CIS5200 Term Project Lab Tutorial

Authors: Jasmine Diep, William Lam, Nina Roberts, Taron Sarksyan

Instructor: Dr. Jongwook Woo

Date: 12/09/2020

Analysis of Movie and TV Reviews: A Study of Amazon Data 2005 - 2015

Objectives

In this hands-on lab, you will learn how to:

- Connect to a node of Oracle BDCE (Big Data Compute)
- Get data manually using wget command
- Use other fundamental shell commands
- Use fundamental Hadoop commands
- Use Hive QL commands to perform the analysis
- Visualize the result in Microsoft Power BI

Platform Spec

- Cluster Version: 20.3.3-20
- CPU Speed: MHz: 2195.287
- Cluster # of Nodes: 3
- # of CPUs Cores: 12
- Memory Size: 180GB
- Storage Size: 957GB

Step 1: Get and merge data

SSH to the Hadoop server using the ip address given by the instructor.

NOTE: You have to use your directory name.

```
$ ssh <your directory name>@129.XX.XX.XX
```

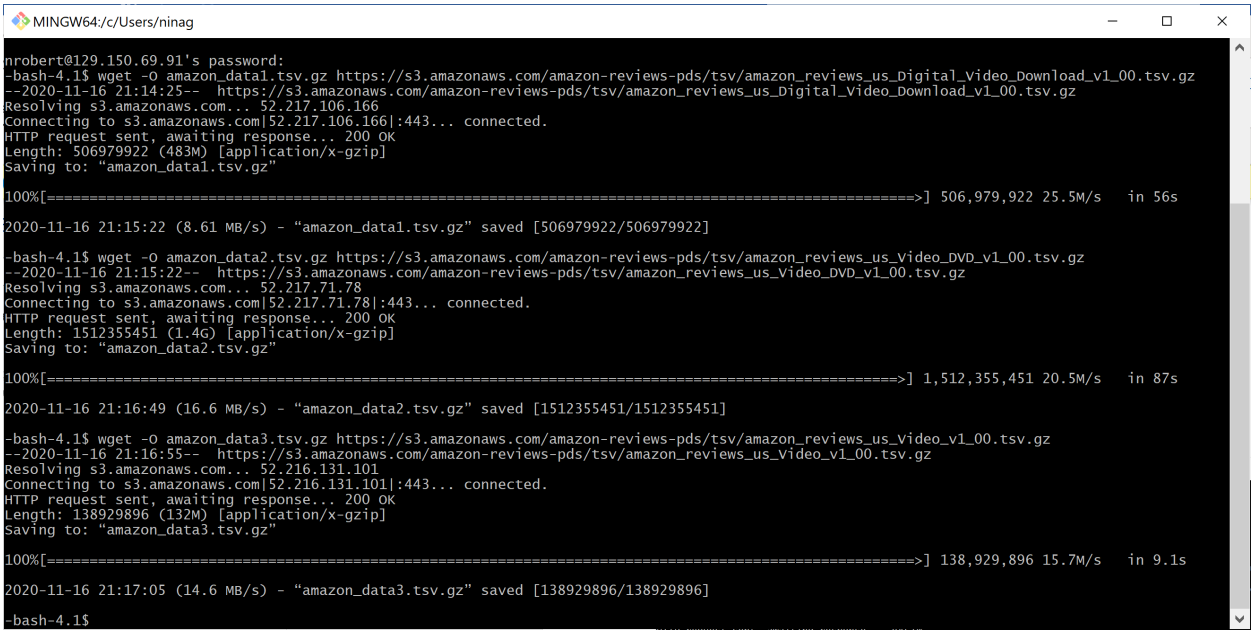
This step will download the Amazon data files from AWS.

Get work files from AWS:

The dataset consists of three files of reviews – DVD reviews, digital download reviews, and VHS reviews. These files contain reviews for all Amazon entertainment such as movies, TV, and instructional videos.

```
$ wget -O amazon_data1.tsv.gz https://s3.amazonaws.com/amazon-reviews-
pds/tsv/amazon_reviews_us_Digital_Video_Download_v1_00.tsv.gz
$ wget -O amazon_data2.tsv.gz https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon_reviews_us_Video_DVD_v1_00.tsv.gz
$ wget -O amazon_data3.tsv.gz https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon_reviews_us_Video_v1_00.tsv.gz

$ ls -al   --check for 3 files
```



Create work directory:

```
$ hdfs dfs -mkdir amazon_data

$ hdfs dfs -ls
```

```
100%[=======================================================================================>] 1,512,355,451 27.5M/s   in 42s

2020-11-09 16:49:07 (34.5 MB/s) - “amazon_data2.tsv.gz” saved [1512355451/1512355451]

-bash-4.1$ hdfs dfs -mkdir amazon_data
-bash-4.1$ hdfs dfs -ls
Found 1 items
drwxr-xr-x   - nrobert hdfs          0 2020-11-09 16:49 amazon_data
-bash-4.1$
```

Unzip and save .tsv files to HDFS:

```
$ gzip -d amazon_data1.tsv.gz
$ gzip -d amazon_data2.tsv.gz
$ gzip -d amazon_data3.tsv.gz
```

List local files:

We see 3 files:

-1.2GB amazon_data1.tsv

-3.7GB amazon_data2.tsv

-337MB amazon_data3.tsv

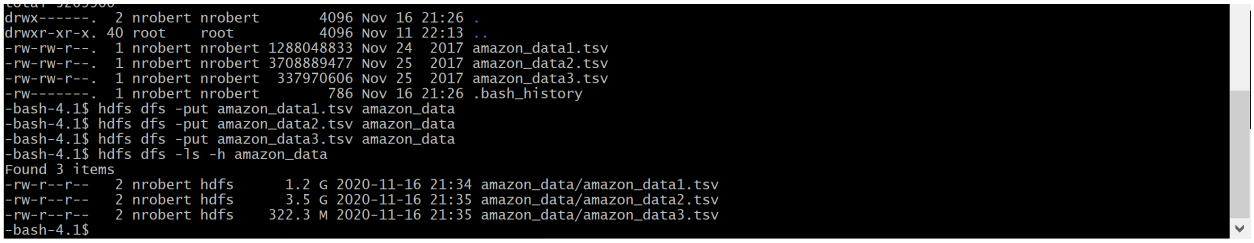
```
$ ls -al
```

```
-bash-4.1$
-bash-4.1$
-bash-4.1$ ls
amazon_data1.tsv  amazon_data2.tsv  amazon_data3.tsv
-bash-4.1$ ls -al
total 5209900
drwx------.  2 nrobert nrobert       4096 Nov 16 21:26 .
drwxr-xr-x. 40 root    root          4096 Nov 11 22:13 ..
-rw-rw-r--.  1 nrobert nrobert 1288048833 Nov 24  2017 amazon_data1.tsv
-rw-rw-r--.  1 nrobert nrobert 3708889477 Nov 25  2017 amazon_data2.tsv
-rw-rw-r--.  1 nrobert nrobert  337970606 Nov 25  2017 amazon_data3.tsv
-rw-------.  1 nrobert nrobert        786 Nov 16 21:26 .bash_history
-bash-4.1$
```

Save 3 data files into HDFS:

```
$ hdfs dfs -put amazon_data1.tsv amazon_data
$ hdfs dfs -put amazon_data2.tsv amazon_data
$ hdfs dfs -put amazon_data3.tsv amazon_data
$ hdfs dfs -ls -h amazon_data
```

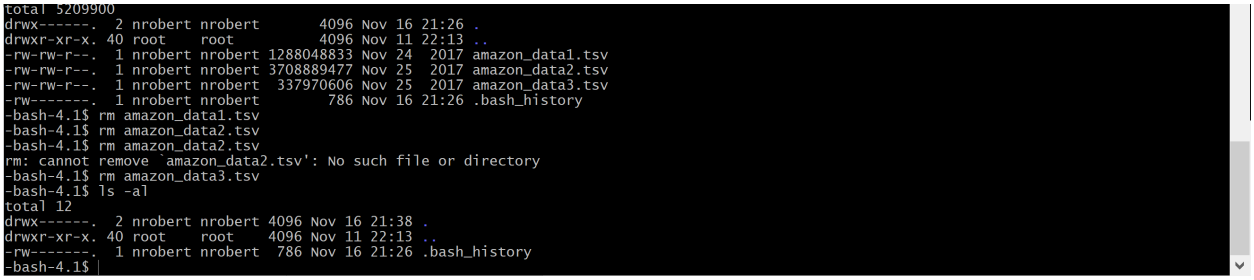
Review data in amazon_data directory on HDFS:



Remove files on local drive and list files so proof files were deleted:

```
$ rm amazon_data1.tsv
$ rm amazon_data2.tsv
$ rm amazon_data3.tsv
```

```
$ ls -al
```

