

## Project Part 1:

---

### Introduction and problem description

Initially, it was difficult to find out products having a high response rate as compared to other products. Due to this, it was difficult to decide on which products we should move our marketing focus to improve response and sales. Since customer rating is a very important parameter which can impact the purchase decision of future customers. Hence if some products are getting poor rating then those products need more focus and promotion. To find out these products, I have decided to analyze the amazon reviews database to gather information related to products amazon is offering and ratings those products are getting from customers. After analyzing Amazon Reviews Database, I realized for few product categories we are having very low ratings as compared to other product categories. My analysis involves the below steps:

### Data Cleaning

In the Amazon Review database there are many records with multiple reviews by the same users for the same product. It is not appropriate to use this data for analysis, it may cause misinterpretation. Hence it is better to exclude such data from the database by creating a filter\_view and excluded records having multiple reviews by the same users for the same product. Also, it is unreliable to use old data for analysis. So for analysis, I have included the data after 2005 and the data having product categories such as Wireless, Automotive, Music, Digital\_Music\_Purchase, Sports, Toys, Digital\_Video\_Games, Video\_Games.

### Create a View with Excluded data:

```
CREATE view filter_view AS
SELECT *
FROM amazon_review.amazon_reviews_parquet
WHERE review_id IN
  (SELECT x.review_id
   FROM
     (SELECT customer_id,
            product_id,
            review_id,
            count(*)
     FROM amazon_review.amazon_reviews_parquet
     GROUP BY customer_id, product_id, review_id
     HAVING (count(*)) =1)as x)and product_category IN
('wireless', 'Automotive', 'Music', 'Digital_Music_Purchase', 'Sports', 'Toys', 'Digital_Video_Games', 'Video_Games');
```

```

at org.apache.hadoop.hive ql.parse.ParseUtils.parse(ParseUtils.java:70)
at org.apache.hadoop.hive ql.Driver.compile(Driver.java:468)
at org.apache.hadoop.hive ql.Driver.compileInternal(Driver.java:1317)
at org.apache.hadoop.hive ql.Driver.runInternal(Driver.java:1457)
at org.apache.hadoop.hive ql.Driver.run(Driver.java:1237)
at org.apache.hadoop.hive ql.Driver.run(Driver.java:1227)
at org.apache.hadoop.hive cli.CliDriver.processLocalCmd(CliDriver.java:233)
at org.apache.hadoop.hive cli.CliDriver.processCmd(CliDriver.java:184)
at org.apache.hadoop.hive cli.CliDriver.processLine(CliDriver.java:403)
at org.apache.hadoop.hive cli.CliDriver.executeDriver(CliDriver.java:821)
at org.apache.hadoop.hive cli.CliDriver.run(CliDriver.java:759)
at org.apache.hadoop.hive cli.CliDriver.main(CliDriver.java:686)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.hadoop.util.RunJar.run(RunJar.java:229)
at org.apache.hadoop.util.RunJar.main(RunJar.java:153)
FAILED: ParseException line 5:5 cannot recognize input near '(' 'SELECT' 'review_id' in expression specification
hive> CREATE view filter_view AS
> SELECT *
> FROM amazon_review.amazon_reviews_parquet
> WHERE review_id IN
> (SELECT x.review_id
> FROM
> (SELECT customer_id,
> product_id,
> review_id,
> count(*)
> FROM amazon_review.amazon_reviews_parquet
> GROUP BY customer_id, product_id, review_id
> HAVING (count(*)=1) as x) and product_category IN ('Wireless','Automotive','Music','Digital_Music_Purchase','Sports','Toys',
> 'Digital_Video_Games','Video_Games');
OK
marketplace customer_id review_id product_id product_parent product_title star_rating helpful_votes total_vot
es vine verified_purchase review_headline review_body review_date year product_category
Time taken: 0.725 seconds
hive>

```

## Create a Table with Excluded data:

```

CREATE TABLE amazon_review.amazon_review_filtered_data
AS
SELECT x.* from
(SELECT *,
row_number()
OVER (partition by customer_id, product_id) AS row_num
FROM filter_view)x
WHERE row_num=1;

```

```

> ro jml
> OV (p ion by customer_id, prc t_id) AS row_num
> FROM f )x
> WHERE row
> WITH DATA:
> AS
> SE T x.* am
> SELEC , dm
> ro ()
> OV (p ition by customer_id, product_id) AS row_num
> FROM f er ew)x
> SE row =1
> = hadoo 32 0921 9_0b4165fc-04e8-41a7-af24-e9b46ad04392
Tot: bs = 1
Laur ng Job 1 o
Tez sion was e
Ses n re-estab ne
Sta : Running ic on YARN cluster with App id application_1586465628541_0011)

-----
VERTICES STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED
-----
p 1 ..... tai SUCCEEDED 13 13 0 0 0 0
p 3 ..... tai SUCCEEDED 13 13 0 0 0 0
ducer 2 ..... tai SUCCEEDED 43 43 0 0 0 0
ducer 4 ..... tai SUCCEEDED 39 39 0 0 0 0
-----
RTICES: 04/04 === =====>] 100% ELAPSED TIME: 1213.15 s
-----
ving data to tor dfs://ip-172-31-39-148.ec2.internal:8020/user/hive/warehouse/amazon_review.db/amazon_review_filtered_data
marketplace str _id x.review_id x.product_id x.product_parent x.product_title x.star_rating x.helpful_vote
otal_votes x ne verified_purchase x.review_headline x.review_body x.review_date x.year x.product_category
ow num
ne taken: 2.25 se ds
ve>

```

## Basic Exploratory Analysis:

Carried out basic exploratory analysis to understand a basic overview of the Amazon Review Database.

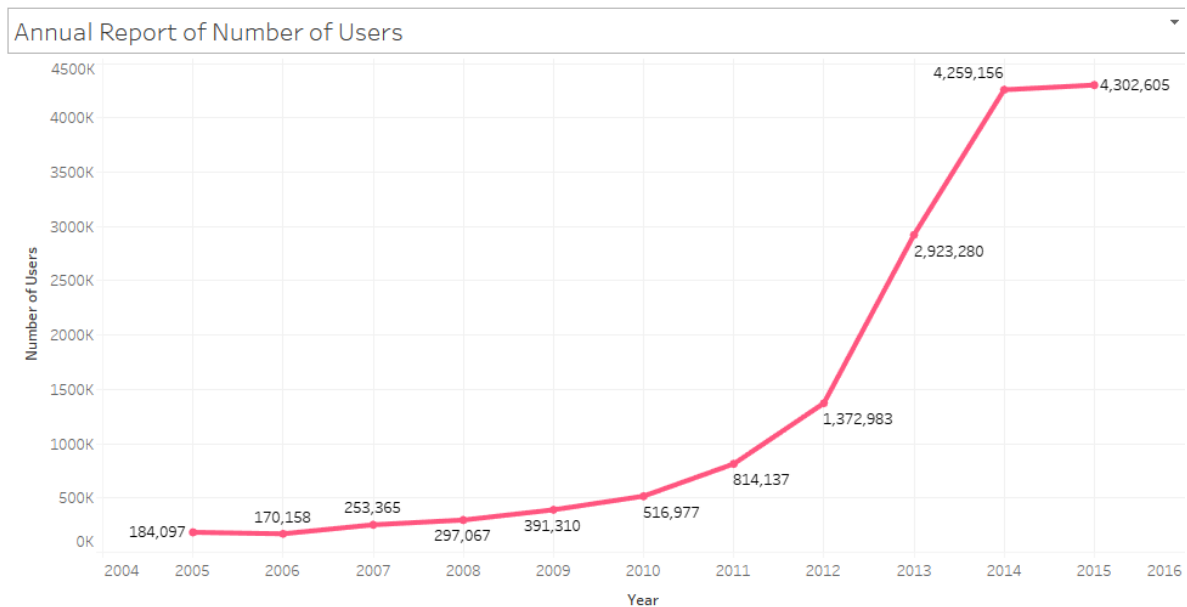
Calculated different parameters like Number\_of\_Reviews, Number\_of\_Users, Average\_Review\_Stars, Avg\_Length\_of\_Review, Verified\_Users, Unverified\_Users, Total\_Helpful\_Votes, Total\_Products.

```
SELECT year,
       count(review_id) AS Number_of_Reviews,
       count(distinct(customer_id)) AS Number_of_Users,
       round(avg(star_rating),
2) AS Average_Review_Stars,
       round(avg(length(review_body)),
2) AS Avg_Length_of_Review,
       sum(case
WHEN verified_purchase = 'Y' then 1
ELSE 0 end) AS Verified_Users, sum(case
WHEN verified_purchase='N' THEN
1
ELSE 0 end) AS Unverified_Users, sum(case
WHEN helpful_votes= 1 THEN
1
ELSE 0 end) AS Total_Helpful_Votes, count(distinct(product_id)) AS
Total_Products
FROM amazon_review.amazon_review_filtered_data
WHERE year>=2005
GROUP BY year
ORDER BY year;
```

