February

QUERIES & RESULTS

2. What are the total sales each month?

```
#Total Sales Each Month
ecom_dataframe.where(ecom_dataframe.event_type=='purchase').groupby(month(ecom_dataframe.event_time).alias('Month')).
```

```
[Stage 52:=======================================================>  (20 + 1) / 21]
```

```
+-----+-----------+
|Month|Total_Sales|
+-----+-----------+
|   10|     245624|
|   11|     322417|
|   12|     213176|
|    1|     263797|
|    2|     241993|
+-----+-----------+
```

- From the results obtained, We could see that the sales are more in November and less in December. From this, we don't find the effect of covid on sales.

QUERIES & RESULTS

3. What are the sales on each day of the month?

```
total_sales.show(31)
```

```
+---+--------+--------+--------+--------+--------+
|Day|oct_sale|nov_sale|dec_sale|jan_sale|feb_sale|
+---+--------+--------+--------+--------+--------+
|  1|    8476|    7761|    7236|    3269|    7435|
|  2|    9100|    7422|    9879|    4875|    8715|
|  3|    8865|    7798|    8217|    6031|    9998|
|  4|    7562|    8853|    8371|    6602|    8937|
|  5|    5940|    8887|    7767|    7227|    9503|
|  6|    7265|    8710|    7383|    6368|    9209|
|  7|    9376|   16489|    6023|    7564|    7880|
|  8|    8664|   16628|    6595|    7602|    7216|
|  9|    8464|    5442|    9294|    8774|    8642|
| 10|    8117|    6208|    9794|    8575|    8981|
| 11|    6922|    8921|    9390|    7470|    9440|
| 12|    6431|    8184|    9396|    8147|    9563|
| 13|    7088|    8434|    7796|   10138|    9088|
| 14|    8864|    7632|    6838|   10316|    6947|
| 15|    8299|    6860|    7019|    9962|    6035|
| 16|    9896|    7829|    9226|    9354|    7404|
| 17|    9236|    7937|    8425|    9265|    9037|
| 18|    7920|    9593|    7818|    6197|    8691|
```

- From analysing each day sales, we could observe spikes of sales on each month, except February. Also, we observe declining sales in February, which will be clearer in visualization. These declining sales may be the effect of Covid.

QUERIES & RESULTS

4. What are the gold sales each month of 2019, 2020, 2021?

```
#Gold sales every month for 2019, 2020, 2021
gold_sales_2019.join(gold_sales_2020,gold_sales_2019.month == gold_sales_2020.month,"left").join(gold_sales_2021,gold
```

```
+-----+----------+----------+----------+
|month|Count_2019|Count_2020|Count_2021|
+-----+----------+----------+----------+
|    1|       535|      1794|      3501|
|    2|       759|      1783|      3381|
|    3|       630|      1624|      4515|
|    4|       664|       556|      3765|
|    5|       658|       586|      3846|
|    6|       651|      1216|      3765|
|    7|       963|      1982|      4261|
|    8|      1045|      2830|      6594|
|    9|       757|      2222|      4616|
|   10|      1502|      2309|      3909|
|   11|      1812|      2394|      8580|
|   12|      1829|      6997|       141|
+-----+----------+----------+----------+
```

- From the above table, we could observe that the gold sales keep increasing from 2019 to 2021. It is not just a slight change but could see a drastic increase of sales during covid which is unbelievable. Because of this increase in gold sales, we thought of analyzing the gold price during these periods, that might be the influencing factor.

QUERIES & RESULTS

5. How does price of gold products change each year from 2019 to 2021?

```
#Gold Price by productID for 2019, 2021
gold_price_2019.join(gold_price_2021,gold_price_2019.Purchased_Product_ID == gold_price_2021.Purchased_Product_ID,"in
```

```
+--------------------+----------+----------+
|Purchased_Product_ID|Price_2019|Price_2021|
+--------------------+----------+----------+
| 1956663848253522808|    142.33|    142.33|
| 1956663836408888274|     488.9|     488.9|
| 1956663840334676577|    623.15|    623.15|
| 1515966222670193008|    187.53|    187.53|
| 1956663846382863131|    239.59|    239.59|
| 1515966223085693366|    794.38|    794.38|
| 1956663846491915163|    291.79|    291.79|
| 1956663836803072373|    205.34|    205.34|
| 1948597376289603702|    287.53|    287.53|
| 1515966223158436127|    575.07|    575.07|
| 1956663846147981803|    136.85|    136.85|
| 1956663831890430685|     513.7|     513.7|
| 1956663831098819307|    445.07|    445.07|
| 1515966222962731904|     73.84|     73.84|
| 1886420208604676355|    150.55|    150.55|
| 1956663831300145191|    183.42|    183.42|
| 1956663840275956246|    142.33|    142.33|
| 1956663848387739708|    138.22|    138.22|
| 1956663847691485626|    236.85|    236.85|
| 1806829191128154472|    128.63|    128.63|
```

- From the above table, we could observe that price of the product remaining the same for each product from 2019 to 2021.