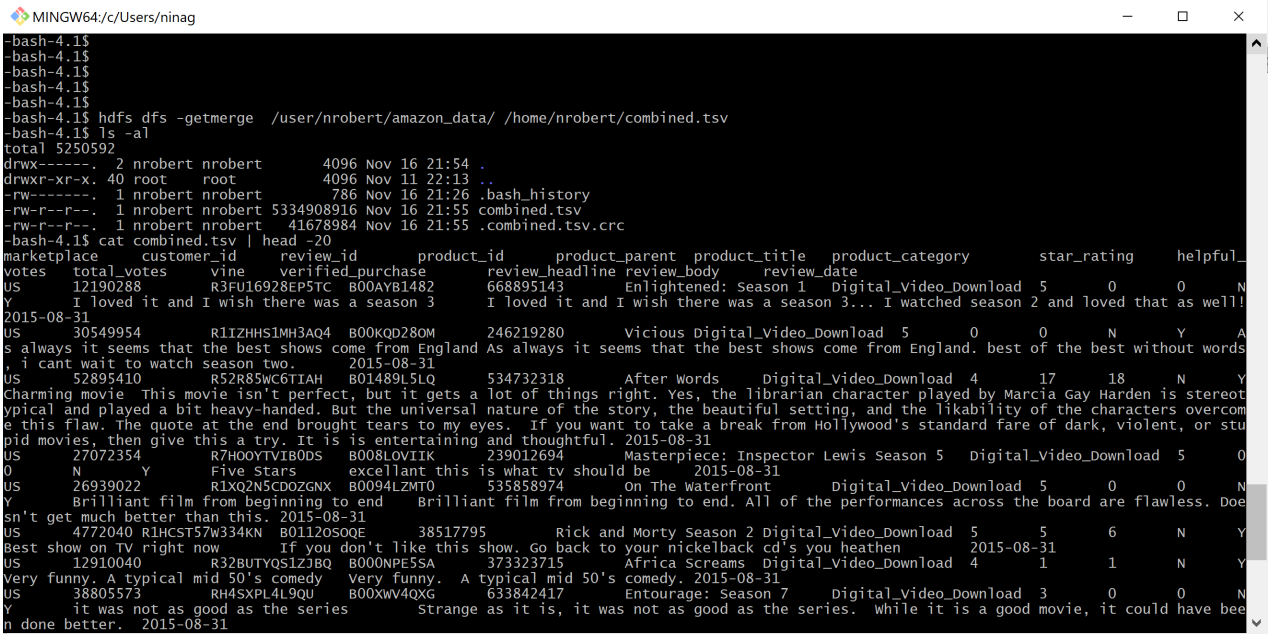


Merge 3 Amazon data files together, view size and view file contents:

-Combined file is **5.33GB** – combined.tsv

NOTE: You have to use your directory name.

```
$ hdfs dfs -getmerge  /user/<your directory name>/amazon_data/
/home/<your directory name>/combined.tsv
$ ls -al
$ hdfs dfs -put combined.tsv /user/<your directory
name>/amazon_data
$ hdfs dfs -cat combined.tsv | head -20
```

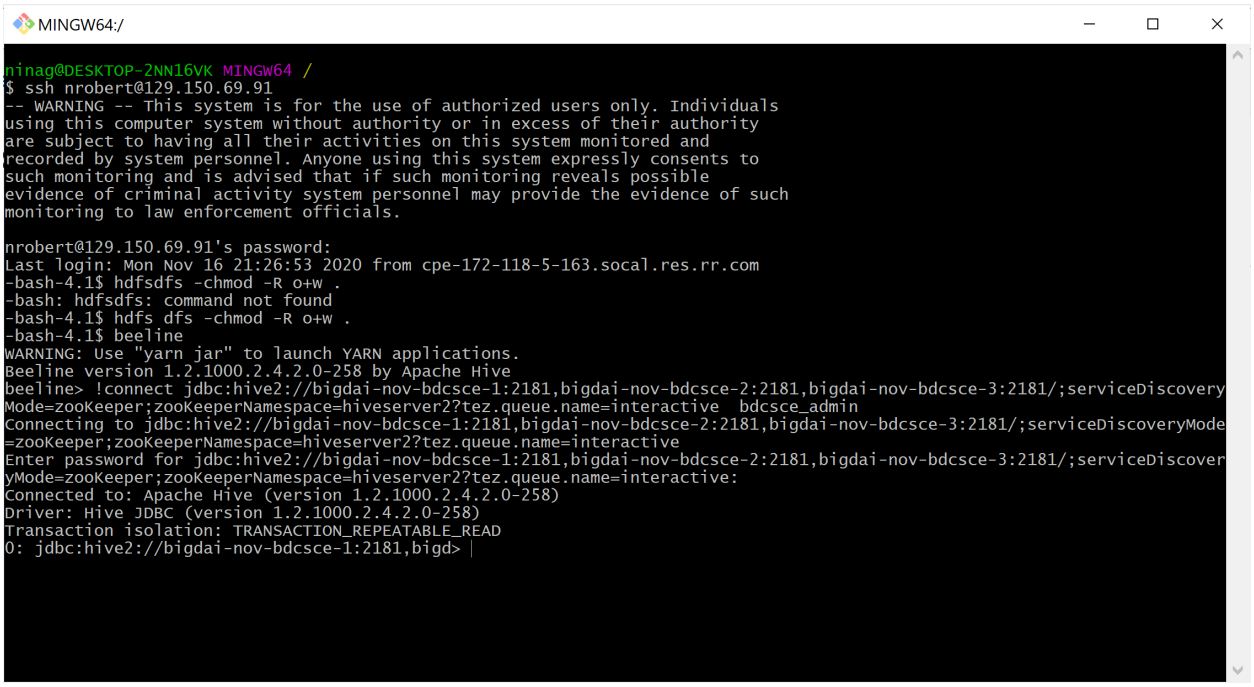

MINGW64:/c/Users/ninag
-bash-4.1$
-bash-4.1$
-bash-4.1$
-bash-4.1$
-bash-4.1$
-bash-4.1$ hdfs dfs -getmerge /user/nrobert/amazon_data/ /home/nrobert/combined.tsv
-bash-4.1$ ls -al
total 5250592
drwx------. 2 nrobert nrobert 4096 Nov 16 21:54 .
drwxr-xr-x. 40 root root 4096 Nov 11 22:13 ..
-rw-------. 1 nrobert nrobert 786 Nov 16 21:26 .bash_history
-rw-r--r--. 1 nrobert nrobert 5334908916 Nov 16 21:55 combined.tsv
-rw-r--r--. 1 nrobert nrobert 41678984 Nov 16 21:55 .combined.tsv.crc
-bash-4.1$ cat combined.tsv | head -20
marketplace customer_id review_id product_id product_parent product_title product_category star_rating helpful_
votes total_votes vine verified_purchase review_headline review_body review_date
US 12190288 R3FU16928EP5TC B00AYB1482 668895143 Enlightened: Season 1 Digital_Video_Download 5 0 0 N
Y I loved it and I wish there was a season 3 I loved it and I wish there was a season 3... I watched season 2 and loved that as well!
2015-08-31
US 30549954 R1IZHHS1MH3AQ4 B00KQD28OM 246219280 Vicious Digital_Video_Download 5 0 0 N Y A
s always it seems that the best shows come from England As always it seems that the best shows come from England. best of the best without words
, i cant wait to watch season two. 2015-08-31
US 52895410 R52R85WC6TIAH B01489L5LQ 534732318 After Words Digital_Video_Download 4 17 18 N Y
Charming movie This movie isn't perfect, but it gets a lot of things right. Yes, the librarian character played by Marcia Gay Harden is stereot
ypical and played a bit heavy-handed. But the universal nature of the story, the beautiful setting, and the likability of the characters overcom
e this flaw. The quote at the end brought tears to my eyes. If you want to take a break from Hollywood's standard fare of dark, violent, or stu
pid movies, then give this a try. It is is entertaining and thoughtful. 2015-08-31
US 27072354 R7HOOYTVIB0DS B008LOVIIK 239012694 Masterpiece: Inspector Lewis Season 5 Digital_Video_Download 5 0
0 N Y Five Stars excellant this is what tv should be 2015-08-31
US 26939022 R1XQ2N5CDOZGNX B0094LZMT0 535858974 On The Waterfront Digital_Video_Download 5 0 0 N
Y Brilliant film from beginning to end Brilliant film from beginning to end. All of the performances across the board are flawless. Doe
sn't get much better than this. 2015-08-31
US 4772040 R1HCST57W334KN B0112OSOQE 38517795 Rick and Morty Season 2 Digital_Video_Download 5 5 6 N Y
Best show on TV right now If you don't like this show. Go back to your nickelback cd's you heathen 2015-08-31
US 12910040 R32BUTYQS1ZJBQ B000NPE5SA 373323715 Africa Screams Digital_Video_Download 4 1 1 N Y
Very funny. A typical mid 50's comedy Very funny. A typical mid 50's comedy. 2015-08-31
US 38805573 RH4SXPL4L9QU B00XWV4QXG 633842417 Entourage: Season 7 Digital_Video_Download 3 0 0 N
Y it was not as good as the series Strange as it is, it was not as good as the series. While it is a good movie, it could have bee
n done better. 2015-08-31

Step 2: Data Engineering and Analysis - Hadoop Hive

This step uses Hadoop Hive to create a table, load data into the table, view table contents and query data from the table.