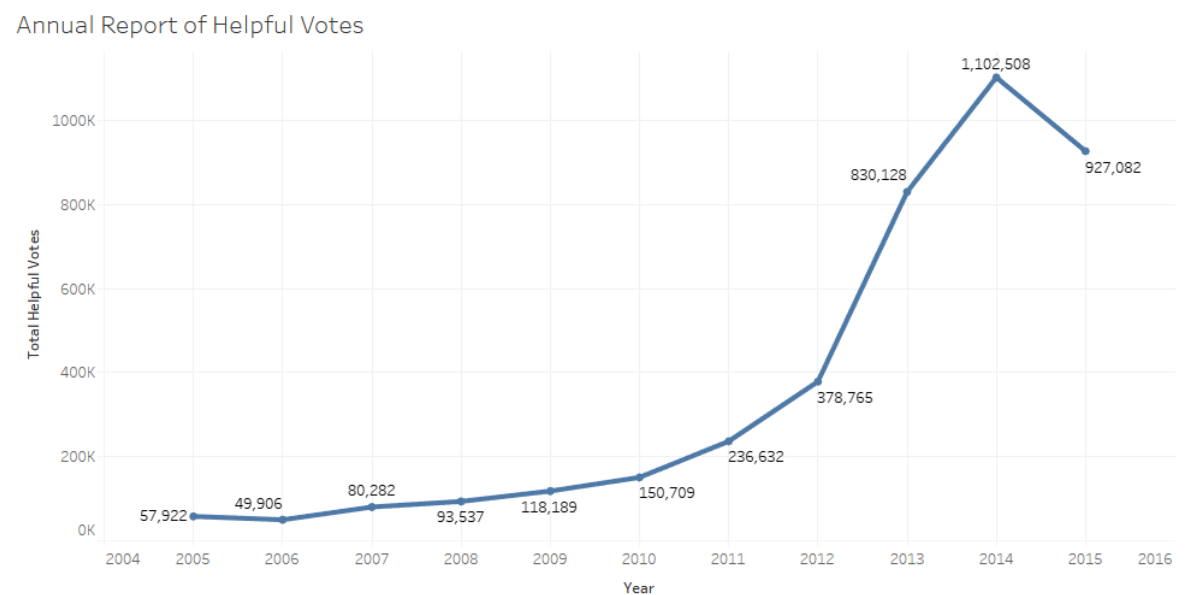
```
> 1
> ELSE 0 end) AS Total_Helpful_Votes, count(distinct(product_id)) AS Total_Products
> FROM amazon_review.amazon_review_filtered_data
> WHERE year>=2005
> GROUP BY year
> ORDER BY year;
Query ID = hadoop_20200409222200_7808d220-bd8a-44ff-a250-80918ac62193
Total jobs = 1
Launching job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1586465628541_0012)

-----
VERTICES    MODE        STATUS      TOTAL    COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED   17       17           0         0         0         0
Reducer 2 ... container  SUCCEEDED   20       20           0         0         0         0
Reducer 3 ... container  SUCCEEDED    1         1           0         0         0         0
-----
VERTICES: 03/03 [=====] 100% ELAPSED TIME: 295.46 s
-----
OK
 year  number_of_reviews  number_of_users  average_review_stars  avg_length_of_review  verified_users  unverified_users  total_helpful_votes  total_products
-----
2005   335321  184098  4.14  909.23  26830  308491  57937  116576
2006   287544  170161  4.15  896.61  38457  249087  49904  118278
2007   403645  253364  4.2   710.97  105276  298369  80279  168786
2008   458048  297063  4.13  691.9  148056  318992  93535  201060
2009   587617  391308  4.11  640.98  259708  327909  118183  254358
2010   762126  516976  4.04  617.88  515795  246331  150708  321031
2011   1227220  814134  4.01  557.53  905100  322120  236621  475102
2012   2245354  1372987  4.08  442.7  1816143  429211  378772  785837
2013   5481096  2022277  4.14  313.39  4887392  593704  838125  1461673
2014   8517649  4258154  4.19  220.88  7240688  127581  1102507  2011962
2015   8545325  4302597  4.22  166.83  7914191  631134  927077  2018222
Time taken: 306.273 seconds, Fetched: 11 row(s)
hive>
```

**VISUALIZATION:**

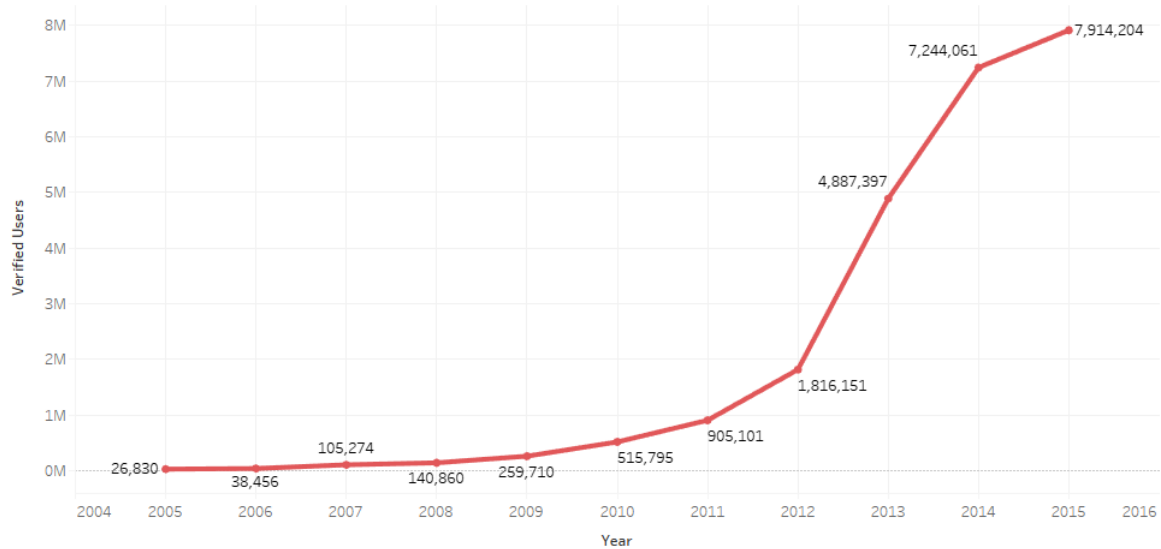


From this graph, we can analyze the number of customers increased gradually from 2008 till 2014 but after 2014, there is no significant growth in customer count.



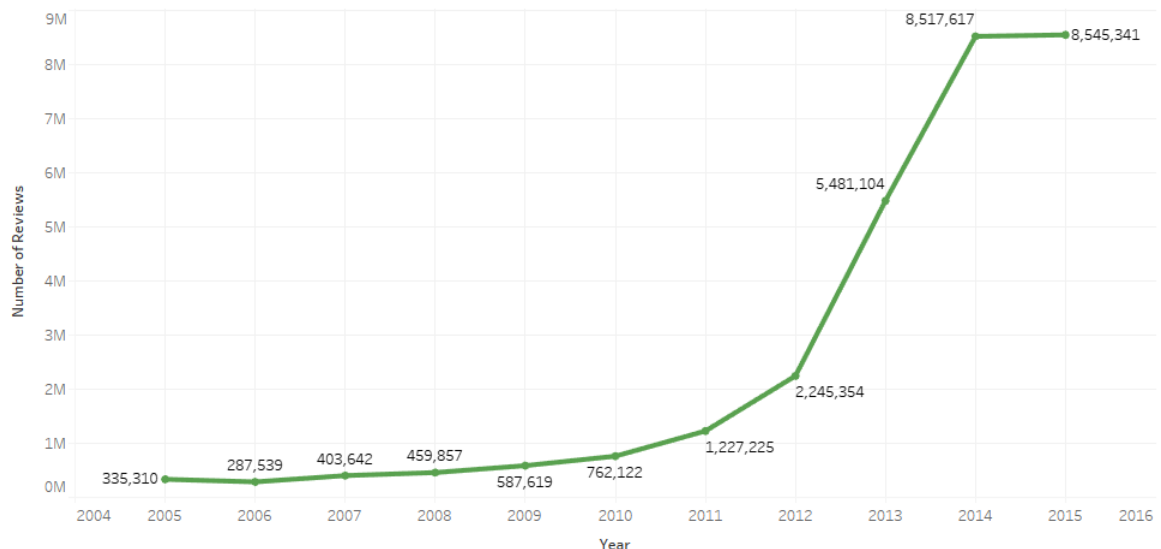
from this graph, number of helpful votes increased from 2005 to 2014. But after 2014, number of helpful votes decreased drastically.

Annual Report of Verified Users



From this graph, we can see number of verified users increased gradually from 2005 to 2015.

Annual Report of Number of Reviews



From this graph, we can interpret number of reviews increased gradually from 2005 to 2012. After 2012, count increased drastically till 2014. After 2014, there is slight decrease in number of reviews.

## Detailed analysis of Music/Digital\_Music\_Purchase and Digital\_Video\_Games/Video\_Games over time.

Performed detailed analysis to find out if there is a correlation between different product categories like

1. music and Digital\_Music\_Purchase
2. Video\_Games and Digital\_Video\_Games.

### Correlation between the categories over time

#### Analysis of Music Related Category

```

SELECT year,
       sum(case
         WHEN product_category = 'Music' then 1
         ELSE 0 end) AS music_customers, sum(case
         WHEN product_category = 'Digital_Music_Purchase' THEN
         1
         ELSE 0 end) AS digital_music_customers
FROM amazon_review.amazon_review_filtered_data
WHERE year >= 2005
GROUP BY year
ORDER BY year;