QUERIES & RESULTS

6. WHAT ARE THE MOST AFFECTED BRANDS IN JANUARY AND FEBRUARY?

```
#Most Affected Brand IN JANUARY
ecom_dataframe.select(col('brand'),col('price')).filter(ecom_dataframe.brand.isNotNull()).where(ecom_dataframe.event_

[Stage 59:==========================================================>  (20 + 1) / 21]

+------------+------------------+
|       brand|             Sales|
+------------+------------------+
|rocknailstar| 2.380000114440918|
|       ovale| 2.859999895095825|
|      weaver|3.4100000858306885|
+------------+------------------+
only showing top 3 rows
```

```
#Most Affected Brand IN FEBRUARY
ecom_dataframe.select(col('brand'),col('price')).filter(ecom_dataframe.brand.isNotNull()).where(ecom_dataframe.event_

[Stage 62:==========================================================>  (20 + 1) / 21]

+------+------------------+
| brand|             Sales|
+------+------------------+
| rorec| 1.399999976158142|
| tazol|               2.0|
|weaver|3.4100008858306885|
+------+------------------+
only showing top 3 rows
```

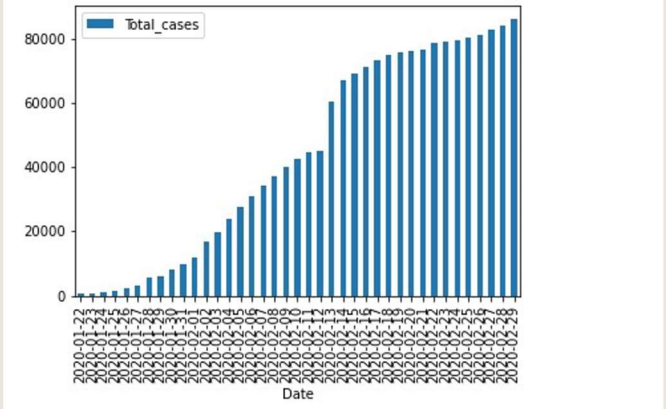
- The brands in the mentioned images are the most affected brands in the months of January and February

VISUALIZATION

1. How covid increases everyday?

```
covid_df_pd.plot(x="Date", y="Total_cases", kind = "bar")
```

```
<AxesSubplot:xlabel='Date'>
```



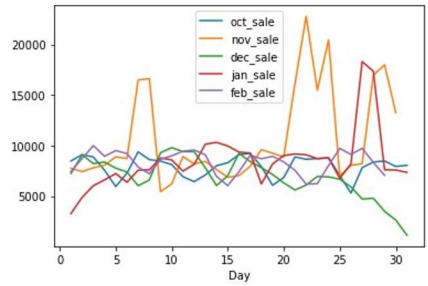
18

VISUALIZATION

2. What are the cosmetics sales each month?

```
total_sales_pd = total_sales.select("*").toPandas()
total_sales_pd.plot(x="Day", y=["oct_sale","nov_sale","dec_sale","jan_sale","feb_sale"])
```

```
<AxesSubplot:xlabel='Day'>
```



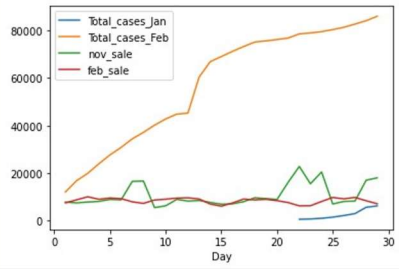
19

VISUALIZATION

3. How covid affect sales in February?

```
covid_sales_combined_pd = covid_sales_combined.select("*").toPandas()
covid_sales_combined_pd.plot(x="Day", y=["Total_cases_Jan","Total_cases_Feb","nov_sale","feb_sale"])
```