New terminal – connect to HIVE:

```
$ hdfs dfs -chmod -R o+w .
$ beeline
$ get connection info from professor
$ use <your database name>;
```



Create table to store data from "combined.tsv":

```
CREATE EXTERNAL TABLE if not exists amazon(
marketplace string,
customer_id string,
review_id string,
product_id string,
product_parent string,
product_title string,
product_category string,
star_rating int,
helpful_votes int,
total_votes int,
vine string,
verified_purchase string,
review_headline string,
review_body string,
review_date date)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
STORED AS TEXTFILE LOCATION '/user/<your database name>/amazon_data/combined.tsv/'
TBLPROPERTIES ('skip.header.line.count'='1');
```

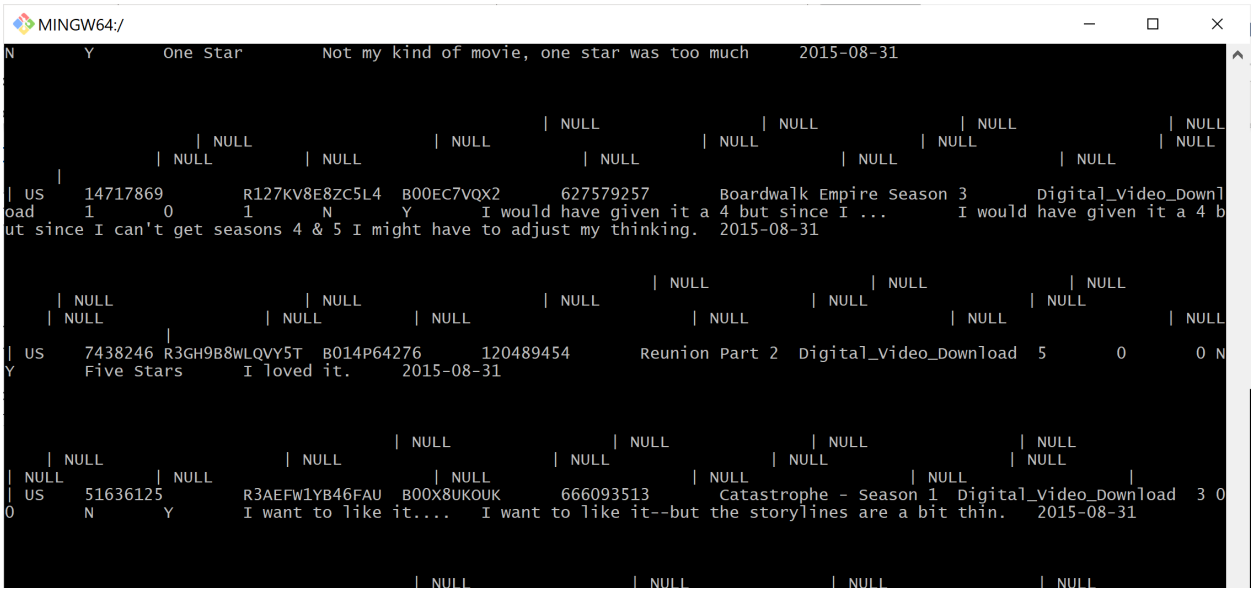
--Review table headers:

```
DESCRIBE amazon;
```

```
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd> review_headline string,
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd> review_body string,
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd> review_date date
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd> );
No rows affected (0.226 seconds)
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd> show tables;
+-----------+--+
| tab_name  |
+-----------+--+
| amazon    |
+-----------+--+
1 row selected (0.189 seconds)
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd> describe amazon;
+--------------------+------------+----------+--+
|      col_name      | data_type  | comment  |
+--------------------+------------+----------+--+
| marketplace        | string     |          |
| customer_id        | string     |          |
| review_id          | string     |          |
| product_id         | string     |          |
| product_parent     | string     |          |
| product_title      | string     |          |
| product_category   | string     |          |
| star_rating        | int        |          |
| helpful_votes      | int        |          |
| total_votes        | int        |          |
| vine               | string     |          |
| verified_purchase  | string     |          |
| review_headline    | string     |          |
| review_body        | string     |          |
| review_date        | date       |          |
+--------------------+------------+----------+--+
15 rows selected (0.213 seconds)
0: jdbc:hive2://bigdai-nov-bdcsce-1:2181,bigd>
```

Viewing 20 rows of data:

```
SELECT * from amazon LIMIT 20;
```



We can query the number of reviews in the data:

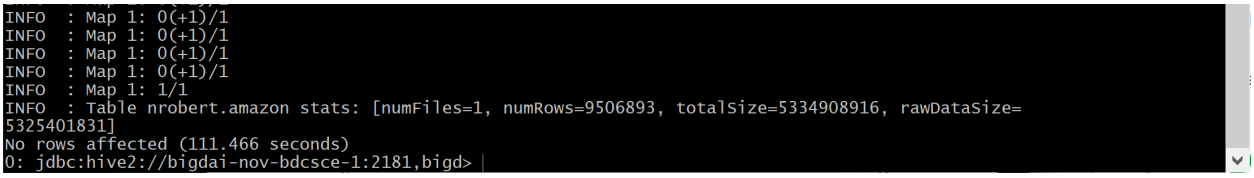
```
SELECT COUNT(*) review_id FROM amazon;
```

NOTE: You have to use your directory name.

Use "ANALYZE" function to view statistics on table "amazon":

```
ANALYZE TABLE amazon COMPUTE STATISTICS;
```

[Table **<your database name>**.amazon stats: [numFiles=1, numRows=9506893, totalSize=5334908916, rawDataSize=5325401831]

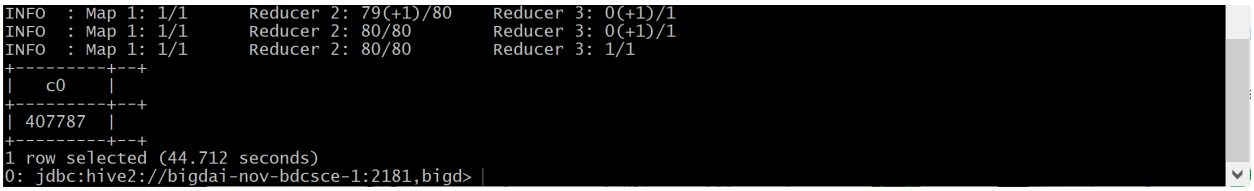


How many movie/TV titles are in the data?

-Count distinct titles

```
SELECT COUNT(DISTINCT (product_title)) FROM amazon;
```

-407,787 disctinct titles exist



--Review years and counts:

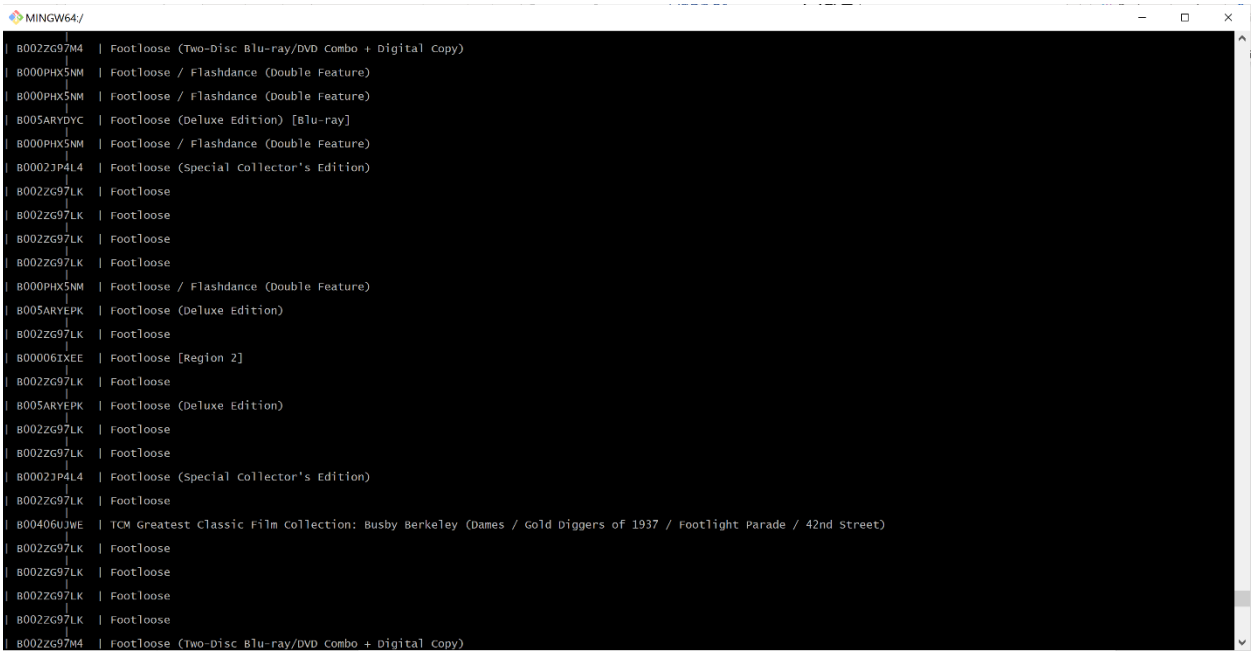
```
SELECT DATE_FORMAT(REVIEW_DATE, 'yyyy') AS YEAR,
    COUNT (*) AS YEAR_COUNT
    FROM AMAZON
    GROUP BY DATE_FORMAT(REVIEW_DATE, 'yyyy') AS YEAR,
    ORDER BY YEAR;
```

-Does a title have one Product_ID or many? How should we group products together to determine ratings?

-In the case of the movie “Footloose” there are many product_ids associated with this title and we see that repeating with other product_title(s) in the data.