```

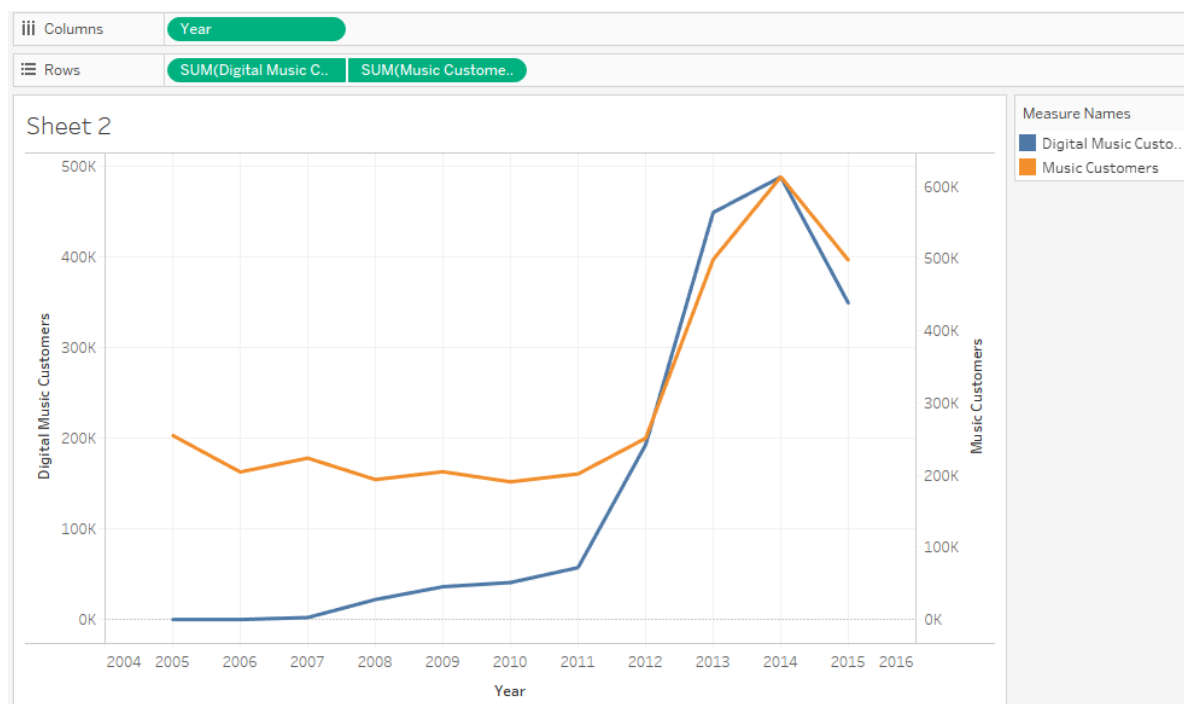
```

> WHEN product_category = 'Music' then 1
> ELSE 0 end) AS music_customers, sum(case
> WHEN product_category = 'Digital_Music_Purchase' THEN
> ELSE 0 end) AS digital_music_customers
> FROM amazon_review.amazon_review_filtered_data
> WHERE year >= 2005
> GROUP BY year
> ORDER BY year;
Query ID = hadoop_20200409223008_fe5be249-4eb9-4c8f-908f-9ec3434003a8
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1586465628541_0012)

-----
VERTICES   MODE           STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED  17      17           0         0         0         0
Reducer 2 ..... container  SUCCEEDED  20      20           0         0         0         0
Reducer 3 ..... container  SUCCEEDED   1         1           0         0         0         0
-----
VERTICES: 03/03 [=====] 100% ELAPSED TIME: 165.05 s
-----
OK
year  music_customers  digital_music_customers
2005  255102           8
2006  204397          21
2007  223646          2235
2008  193912          22040
2009  204798          36182
2010  190792          40889
2011  201767          57245
2012  251107          192641
2013  498750          448609
2014  618325          487911
2015  497941          348712
Time taken: 165.678 seconds, Fetched: 11 row(s)
hive>

```

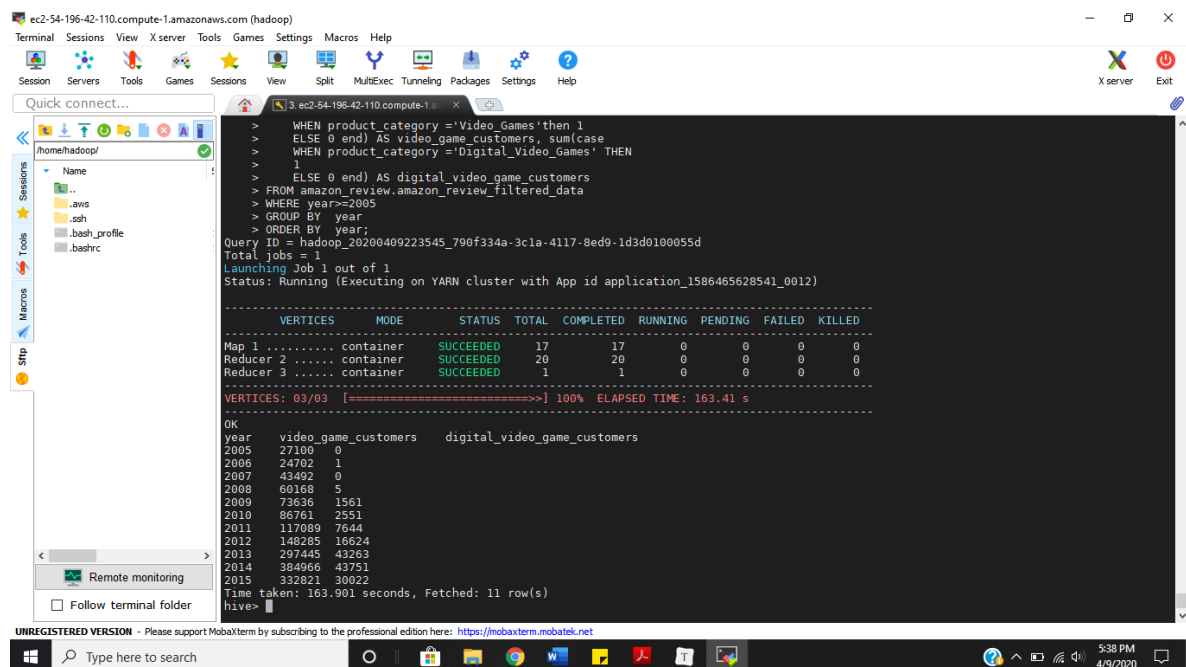
## VISUALIZATION:



For the music category, there was no drastic change in the number of customers from 2005 to 2012 but after 2012 to 2014 the number of customers increased but after 2014, the number of customers decreased drastically. Similarly, for the Digital Music category, initially, in 2005 there was not a single customer who reviewed digital music products but from 2007 number of customers started increasing till 2014.

### Analysis on Game Related Category

```
SELECT year,
       sum(case
         WHEN product_category = 'video_games' then 1
         ELSE 0 end) AS video_game_customers, sum(case
         WHEN product_category = 'Digital_Video_Games' THEN
         1
         ELSE 0 end) AS digital_video_game_customers
FROM amazon_review.amazon_review_filtered_data
WHERE year >= 2005
GROUP BY year
ORDER BY year;
```

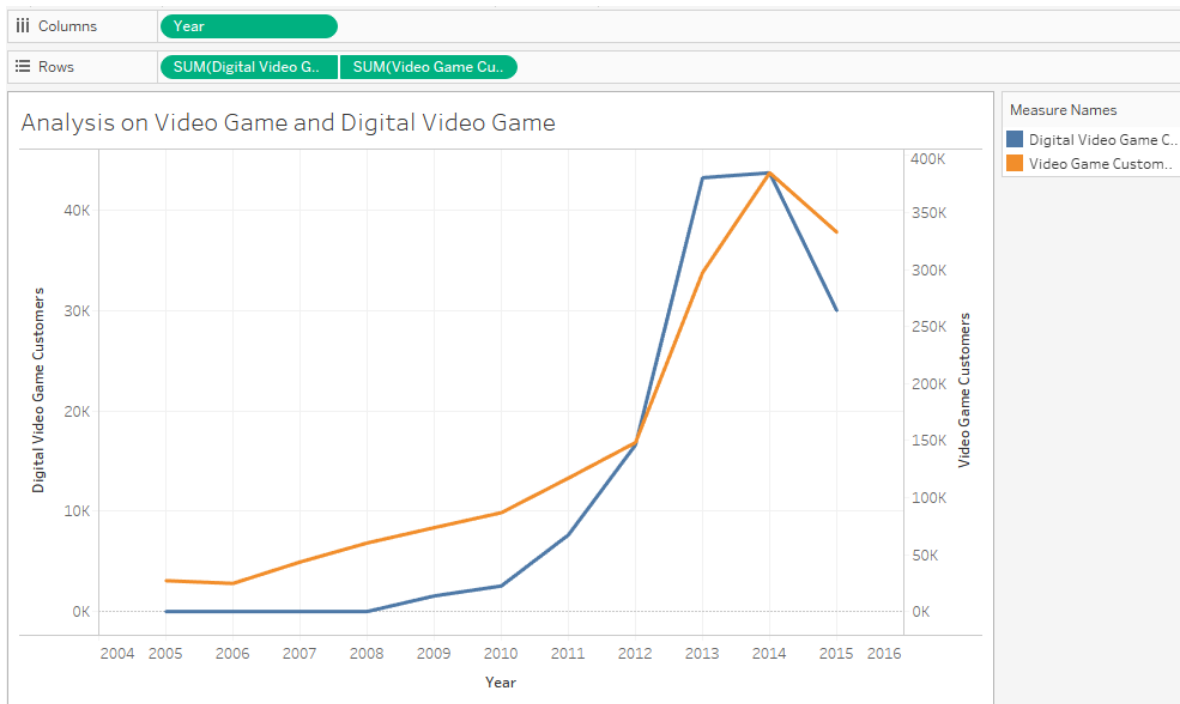


```
ec2-54-196-42-110.compute-1.amazonaws.com (hadoop)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
/home/hadoop/
Name
.aws
.ssh
.bash_profile
.bashrc
Sftp
Remote monitoring
Follow terminal folder

> WHEN product_category = 'Video Games' then 1
> ELSE 0 end) AS video_game_customers, sum(case
> WHEN product_category = 'Digital_Video_Games' THEN
> 1
> ELSE 0 end) AS digital_video_game_customers
> FROM amazon_review.amazon_review_filtered_data
> WHERE year >= 2005
> GROUP BY year
> ORDER BY year;
Query ID = hadoop_20200409223545_790f334a-3c1a-4117-8ed9-1d3d0100055d
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1586465628541_0012)

-----
VERTICES      MODE      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED  17      17          0         0         0         0
Reducer 2 ..... container  SUCCEEDED  20      20          0         0         0         0
Reducer 3 ..... container  SUCCEEDED   1         1          0         0         0         0
-----
VERTICES: 03/03 [=====] 100% ELAPSED TIME: 163.41 s
-----
OK
year  video_game_customers  digital_video_game_customers
2005  27100  0
2006  24702  1
2007  43492  0
2008  60168  5
2009  73636  1561
2010  86761  2551
2011  117089  7644
2012  148285  16624
2013  297445  43263
2014  384966  43751
2015  332821  30022
Time taken: 163.901 seconds, Fetched: 11 row(s)
hive>
```

### VISUALIZATION:



For the Video\_Game category, the number of customers gradually increased from 2005 to 2012. After 2012 to 2014 the number of customers increased drastically. Similarly, for the Digital Music category, initially, from 2005 to 2008 there was not a single customer reviewing digital music products but after 2008 till 2014 number of customers increased gradually. After 2014, the number of customers started decreasing.