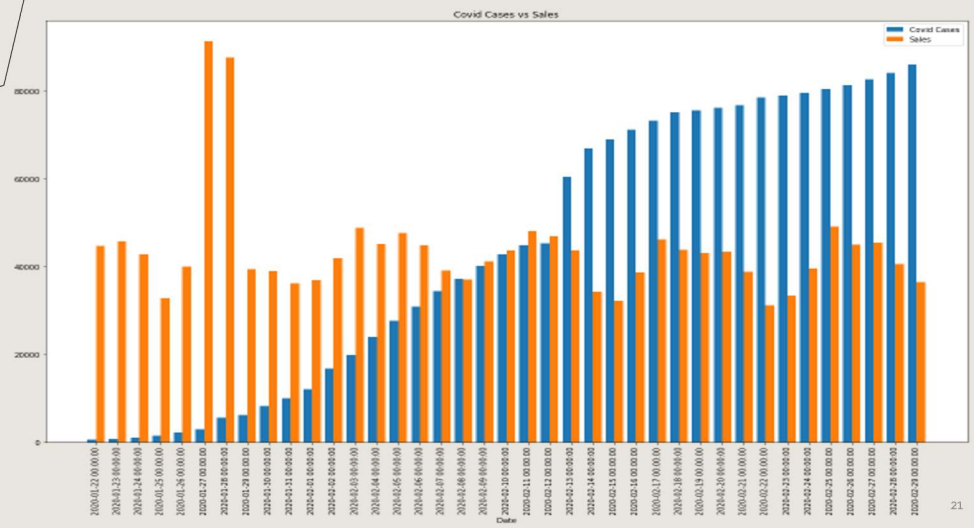
```
<AxesSubplot:xlabel='Day'>
```



20

VISUALIZATION

4. Covid Vs Sales



21

FINDINGS

- As shown in the graph from previous slides, The trends for sales is similar from October to January.We see no spike in the sales for the month Feb. Which indicates a decrease in sales due to Covid.
- In case of Cosmetic Data, by comparing sales of each month, the sales in November are high this might be because of external factors like Thanksgiving or pre-covid stock up. We didn't find any Covid cases data affecting the Cosmetic Sales.
- For the Jewelry dataset, we could obtain that sale of gold is increasing instantly after covid, this is very unusual, with the price of gold remaining constant every year.

22

CHALLENGES FACED & PROCESS ATTEMPTED

- Kafka Implementation
- AWS EMR + Spark + Jupyter
- Hadoop + Spark + Jupyter

23

LIMITATIONS

Dataset :

- The main limitation is that we were unable to find the e - commerce dataset for Jan 2019.
- our dataset contains records starting from Oct 2019 to Feb 2020. To accurately find the sales and covid relationship, we needed the dataset for Jan 2019.
- In that case, we would have been clearly able to compare and correlate the sales in Jan 2019 and Jan 2020.

24

FUTURE PLANS

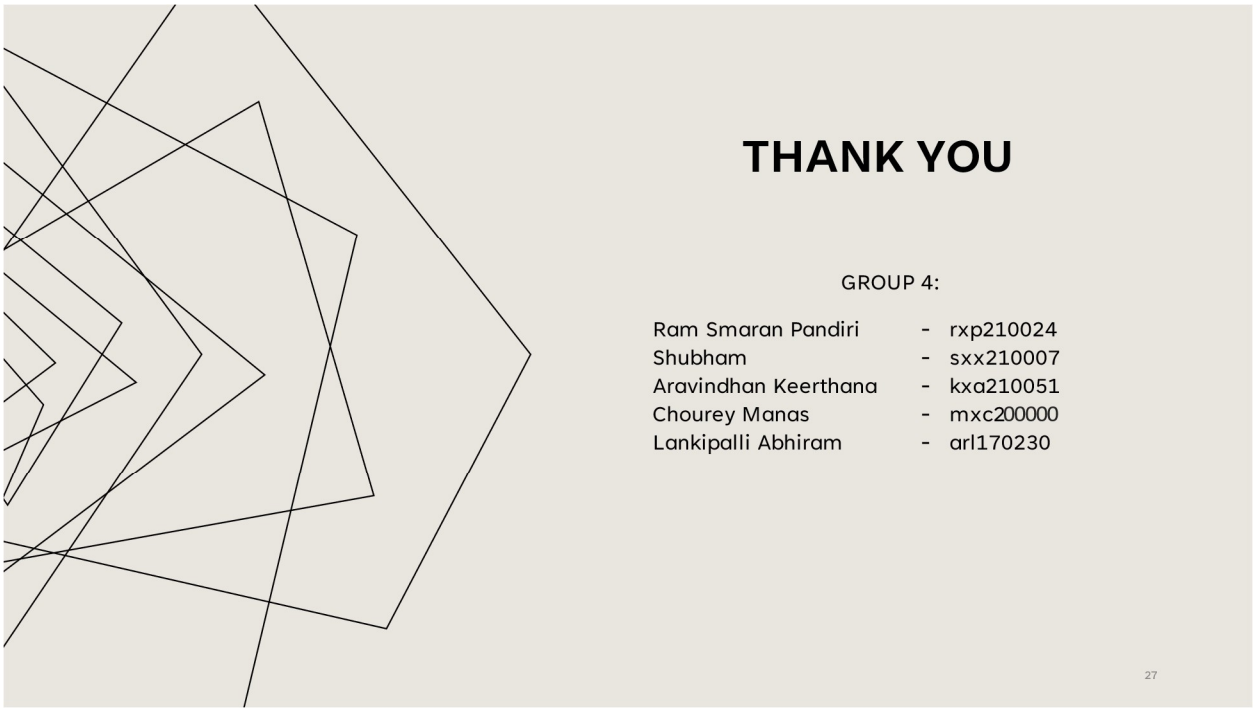
- We want to implement the same model using AWS Cloud and Kafka streaming.
- Find dataset related to 2019 Ecommerce data in order to find better correlation between the Sales and Covid cases.

25

CONCLUSION

- From the insights we have obtained, Covid had a negative affect in the month of February on sales. We would be more precise on this with more data on hand

26


THANK YOU
GROUP 4:
Ram Smaran Pandiri - rxp210024
Shubham - sxx210007
Aravindhan Keerthana - kxa210051
Chourey Manas - mxc200000
Lankipalli Abhiram - arl170230
27