Example 1 – Title = “Footloose”

```
SELECT * FROM AMAZON WHERE PRODUCT_TITLE = 'Footloose'
order by review_date desc;
```

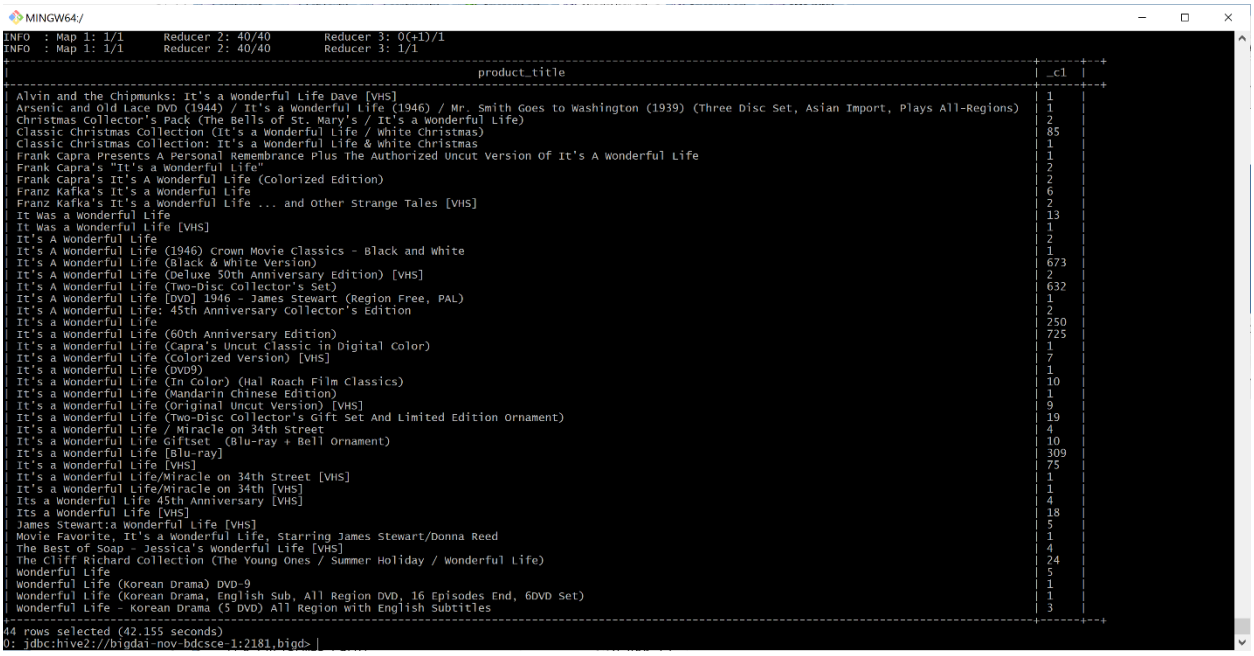


Are movie/TV titles distinct? Or are there various spellings for each title?

Likewise, there are various product_title for this one movie. This means we will need to normalize or standardize the data before we can group titles correctly.

Example 2 – Title = “It’s A Wonderful Life”

```
SELECT product_title, COUNT(product_title)
FROM    AMAZON
WHERE
    product_title LIKE '%It’s A Wonderful Life%'
GROUP BY
    product_title
ORDER BY
    product_title;
```



We can see that the product_title field in these two examples has these types of notations which will prevent us from grouping successfully on product_title:

-[Blu-Ray]
-[VHS]
-[Theatrical Release]
-[DVD + Digital Copy + UltraViolet]
-[DVD + Digital]
-[Region 2]
-[Ultra HD]
-[PAL]
-[Blu-ray + Digital Copy]
-(Blu-ray/DVD Combo)

Assumptions:
-For the most part, TV listings do not seem problematic for popular titles.
-For movies between 1990 and 2011, Amazon saved the movie year in the product_title field.
-We discovered that if you paste the product_id into amazon.com it will link you to the title.

NOTE: The following will prevent accurate grouping.
-product_title contains both “A” and “a”
-product_title contains both “The” and “the”
-product_title contains both “Its” and “It’s”

To delete text inside either brackets or parenthesis in product_title:

```
SELECT product_title =
regexp_replace(product_title,'\s*\[[^()]*\]', '')
FROM amazon;

SELECT product_title =
regexp_replace(product_title,'\s*\([^()]*\)', '')
FROM amazon;
```

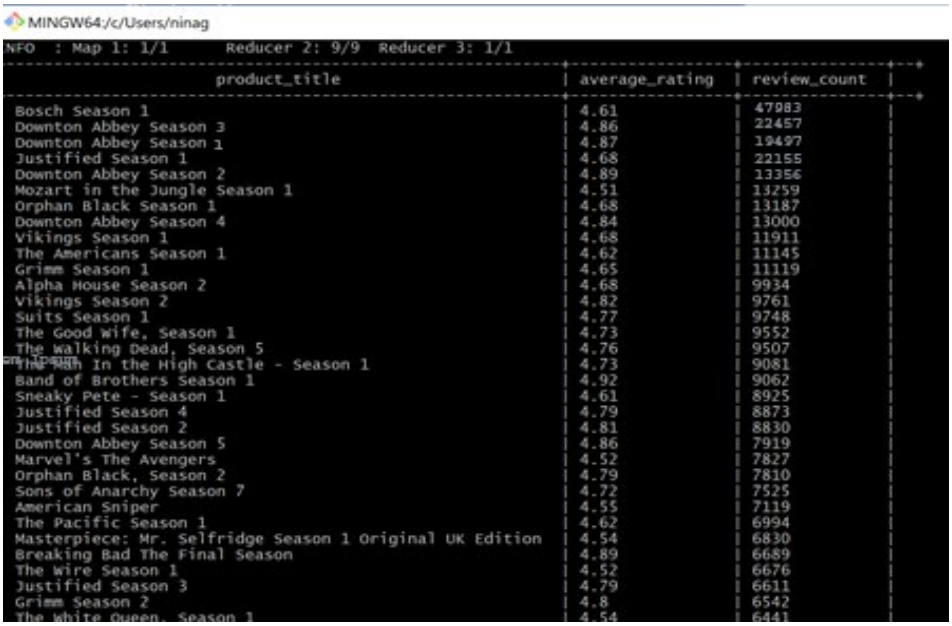
Pitfalls of this method:
There will be some titles that will erroneously be grouped together. These examples were identified and manually fixed:

- Little Women
- Midway

Next query top movie titles:

Run query to select top 100 movie reviews by average star rating and count of ratings.

```
SELECT
product_title,
ROUND(avg(star_rating), 2) as average_rating,
COUNT(review_id) as review_count
FROM amazon
WHERE (review_date >= '01-Jan-05' AND review_date <= '31-Dec-
15') and (product_title != 'Pilot')
GROUP by product_title HAVING avg(star_rating) >= 4.5
ORDER by review_count DESC
LIMIT 100;
```



product_title	average_rating	review_count
Bosch Season 1	4.61	47083
Downton Abbey Season 3	4.86	22457
Downton Abbey Season 1	4.87	19497
Justified Season 1	4.68	22155
Downton Abbey Season 2	4.89	13356
Mozart in the Jungle Season 1	4.51	13259
Orphan Black Season 1	4.68	13187
Downton Abbey Season 4	4.84	13000
Vikings Season 1	4.68	11911
The Americans Season 1	4.62	11145
Grimm Season 1	4.65	11119
Alpha House Season 2	4.68	9934
Vikings Season 2	4.82	9761
Suits Season 1	4.77	9748
The Good Wife, Season 1	4.73	9552
The Walking Dead, Season 5	4.76	9507
[illegible] In the High Castle - Season 1	4.73	9081
Band of Brothers Season 1	4.92	9062
Sneaky Pete - Season 1	4.61	8925
Justified Season 4	4.79	8873
Justified Season 2	4.81	8830
Downton Abbey Season 5	4.86	7919
Marvel's The Avengers	4.52	7827
Orphan Black, Season 2	4.79	7810
Sons of Anarchy Season 7	4.72	7525
American Sniper	4.55	7119
The Pacific Season 1	4.62	6994
Masterpiece: Mr. Selfridge Season 1 Original UK Edition	4.54	6830
Breaking Bad The Final Season	4.89	6689
The Wire Season 1	4.52	6676
Justified Season 3	4.79	6611
Grimm Season 2	4.8	6542
The White Queen, Season 1	4.54	6441

Working in Hadoop Hive you can create a table with your data:

Create tmp directory in HDFS:

```
$ hdfs dfs -mkdir tmp;
```

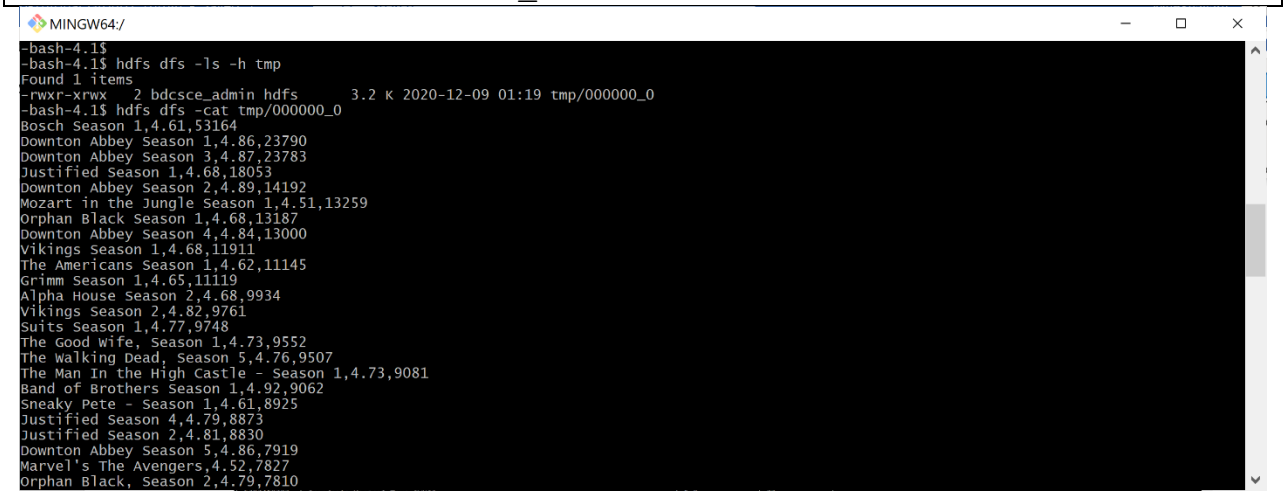
Create Top 100 table and save as comma delimited file in HDFS tmp directory:

NOTE: You have to use your directory name.

```
CREATE TABLE IF NOT EXISTS amazon_top
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ","
STORED AS TEXTFILE LOCATION "/user/<your directory name>/tmp"
AS
SELECT
product_title,
ROUND(avg(star_rating), 2) as average_rating,
COUNT(review_id) as review_count
FROM amazon
WHERE (review_date >= '01-Jan-05' AND review_date <= '31-Dec-
15') and (product_title != 'Pilot')
GROUP by product_title HAVING avg(star_rating) >= 4.5
ORDER by review_count DESC
LIMIT 100;
```