## Are there the same users reviewing in both categories?

To find out whether there are same customer who are reviewing for products in both categories:

### Analysis of music related categories:

```
SELECT count(x.customer_id) AS count,
       x.year
FROM amazon_review.amazon_review_filtered_data x,
     (SELECT distinct(customer_id)
      FROM amazon_review.amazon_review_filtered_data
      WHERE product_category='Music'
        AND year>=2005 intersect SELECT distinct(customer_id)
      FROM amazon_review.amazon_review_filtered_data
      WHERE product_category='Digital_Music_Purchase'
        AND year>=2005)y
WHERE x.customer_id=y.customer_id
     AND x.year>=2005
     AND x.product_category IN ('Music','Digital_Music_Purchase')
GROUP BY year
ORDER BY year;
```





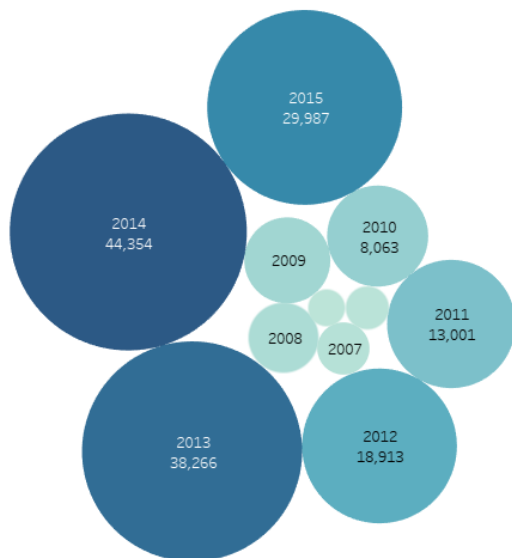
```

> GROUP BY year
> ORDER BY year;
Query ID = hadoop_20200409230129_57c852f1-0343-4146-a498-fc491762588e
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1586465628541_0013)

-----
VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED  17      17           0         0         0         0
Map 4 ..... container  SUCCEEDED  17      17           0         0         0         0
Map 8 ..... container  SUCCEEDED  17      17           0         0         0         0
Reducer 2 ..... container  SUCCEEDED  11      11           0         0         0         0
Reducer 3 ..... container  SUCCEEDED   1         1           0         0         0         0
Reducer 5 ..... container  SUCCEEDED  10      10           0         0         0         0
Reducer 7 ..... container  SUCCEEDED   5         5           0         0         0         0
Reducer 9 ..... container  SUCCEEDED  10      10           0         0         0         0
-----
VERTICES: 08/08 [=====] 100% ELAPSED TIME: 468.76 s
-----
OK
count  x.year
1082   2005
1484   2006
2158   2007
3883   2008
5842   2009
8063   2010
13091  2011
18913  2012
38266  2013
44350  2014
29991  2015
Time taken: 477.17 seconds, Fetched: 11 row(s)
hive>

```

Annual Analysis of Customers reviewing both Video\_Games and Digital Video Games



SUM(Count)	
1,082	44,354

From the above bubble chart, we can conclude that in 2014 the number of customers reviewing both Video\_Games and Digital video game categories was more as compared to other years.

## Total Number of same users reviewing in both categories

Users reviewing Music and Digital\_Music\_Purchase categories:

```

SELECT count(x.customer_id) AS count
FROM amazon_review.amazon_review_filtered_data x,
     (SELECT distinct(customer_id)
      FROM amazon_review.amazon_review_filtered_data
      WHERE product_category='Music'
           AND year>=2005 intersect SELECT distinct(customer_id)
      FROM amazon_review.amazon_review_filtered_data
      WHERE product_category='Digital_Music_Purchase'
           AND year>=2005)y
WHERE x.customer_id=y.customer_id
      AND x.year>=2005
      AND x.product_category IN ('Music','Digital_Music_Purchase');

```

```

ec2-54-196-42-110.compute-1.amazonaws.com (hadoop)
Terminal Sessions View X server Tools Games Settings Macros Help
Quick connect...
/home/hadoop/
Name
aws
ssh
.bash_profile
.bashrc
Remote monitoring
Follow terminal folder
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

44350 2014
29991 2015
Time taken: 477.17 seconds, Fetched: 11 row(s)
hive> SELECT count(x.customer_id) AS count
> FROM amazon_review.amazon_review_filtered_data x,
> (SELECT distinct(customer_id)
> FROM amazon_review.amazon_review_filtered_data
> WHERE product_category='Music'
> AND year>=2005 intersect SELECT distinct(customer_id)
> FROM amazon_review.amazon_review_filtered_data
> WHERE product_category='Digital_Music_Purchase'
> AND year>=2005)y
> WHERE x.customer_id=y.customer_id
> AND x.year>=2005
> AND x.product_category IN ('Music','Digital_Music_Purchase');
Query ID = hadoop_20200409231027_082399f8-d076-40e6-a96f-9e0fda03f3e7
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1586465628541_0013)

-----
VERTICES      MODE      STATUS      TOTAL      COMPLETED      RUNNING      PENDING      FAILED      KILLED
-----
Map 1 ..... container      SUCCEEDED      17           17           0           0           0           0
Map 3 ..... container      SUCCEEDED      17           17           0           0           0           0
Map 7 ..... container      SUCCEEDED      17           17           0           0           0           0
Reducer 2 ..... container      SUCCEEDED      1           1           0           0           0           0
Reducer 4 ..... container      SUCCEEDED      10          10           0           0           0           0
Reducer 6 ..... container      SUCCEEDED      5           5           0           0           0           0
Reducer 8 ..... container      SUCCEEDED      10          10           0           0           0           0
VERTICES: 07/07 [=====] 100% ELAPSED TIME: 474.26 s
-----
OK
count
1270137
Time taken: 475.038 seconds, Fetched: 1 row(s)
hive>

```

## Results

	count
1	1270134

Users reviewing video\_games and Digital\_Video\_Games categories:

```

SELECT count(x.customer_id) AS count
FROM amazon_review.amazon_review_filtered_data x,
(SELECT distinct(customer_id)
FROM amazon_review.amazon_review_filtered_data
WHERE product_category='Video_Games'
AND year>=2005 intersect SELECT distinct(customer_id)
FROM amazon_review.amazon_review_filtered_data
WHERE product_category='Digital_Video_Games'
AND year>=2005)y
WHERE x.customer_id=y.customer_id
AND x.year>=2005
AND x.product_category IN ('video_games','Digital_Video_Games');

```

```

ec2-107-22-54-24.compute-1.amazonaws.com (hadoop)
Terminal Sessions View X server Tools Games Settings Macros Help
Quick connect...
/home/hadoop/
Name
aws
ssh
.bash_profile
.bashrc
Remote monitoring
Follow terminal folder
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Moving data to directory hdfs://ip-172-31-43-185.ec2.internal:8020/user/hive/warehouse/amazon_review.db/amazon_review_filtered_data
OK
Time taken: 727.208 seconds
hive> SELECT count(x.customer_id) AS count
> FROM amazon_review.amazon_review_filtered_data x,
> (SELECT distinct(customer_id)
> FROM amazon_review.amazon_review_filtered_data
> WHERE product_category='Video Games'
> AND year>=2005 intersect SELECT distinct(customer_id)
> FROM amazon_review.amazon_review_filtered_data
> WHERE product_category='Digital_Video_Games'
> AND year>=2005)y
> WHERE x.customer_id=y.customer_id
> AND x.year>=2005
> AND x.product_category IN ('Video Games','Digital_Video_Games');
Query ID = hadoop_20200419005349_21ab0460-eba5-40b9-a8af-fc152512f294
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1586735534229_0010)

-----
VERTICES    MODE    STATUS    TOTAL    COMPLETED    RUNNING    PENDING    FAILED    KILLED
-----
Map 1 ..... container    SUCCEEDED    28        28            0            0            0            0
Map 3 ..... container    SUCCEEDED    28        28            0            0            0            0
Map 7 ..... container    SUCCEEDED    28        28            0            0            0            0
Reducer 2 ... container    SUCCEEDED     1           1            0            0            0            0
Reducer 4 ... container    SUCCEEDED    10          10            0            0            0            0
Reducer 6 ... container    SUCCEEDED     5           5            0            0            0            0
Reducer 8 ... container    SUCCEEDED    10          10            0            0            0            0
-----
VERTICES: 07/07 [=====] 100% ELAPSED TIME: 129.66 s
-----
OK
167033
Time taken: 130.77 seconds, Fetched: 1 row(s)
hive>

```

## Results

	count
1	167033

Can you identify similar items in both categories? Do they get the same rating?

Performed analysis to find out whether there are similar items in both categories having the same rating.

```

CREATE view music_category AS
(SELECT product_id,
round(avg(star_rating),
2) AS Music_Ranking
FROM amazon_review.amazon_review_filtered_data
WHERE product_category='Music'
AND year>= 2005
GROUP BY product_id);

```

```

CREATE view digital_music_category AS
(SELECT product_id,
round(avg(star_rating),
2) AS Digital_Music_Ranking
FROM amazon_review.amazon_review_filtered_data
WHERE product_category='Digital_Music_Purchase'
AND year>= 2005
GROUP BY product_id);

```

```

CREATE view game_category AS
(SELECT product_id,
round(avg(star_rating),

```

```

2) AS Game_Ranking
FROM amazon_review.amazon_review_filtered_data
WHERE product_category='Video_Games'
      AND year>= 2005
GROUP BY product_id);

```

```

CREATE view digital_game_category AS
(SELECT product_id,
      round(avg(star_rating),
2) AS Digital_Game_Ranking
FROM amazon_review.amazon_review_filtered_data
WHERE product_category='Digital_Video_Games'
      AND year>= 2005
GROUP BY product_id);

```

```

ec2-54-196-42-110.compute-1.amazonaws.com (hadoop)
Terminal Sessions View X server Tools Games Settings Macros Help
Quick connect...
/home/hadoop/
Name
.aws
.ssh
.bash_profile
.bashrc
Remote monitoring
Follow terminal folder
Time taken: 0.206 seconds
hive>
>
> CREATE view digital_music_category AS
> (SELECT product_id,
>      round(avg(star_rating),
>      2) AS Digital_Music_Ranking
>      FROM amazon_review.amazon_review_filtered_data
>      WHERE product_category='Digital_Music_Purchase'
>      AND year>= 2005
>      GROUP BY product_id);
>
OK
Time taken: 0.157 seconds
hive>
>
> CREATE view game_category AS
> (SELECT product_id,
>      round(avg(star_rating),
>      2) AS Game_Ranking
>      FROM amazon_review.amazon_review_filtered_data
>      WHERE product_category='Video_Games'
>      AND year>= 2005
>      GROUP BY product_id);
>
OK
Time taken: 0.162 seconds
hive>
>
> CREATE view digital_game_category AS
> (SELECT product_id,
>      round(avg(star_rating),
>      2) AS Digital_Game_Ranking
>      FROM amazon_review.amazon_review_filtered_data
>      WHERE product_category='Digital_Video_Games'
>      AND year>= 2005
>      GROUP BY product_id);
>
OK
Time taken: 0.143 seconds
hive>

```

**Analysis between Music and Digital music category:**

```

SELECT m.product_id ,
      Music_Ranking,
      Digital_Music_Ranking
FROM music_category m
INNER JOIN digital_music_category dm
      ON m.product_id=dm.product_id;

```

```

hive> CREATE view digital_game_category AS
> (SELECT product_id,
> round(avg(star_rating),
> 2) AS Digital_Game_Ranking
> FROM amazon_review.amazon_review_filtered_data
> WHERE product category='Digital_Video_Games'
> AND year>= 2005
> GROUP BY product_id);
OK
Time taken: 0.143 seconds
hive> SELECT m.product_id ,
> Music_Ranking,
> Digital_Game_Ranking
> FROM music_category m
> INNER JOIN digital_game_category dm
> ON m.product_id=dm.product_id;
Query ID = hadoop_20200409233154_fa69f963-f008-4773-9ceb-21170087a101
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1586465628541_0014)

-----
VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED  17      17           0         0         0         0
Map 4 ..... container  SUCCEEDED  17      17           0         0         0         0
Reducer 2 .... container  SUCCEEDED  10      10           0         0         0         0
Reducer 3 .... container  SUCCEEDED  10      10           0         0         0         0
Reducer 5 .... container  SUCCEEDED  10      10           0         0         0         0
-----
VERTICES: 05/05 [=====] 100% ELAPSED TIME: 320.73 s
-----
OK
B0019M1ZJS      3.0      5.0
Time taken: 321.456 seconds, Fetched: 1 row(s)
hive> +g

```

From this, we can interpret that we have the same item in both Music and Digital\_Music\_Purchase categories with different ratings.

### Analysis between Video\_Games and Digital\_Game\_Ranking category:

```

SELECT g.product_id ,
       Game_Ranking,
       Digital_Game_Ranking
FROM game_category g
INNER JOIN digital_game_category dg
ON g.product_id=dg.product_id;

```

```

Reducer 5 ..... container  SUCCEEDED  10      10           0         0         0         0
VERTICES: 05/05 [=====] 100% ELAPSED TIME: 320.73 s
-----
OK
B0019M1ZJS      3.0      5.0
Time taken: 321.456 seconds, Fetched: 1 row(s)
hive> SELECT g.product_id ,
> Game_Ranking,
> Digital_Game_Ranking
> FROM game_category g
> INNER JOIN digital_game_category dg
> ON g.product_id=dg.product_id;
Query ID = hadoop_20200410000148_97479ac2-6fa6-40f3-b297-59742ce70b10
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1586465628541_0015)

-----
VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED  17      17           0         0         0         0
Map 4 ..... container  SUCCEEDED  17      17           0         0         0         0
Reducer 2 .... container  SUCCEEDED  10      10           0         0         0         0
Reducer 3 .... container  SUCCEEDED  10      10           0         0         0         0
Reducer 5 .... container  SUCCEEDED  10      10           0         0         0         0
-----
VERTICES: 05/05 [=====] 100% ELAPSED TIME: 306.22 s
-----
OK
B0047T7MEW      4.5      3.69
B00NB8ME0Y      3.47     5.0
B004YNI0Y       4.0      2.92
B00B4WYTUS      5.0      4.64
Time taken: 315.741 seconds, Fetched: 4 row(s)
hive>

```

From this, we can interpret that we have the same items in both Music and Digital\_Music\_Purchase categories with a different rating.

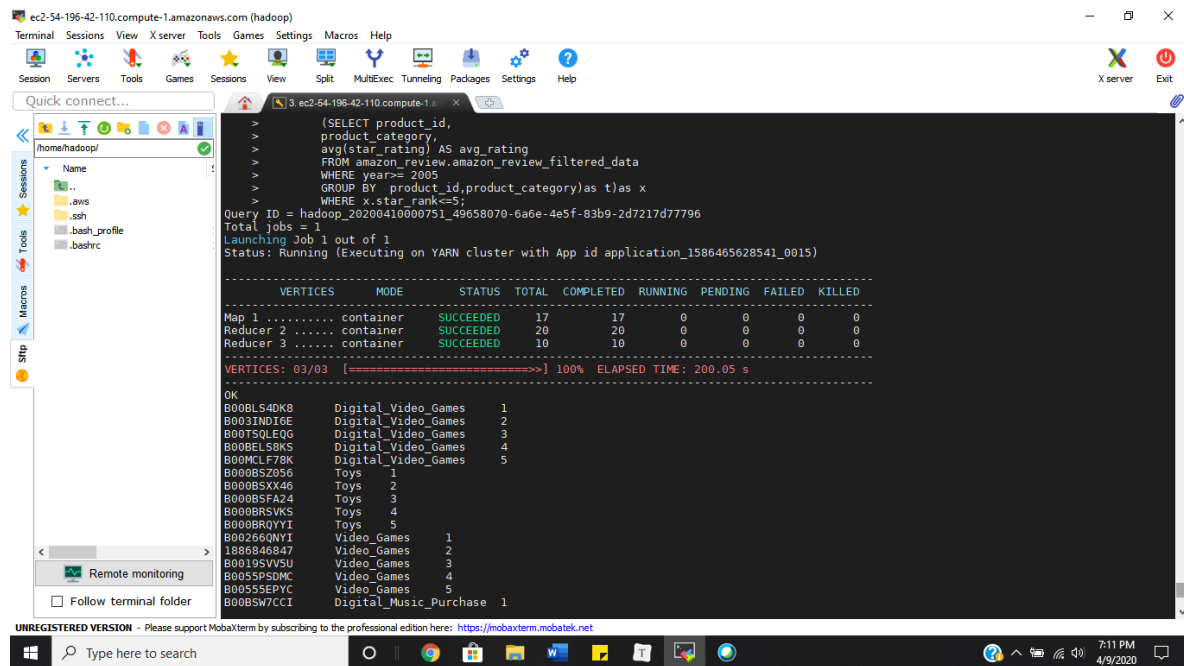
## Hive Advanced Functions

### Ranking based on products under different product categories

Calculated rank of different products in a specific product category to find out popular products and least rated products in that category.

```
SELECT x.product_id,
       x.product_category,
       x.star_rank

FROM
  (SELECT t.product_id,
         t.product_category,
         Row_number()
         OVER (partition by t.product_category
              ORDER BY t.avg_rating desc) AS star_rank
   FROM
     (SELECT product_id,
            product_category,
            avg(star_rating) AS avg_rating
      FROM amazon_review.amazon_review_filtered_data
     WHERE year >= 2005
      GROUP BY product_id, product_category) as t) as x
 WHERE x.star_rank <= 5;
```



The screenshot shows a MobaXterm window with a terminal session. The terminal displays the execution of a Hadoop job. The job is running on a YARN cluster with App id application\_1586465628541\_0015. The job is currently running and has completed 100% of the vertices.