View Top 100 file:

```
$ hdfs dfs -cat tmp/000000_0
```



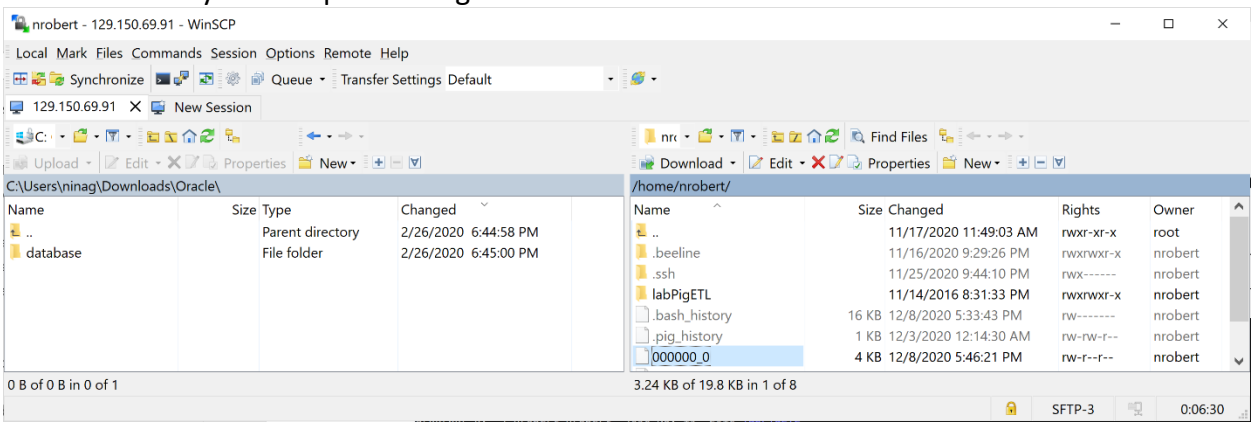
Export File to your local drive and then computer using WinSCP, putty or other method.

Move file to local drive

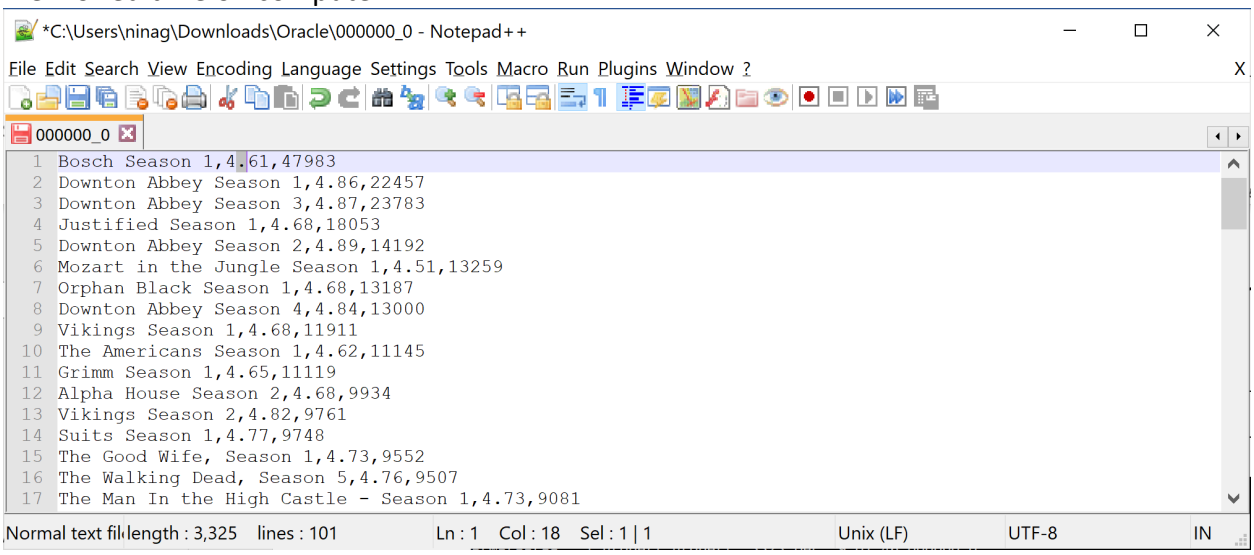
```
hdfs dfs -get tmp/000000_0
```

```
-bash-4.1$ ls -al
total 56
drwx------.  5 nrobert nrobert  4096 Dec  9 01:46 .
drwxr-xr-x. 42 root    root     4096 Nov 17 19:49 ..
-rw-r--r--.  1 nrobert nrobert  3325 Dec  9 01:46 000000_0
-rw-------.  1 nrobert nrobert 15511 Dec  9 01:33 .bash_history
drwxrwxr-x.  2 nrobert nrobert  4096 Nov 17 05:29 .beeline
-rw-r--r--.  1 nrobert nrobert  1119 Dec  3 06:24 high_cost_sites2
drwxrwxr-x.  4 nrobert nrobert  4096 Nov 15  2016 labPigETL
-rw-rw-r--.  1 nrobert nrobert  1288 Dec  3 07:33 pig_1606980811776.log
```

Transfer file to your computer using WinSCP



View or edit file on computer

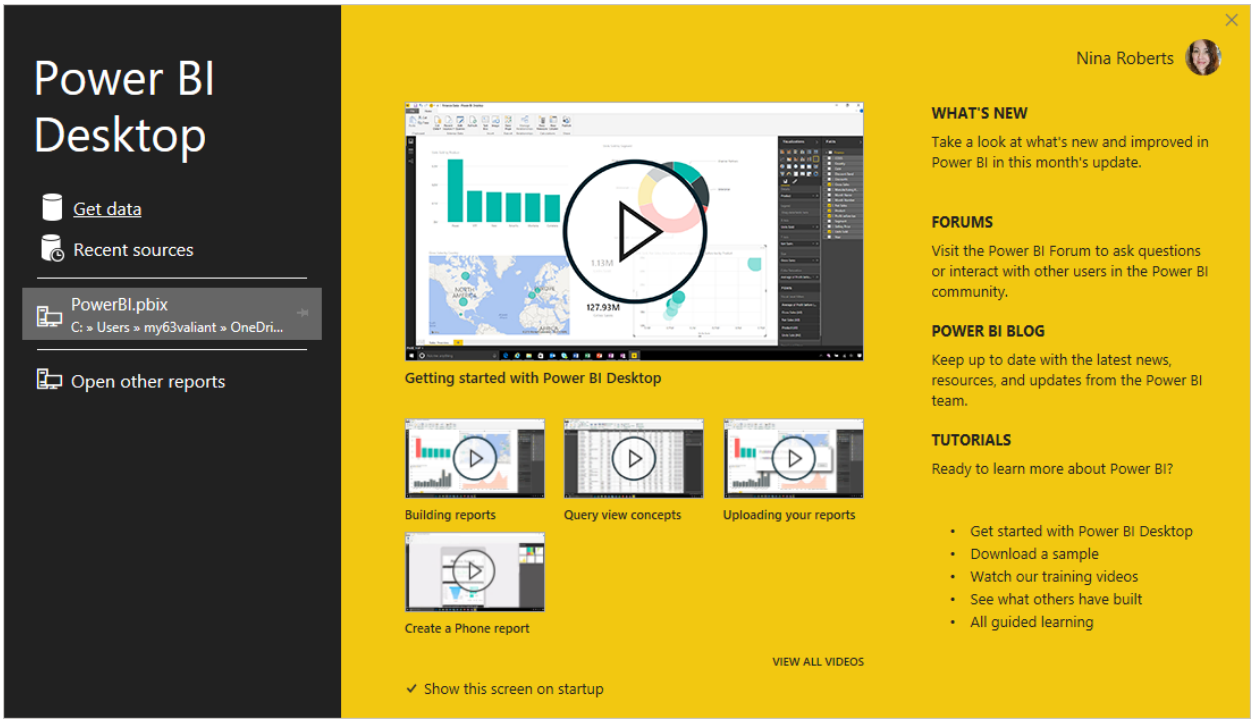


Step 3 : Visualization

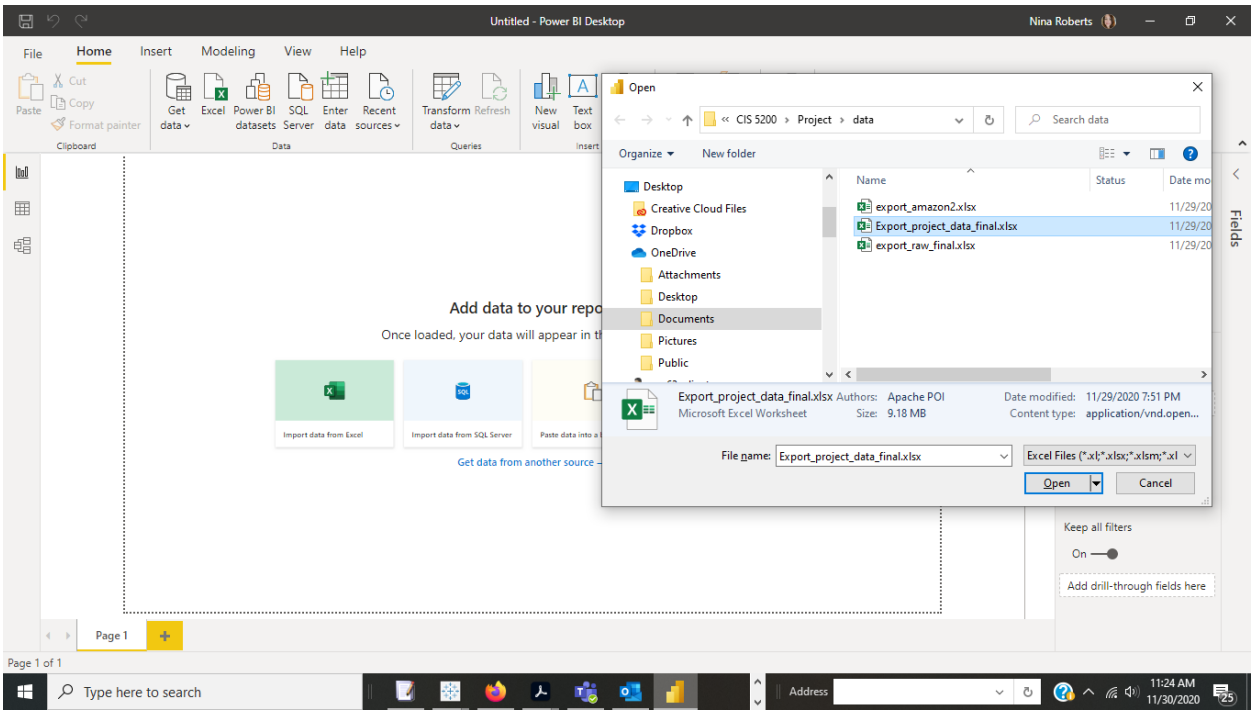
Visualization #1 – Power BI

Download and install Power BI Desktop
https://www.microsoft.com/en-us/download/details.aspx?id=58494