The output shows the following table of vertices and their status:

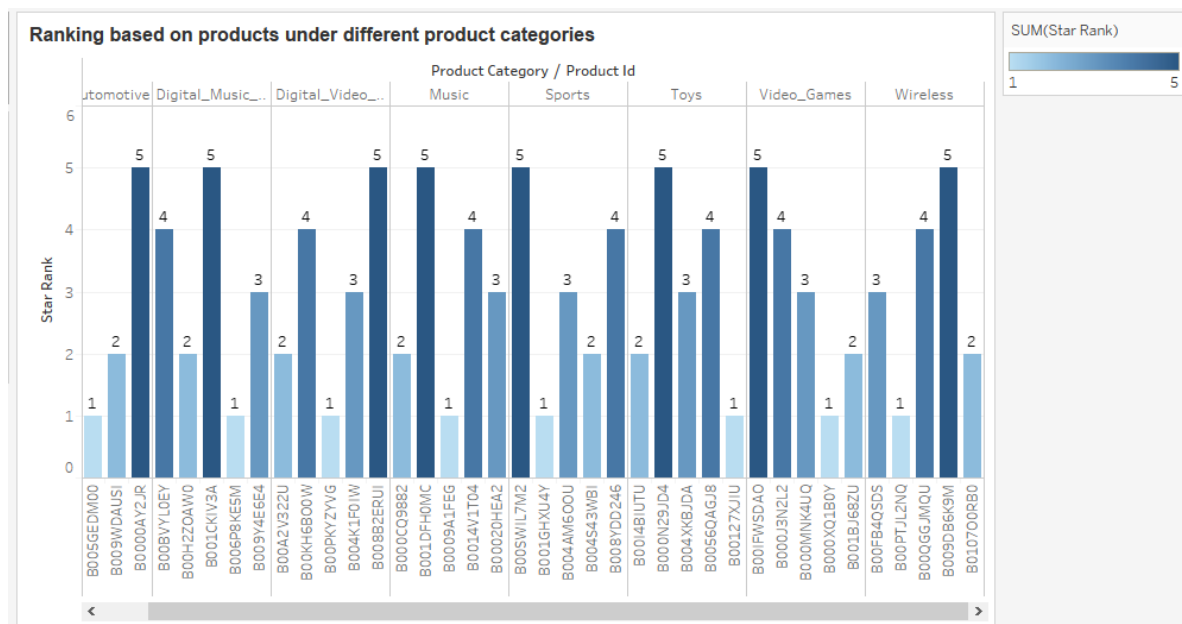
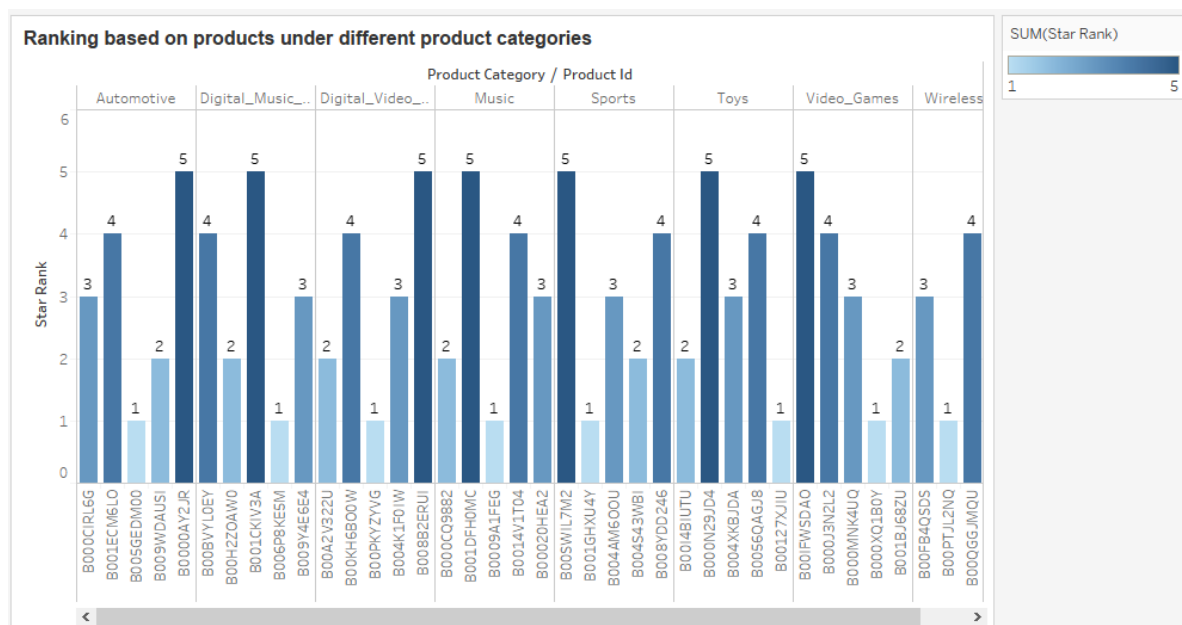
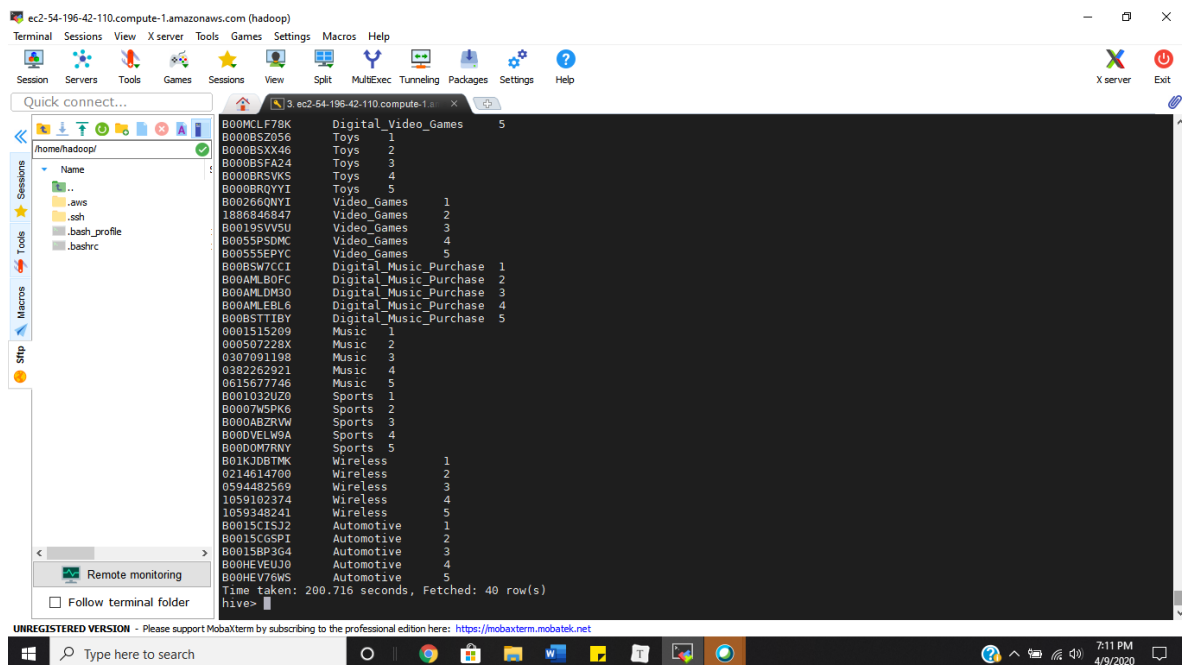
VERTICES	MODE	STATUS	TOTAL	COMPLETED	RUNNING	PENDING	FAILED	KILLED
Map 1	container	SUCCEEDED	17	17	0	0	0	0
Reducer 2	container	SUCCEEDED	20	20	0	0	0	0
Reducer 3	container	SUCCEEDED	10	10	0	0	0	0

The output also shows the following table of vertices and their status:

VERTICES	MODE	STATUS	TOTAL	COMPLETED	RUNNING	PENDING	FAILED	KILLED
Map 1	container	SUCCEEDED	17	17	0	0	0	0
Reducer 2	container	SUCCEEDED	20	20	0	0	0	0
Reducer 3	container	SUCCEEDED	10	10	0	0	0	0

The output also shows the following table of vertices and their status:

VERTICES	MODE	STATUS	TOTAL	COMPLETED	RUNNING	PENDING	FAILED	KILLED
Map 1	container	SUCCEEDED	17	17	0	0	0	0
Reducer 2	container	SUCCEEDED	20	20	0	0	0	0
Reducer 3	container	SUCCEEDED	10	10	0	0	0	0



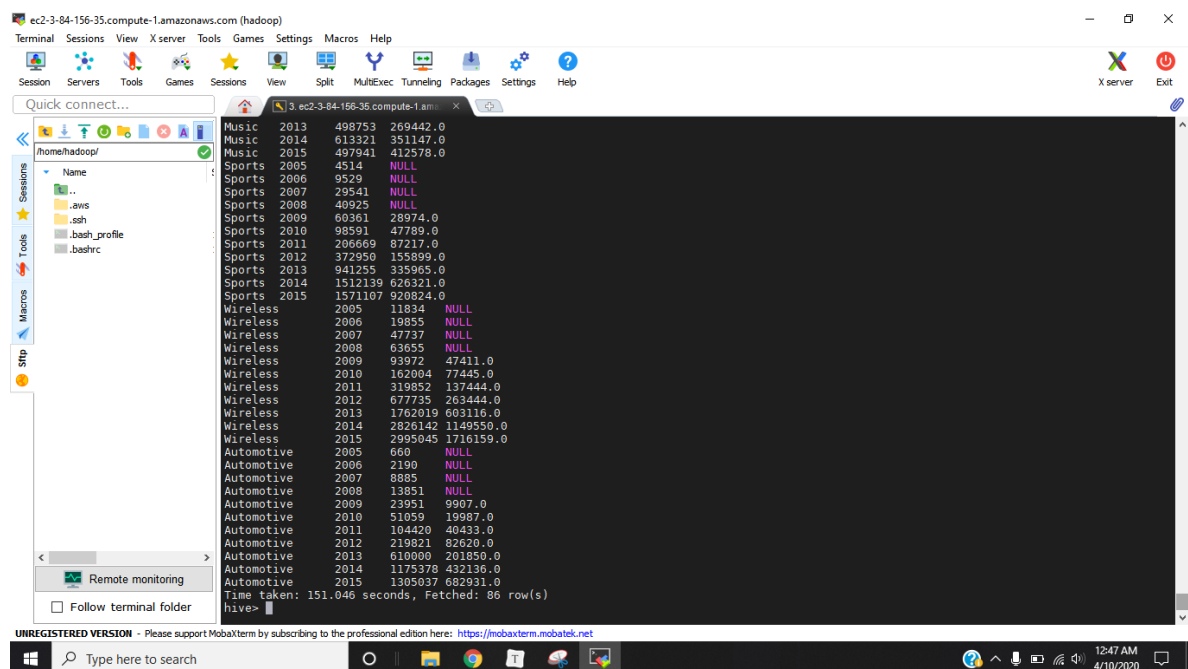
From the above bar graph, we can see products represented in light color have a low rating as compared to products represented in a darker shade. From this, we can find out the products on which we need to focus more.