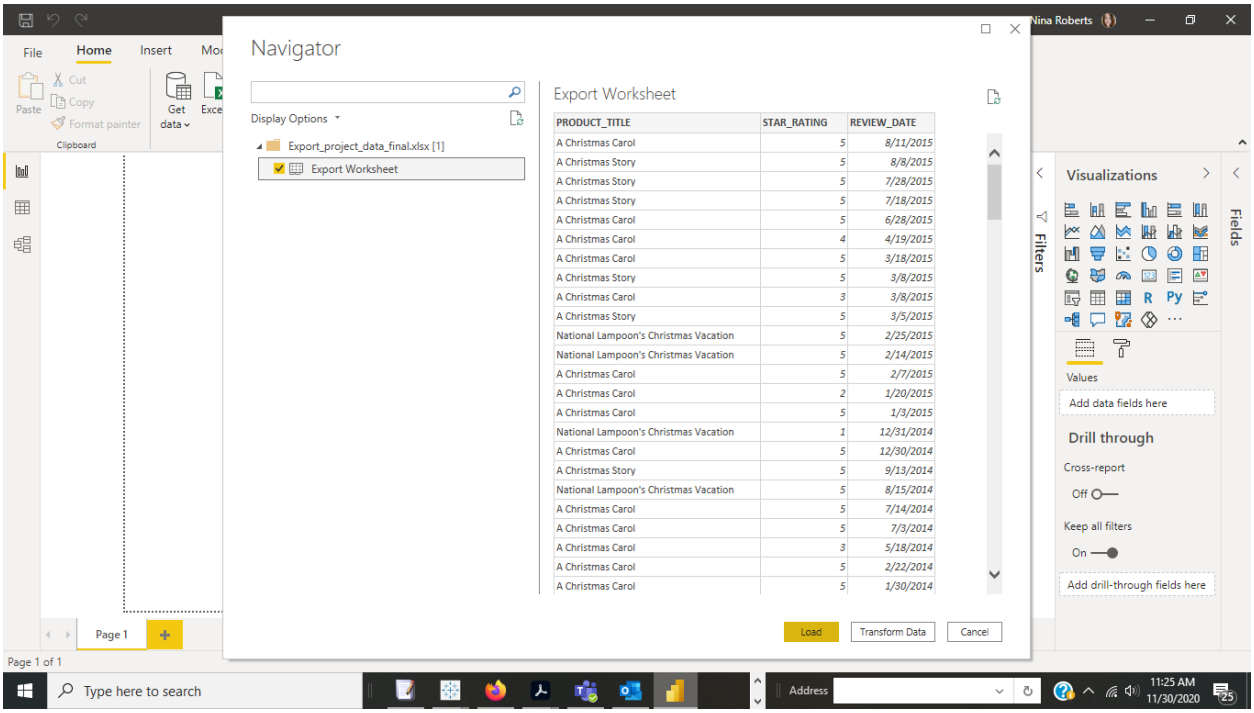
-Click “Get data” in left hand menu



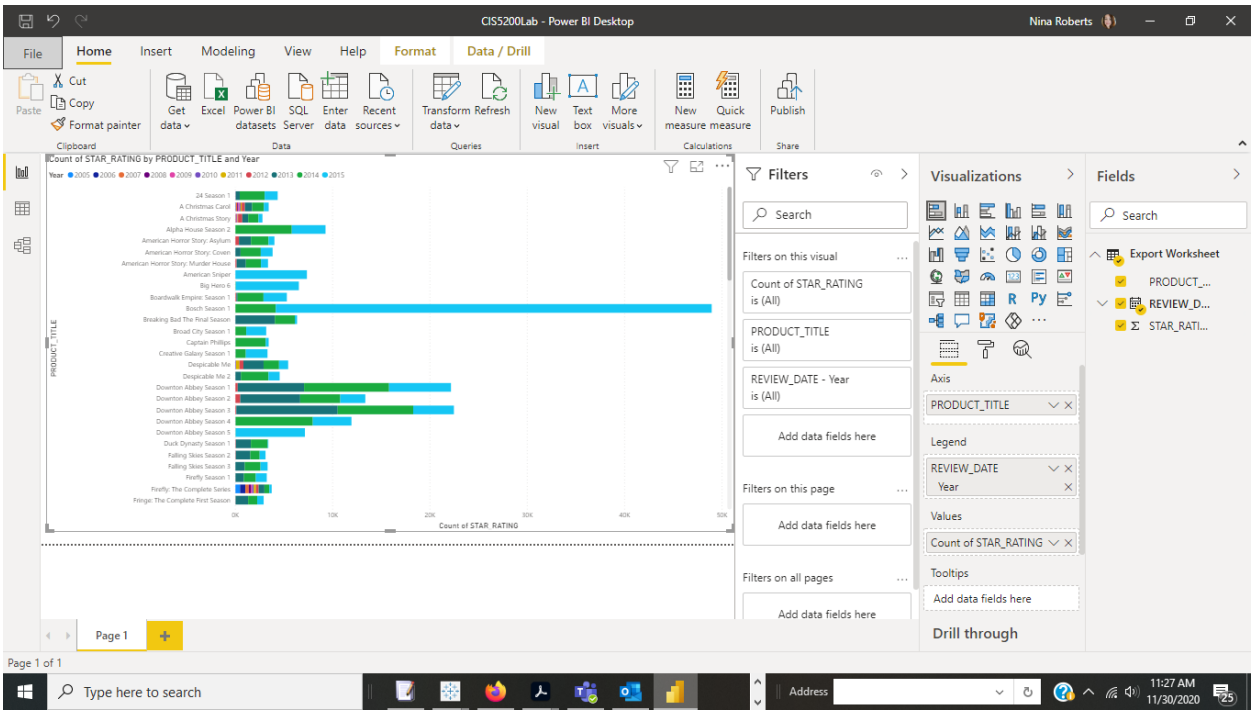
-Click “import data from Excel” and browse to a data file
-Click “Open”



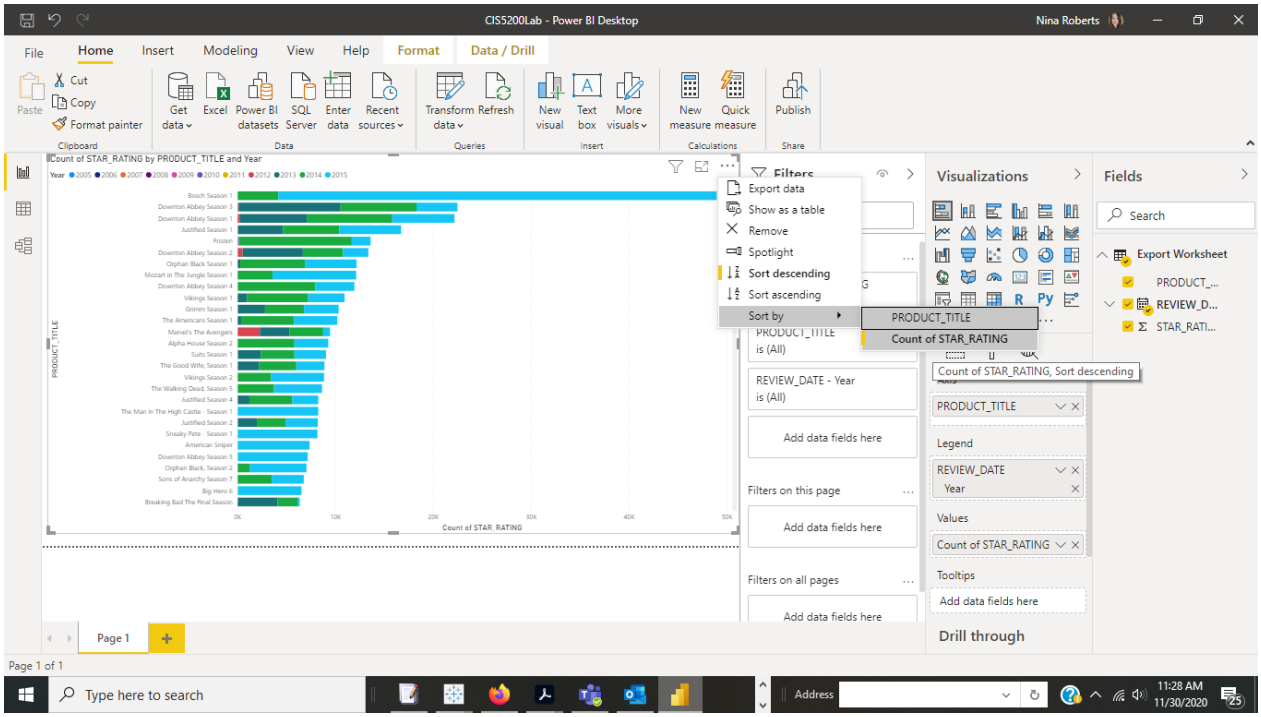
-Click sheet in Excel workbook that contains the data
-Click “Load”