## Compared the growth of the products by calculating moving average:

Calculated moving average for product categories to evaluate growth.

```
SELECT x.product_category,x.year,x.count,
      (case
      WHEN row_number()
      OVER (partition by x.product_category order by x.year) > 4 THEN
      round(AVG(x.count)
      OVER (PARTITION BY x.product_category
      ORDER BY x.year ROWS 4 PRECEDING))) end) AS five_year_moving_avg
FROM
      (SELECT product_category,
      year,
      count(review_id) AS count
      FROM amazon_review.amazon_review_filtered_data
      GROUP BY product_category,year
      ORDER BY year desc) AS x
WHERE x.year>= 2005;
```



Growth of the products by Moving Average



## Calculated Standard Deviation to analyze normal distribution of star rating of particular product category

```
SELECT year,
        product_category,
        round(stddev(star_rating),2) as Standard_Deviation,
        round(avg(star_rating),2) as Avg_Rating
FROM amazon_review.amazon_review_filtered_data
WHERE year >= 2005
GROUP BY product_category, year;
```

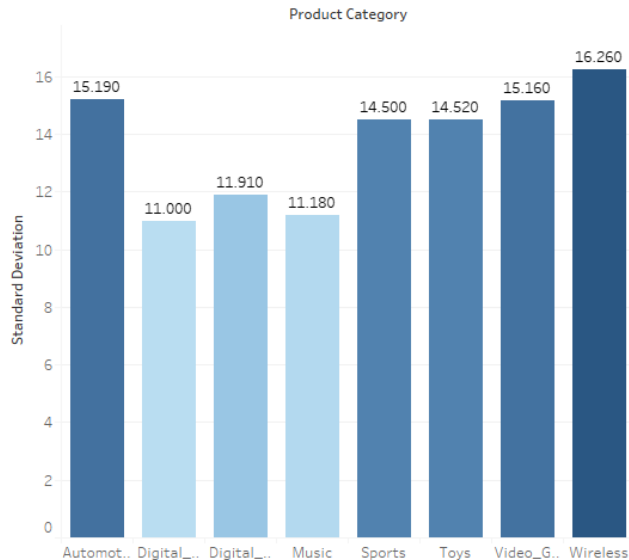
```
ec2-3-86-62-103.compute-1.amazonaws.com (hadoop)
Terminal Sessions View X server Tools Games Settings Macros Help
Quick connect...
/home/hadoop/
Name
.aws
.ssh
.bash_profile
.bashrc
Remote monitoring
Follow terminal folder

> round(stddev(star_rating),2) as Standard_Deviation,
> round(avg(star_rating),2) as Avg_Rating
> FROM amazon_review.amazon_review_filtered_data
> WHERE year >= 2005
> GROUP BY product_category, year;
Query ID = hadoop_20200410214500_bff47fef-7377-4139-a183-f1da8e0be558
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1586549970210_0010)

-----
VERTICES   MODE      STATUS    TOTAL    COMPLETED  RUNNING  PENDING  FAILED  KILLED
Map 1 ..... container  SUCCEEDED 28       28         0        0        0        0
Reducer 2 ..... container  SUCCEEDED 20       20         0        0        0        0
-----
VERTICES: 02/02 [=====] 100% ELAPSED TIME: 92.14 s
-----
OK
year      product_category      standard_deviation      avg_rating
2010      Digital_Video_Games    1.44      3.73
2010      Sports      1.35      4.0
2012      Automotive      1.33      4.14
2012      Toys      1.32      4.09
2013      Toys      1.23      4.22
2013      Video_Games      1.35      4.1
2015      Sports      1.2      4.3
2015      Video_Games      1.3      4.24
2008      Sports      1.32      4.05
2009      Sports      1.35      4.01
2011      Wireless      1.52      3.67
2012      Video_Games      1.41      3.95
2008      Automotive      1.41      3.97
2008      Wireless      1.43      3.77
2010      Digital_Music_Purchase 1.05      4.49
2015      Wireless      1.44      3.99
```

```
2008      Digital_Music_Purchase 1.08      4.46
2008      Music      1.05      4.37
2011      Video_Games      1.4      3.89
2013      Automotive      1.26      4.21
2014      Sports      1.22      4.25
2005      Video_Games      1.45      3.77
2008      Video_Games      1.44      3.79
2015      Digital_Video_Games    1.5      4.03
2006      Sports      1.42      3.9
2007      Digital_Music_Purchase 1.14      4.41
2009      Automotive      1.42      3.98
2009      Music      1.04      4.39
2009      Toys      1.32      4.01
2010      Automotive      1.39      4.0
2010      Video_Games      1.4      3.86
2014      Digital_Video_Games    1.51      3.94
2005      Toys      1.47      3.82
2009      Wireless      1.47      3.72
2010      Toys      1.36      3.97
2011      Sports      1.3      4.09
2005      Sports      1.53      3.69
2007      Video_Games      1.31      3.96
2015      Digital_Music_Purchase 0.79      4.7
2005      Digital_Music_Purchase 0.7      4.63
2005      Music      1.17      4.27
2006      Automotive      1.53      3.75
2009      Video_Games      1.35      3.92
2011      Music      1.03      4.42
2013      Sports      1.21      4.21
2011      Digital_Music_Purchase 1.08      4.48
2014      Video_Games      1.32      4.17
2015      Music      0.87      4.62
2006      Toys      1.47      3.81
2009      Digital_Video_Games    1.38      3.89
2011      Automotive      1.37      4.06
2014      Wireless      1.46      3.92
Time taken: 102.442 seconds, Fetched: 86 row(s)
hive>
```

### Standard Deviation to analyze normal distribution of star rating



SUM(Standard Deviation)

11.000 16.260

### Maximum Rating of a product in specific product category:

Calculated maximum rating of a product in a specific product category to find out the most popular product in that category.

```
SELECT product_category,
       round(avg(star_rating),
             2) AS avg_rating
FROM amazon_review.amazon_review_filtered_data
WHERE year >= 2005
GROUP BY product_category
HAVING avg(star_rating) in (
    (SELECT max(x.avg_stars)
     FROM
        (SELECT product_category,
                avg(star_rating) AS avg_stars
         FROM amazon_review.amazon_review_filtered_data
          WHERE year >= 2005
          GROUP BY product_category) AS x));
```

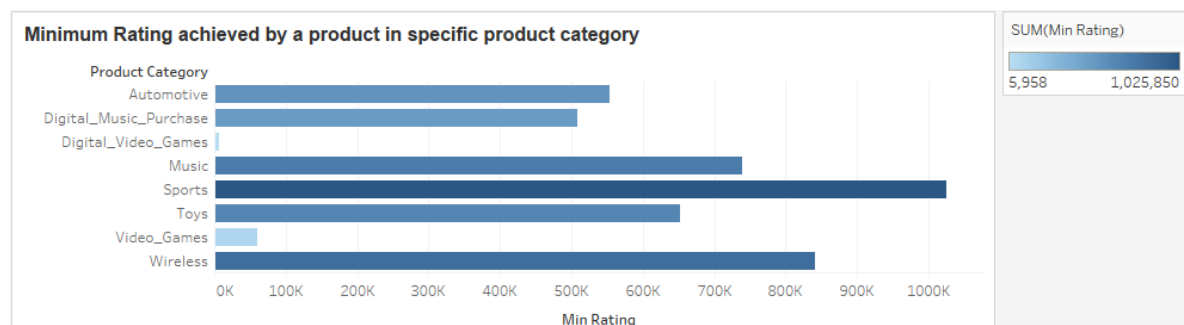
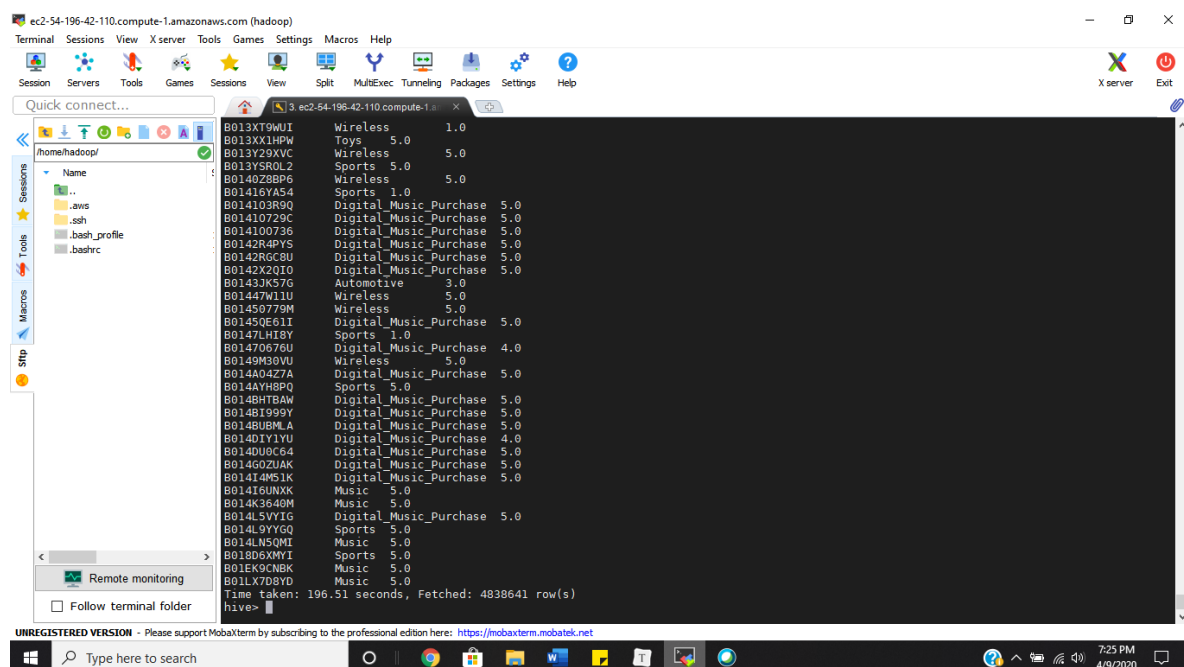
```
1771004
Time taken: 41.2 seconds, Fetched: 1 row(s)
hive> SELECT product_category,
>       round(avg(star_rating),
>             2) AS avg_rating
> FROM amazon_review.amazon_review_filtered_data
> WHERE year >= 2005
> GROUP BY product_category
> HAVING avg(star_rating) in (
>     (SELECT max(x.avg_stars)
>      FROM
>         (SELECT product_category,
>                avg(star_rating) AS avg_stars
>          FROM amazon_review.amazon_review_filtered_data
>           WHERE year >= 2005
>           GROUP BY product_category) AS x));
Query ID = hadoop_20200411054737_c2a5e107-31be-46a7-be62-95e8d473ddd1
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1586576550638_0012)

-----
VERTICES      MODE      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED  28      28          0         0         0         0
Map 3 ..... container  SUCCEEDED  28      28          0         0         0         0
Reducer 2 ... container  SUCCEEDED  20      20          0         0         0         0
Reducer 4 ... container  SUCCEEDED  20      20          0         0         0         0
Reducer 5 ... container  SUCCEEDED  1       1          0         0         0         0
-----
VERTICES: 05/05 [=====] 100% ELAPSED TIME: 83.70 s
-----
OK
Digital_Music_Purchase 4.65
Time taken: 91.409 seconds, Fetched: 1 row(s)
hive>
```