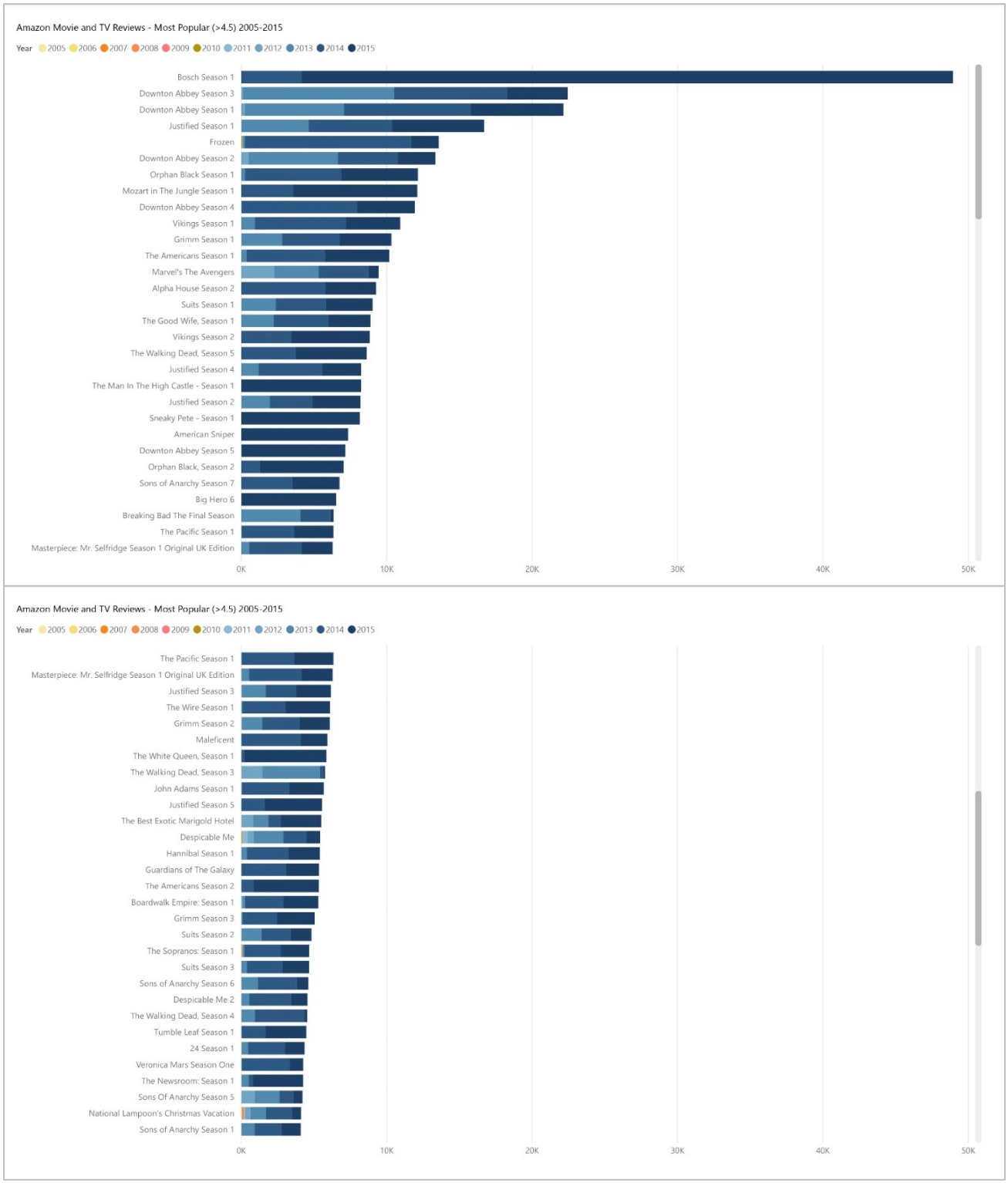
-Select “stacked bar chart” option in “Visualizations” menu
-Assign PRODUCT_TITLE, REVIEW_DATE(YEAR), AND COUNT OF STAR_RATING as graph values

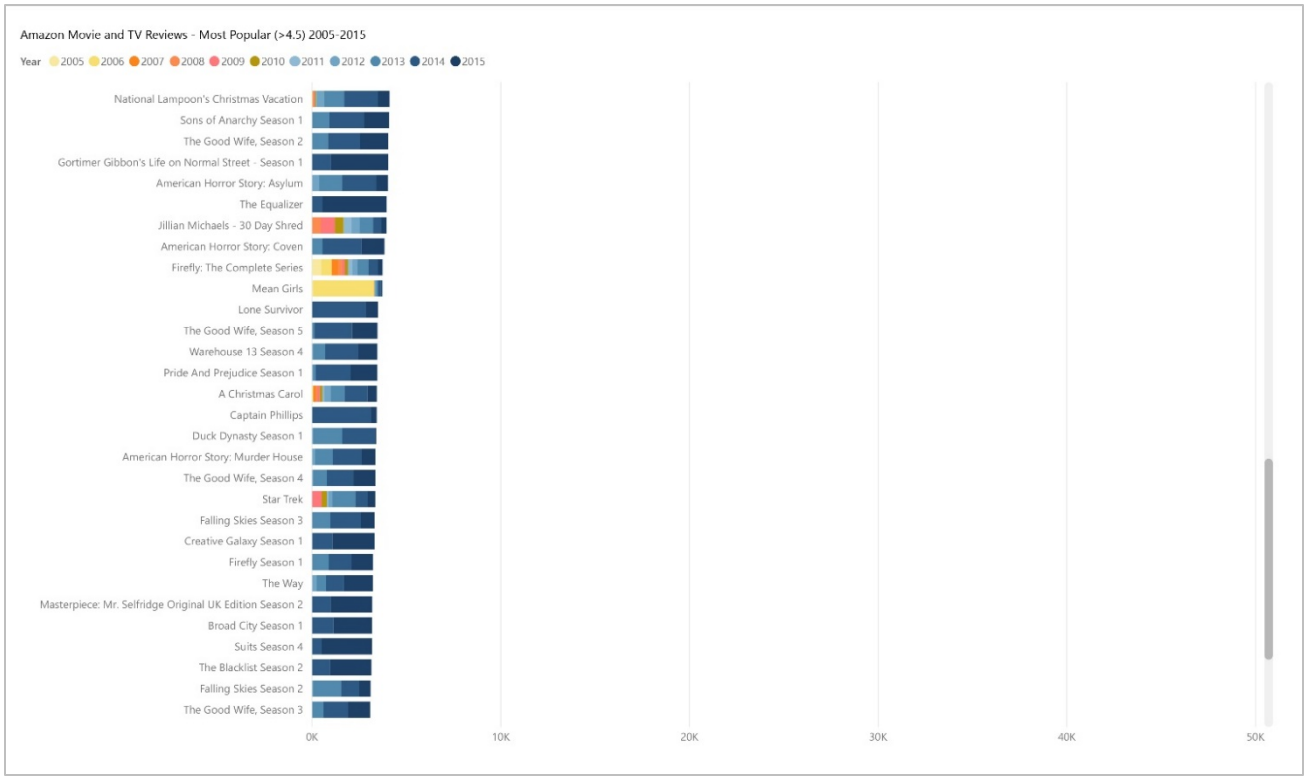


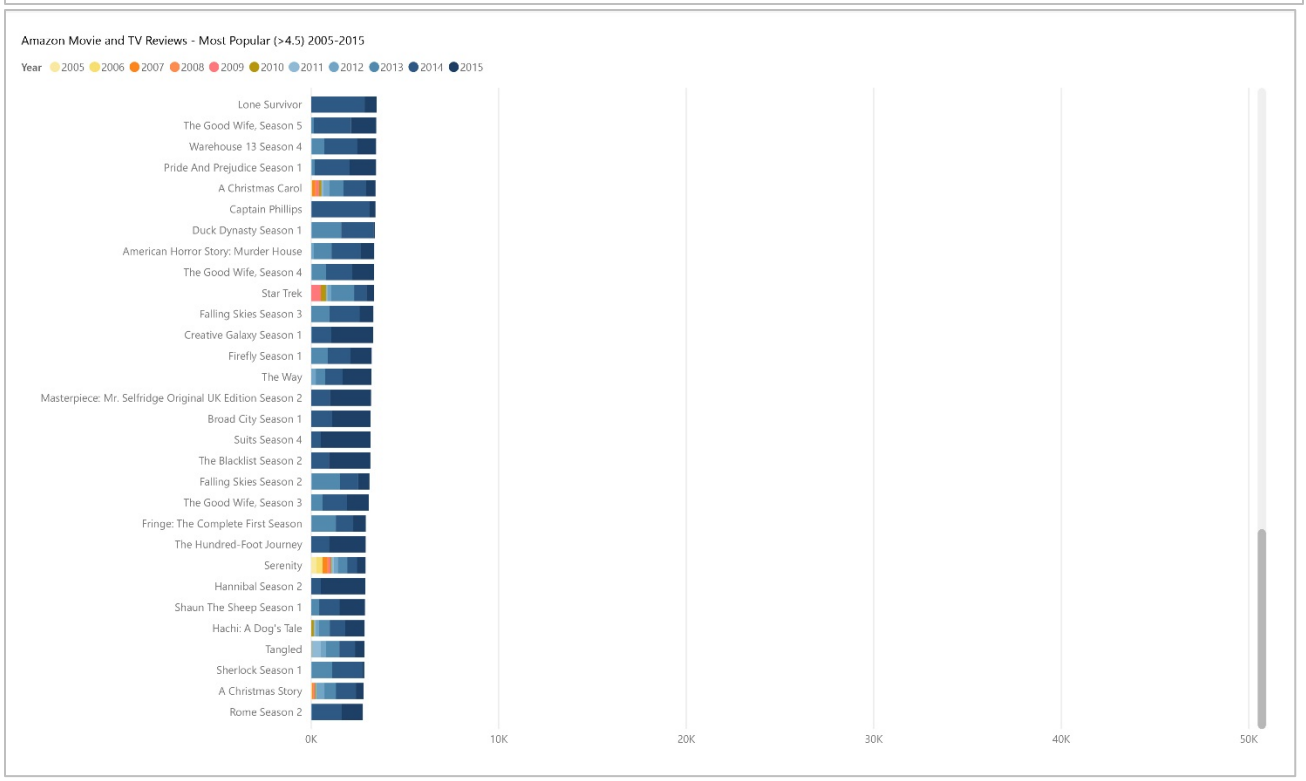
-Select menu under three dots in upper right hand corner to sort by “Count of STAR_RATING”