## Minimum Rating of a product in the specific product category:

Calculated minimum rating of a product in a specific product category to find out the least popular product in that category.

```
SELECT x.product_id,  
       x.product_category,  
       round(min(x.avg_rating),  
             2) AS Min_Rating  
FROM  
  (SELECT product_id,  
         product_category,  
         avg(star_rating) AS avg_rating  
   FROM amazon_review.amazon_review_filtered_data  
   WHERE year >= 2005  
   GROUP BY product_id, product_category) AS x  
GROUP BY x.product_id, x.product_category;
```



From the above diagram, we can interpret that the Digital\_Video\_Games product category is the least popular product category as compared to others.

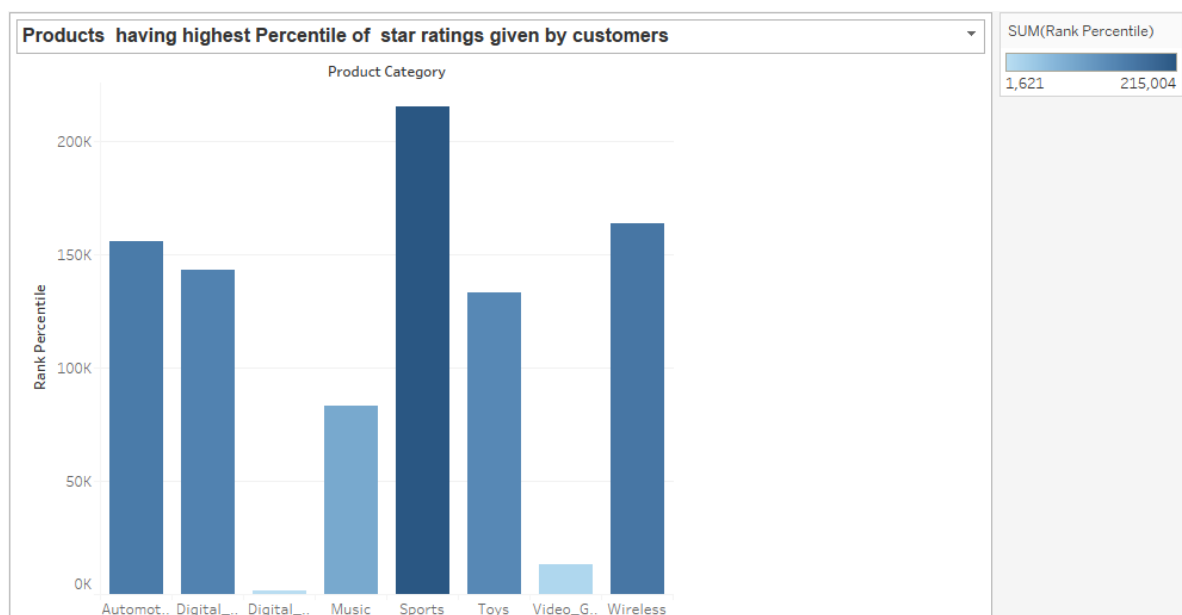
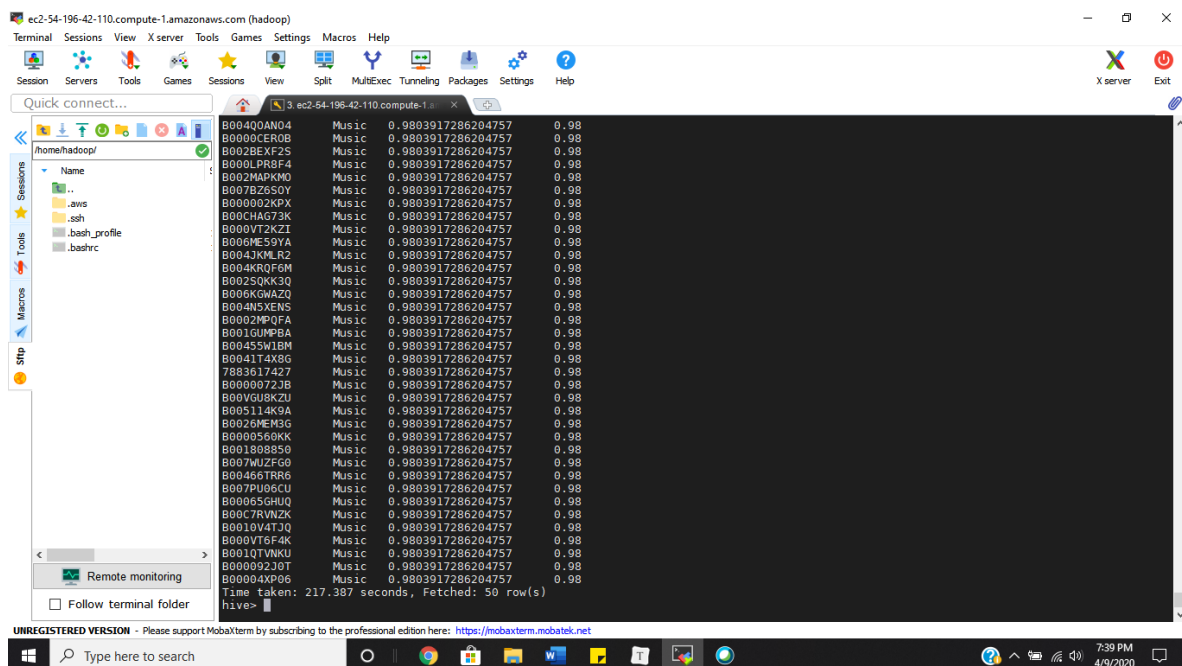
## Products having highest Percentile of star ratings given by customers:

Calculated percentile of product categories to find out product category having the highest percentile.

```

SELECT y.product_id,
       y.product_category, y.star_rank,
       round(y.star_rank,
             2) AS Rank_Percentile from
(SELECT x.product_id,
       x.product_category,
       PERCENT_RANK()
       OVER (partition by x.product_category
ORDER BY x.avg_rating desc) AS star_rank
FROM
       (SELECT product_id,
        product_category,
        avg(star_rating) AS avg_rating
FROM amazon_review.amazon_review_filtered_data
WHERE year>= 2005
GROUP BY product_id,product_category)as x)as y order by y.star_rank
desc;

```



From the above diagram, we can interpret that the sports product category is the most popular product category since it has the highest percentile of rating as compared to others.

## Conclusion:

Analyzed different parameters like Total Number\_of\_Reviews, Number\_of\_Users, Average\_Review\_Stars, Avg\_Length\_of\_Review, Verified\_Users, Unverified\_Users, Total\_Helpful\_Votes, Total\_Products by performing exploratory analysis on cleaned amazon\_reviews database. Realized few product categories are having low average ratings after analyzing these parameters. Found correlation between Music and Digital Music Purchase categories from the detailed analysis of these categories. On the other hand, Video\_Games and Digital\_Video\_Games categories are not correlated. In 2014 the number of customers reviewing both Music/Digital\_Music\_Purchase and Video\_Games/Digital\_Video\_Games categories was more as compared to other years. Hence moved to focus on 2014 to find out the reasons behind this result. After interpreting the results of this analysis I realized there are the same products in both Music and Digital Music Purchase and Video\_Games/Digital\_Video\_Games categories but they don't have the same rating. Also, the calculated rank of different products in a specific product category to find out popular products and least rated products in that category and discovered positive annual growth for those products by calculating and comparing moving average. Also, discovered that the sports category has the highest percentile of products with a maximum star rating and the Digital\_Video\_Games category has the lowest percentile of products with maximum star rating.

## References:

**Amazon reviews dataset:**

<https://registry.opendata.aws/amazon-reviews/>

**Documentation:**

<https://s3.amazonaws.com/amazon-reviews-pds/readme.html>

**Code References:**

<https://www.tutorialspoint.com/index.htm>