-Format colors, title and legend and export or print



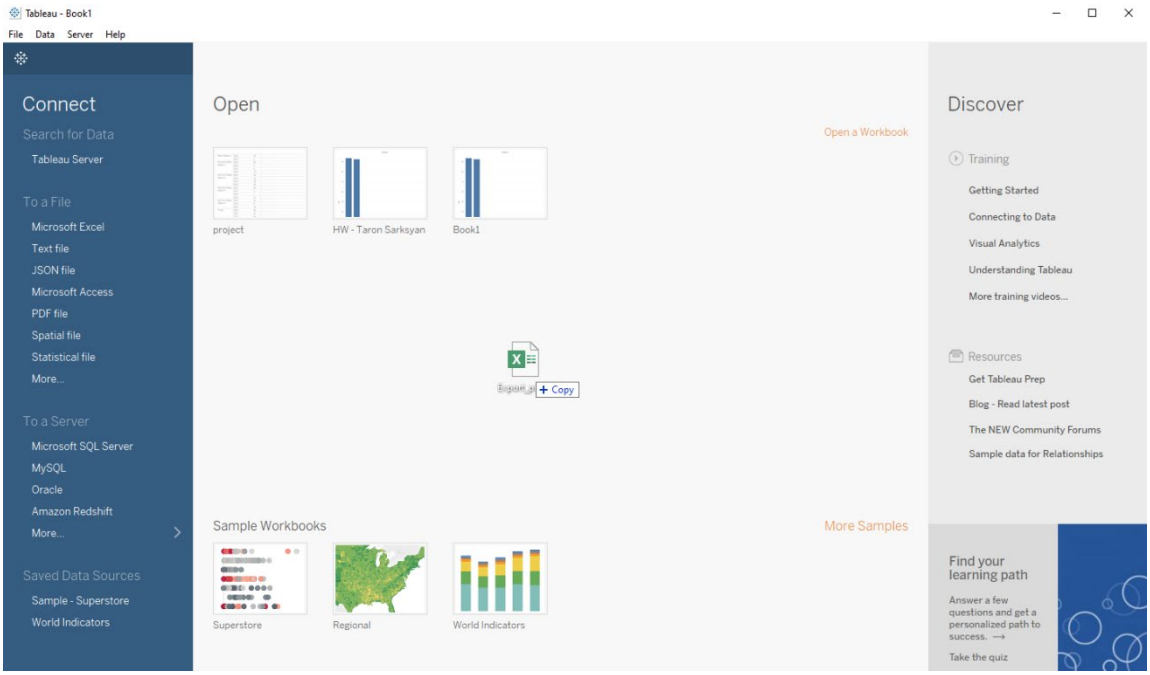

Amazon Movie and TV Reviews - Most Popular (>4.5) 2005-2015
Year 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
National Lampoon's Christmas Vacation
Sons of Anarchy Season 1
The Good Wife, Season 2
Gortimer Gibbon's Life on Normal Street - Season 1
American Horror Story: Asylum
The Equalizer
Jillian Michaels - 30 Day Shred
American Horror Story: Coven
Firefly: The Complete Series
Mean Girls
Lone Survivor
The Good Wife, Season 5
Warehouse 13 Season 4
Pride And Prejudice Season 1
A Christmas Carol
Captain Phillips
Duck Dynasty Season 1
American Horror Story: Murder House
The Good Wife, Season 4
Star Trek
Falling Skies Season 3
Creative Galaxy Season 1
Firefly Season 1
The Way
Masterpiece: Mr. Selfridge Original UK Edition Season 2
Broad City Season 1
Suits Season 4
The Blacklist Season 2
Falling Skies Season 2
The Good Wife, Season 3
0K
10K
20K
30K
40K
50K


Amazon Movie and TV Reviews - Most Popular (>4.5) 2005-2015
Year 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
Lone Survivor
The Good Wife, Season 5
Warehouse 13 Season 4
Pride And Prejudice Season 1
A Christmas Carol
Captain Phillips
Duck Dynasty Season 1
American Horror Story: Murder House
The Good Wife, Season 4
Star Trek
Falling Skies Season 3
Creative Galaxy Season 1
Firefly Season 1
The Way
Masterpiece: Mr. Selfridge Original UK Edition Season 2
Broad City Season 1
Suits Season 4
The Blacklist Season 2
Falling Skies Season 2
The Good Wife, Season 3
Fringe: The Complete First Season
The Hundred-Foot Journey
Serenity
Hannibal Season 2
Shaun The Sheep Season 1
Hachi: A Dog's Tale
Tangled
Sherlock Season 1
A Christmas Story
Rome Season 2
0K
10K
20K
30K
40K
50K

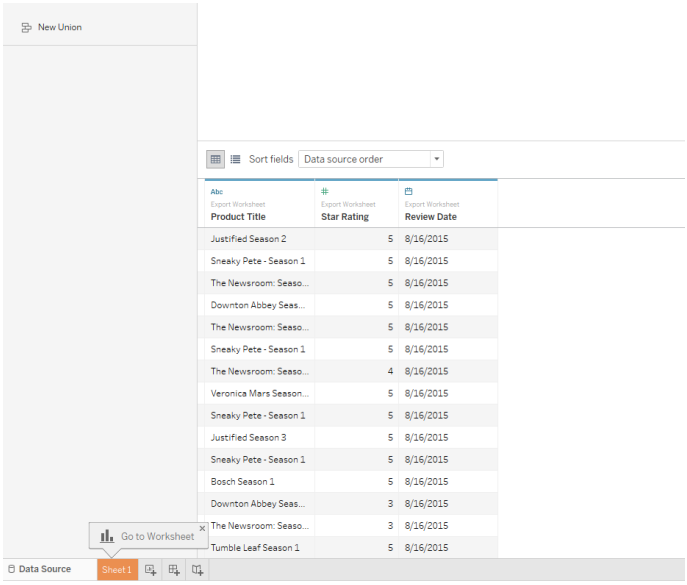
Visualization #2 – Tableau (Animated Data Visualization)

(Total Count of Star Ratings vs Product Title/Year/Rank)

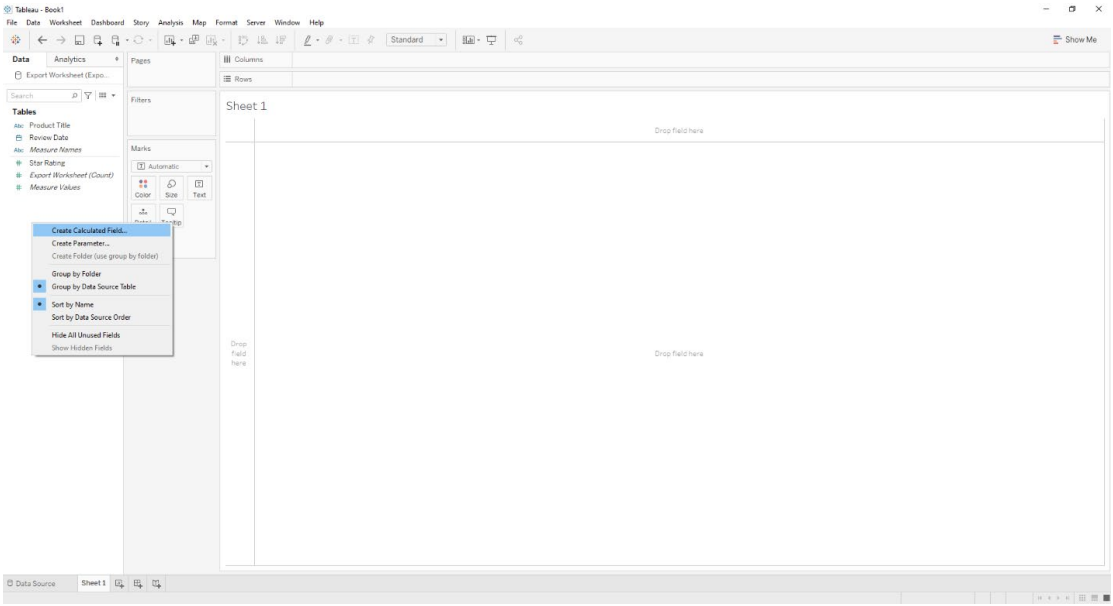
Step 1:
By starting on the animated visualization graph, first you open Tableau and then drag your Excel file to Tableau.



Step 2:
After making sure the data displayed is correct, create a new sheet by clicking on Sheet1.

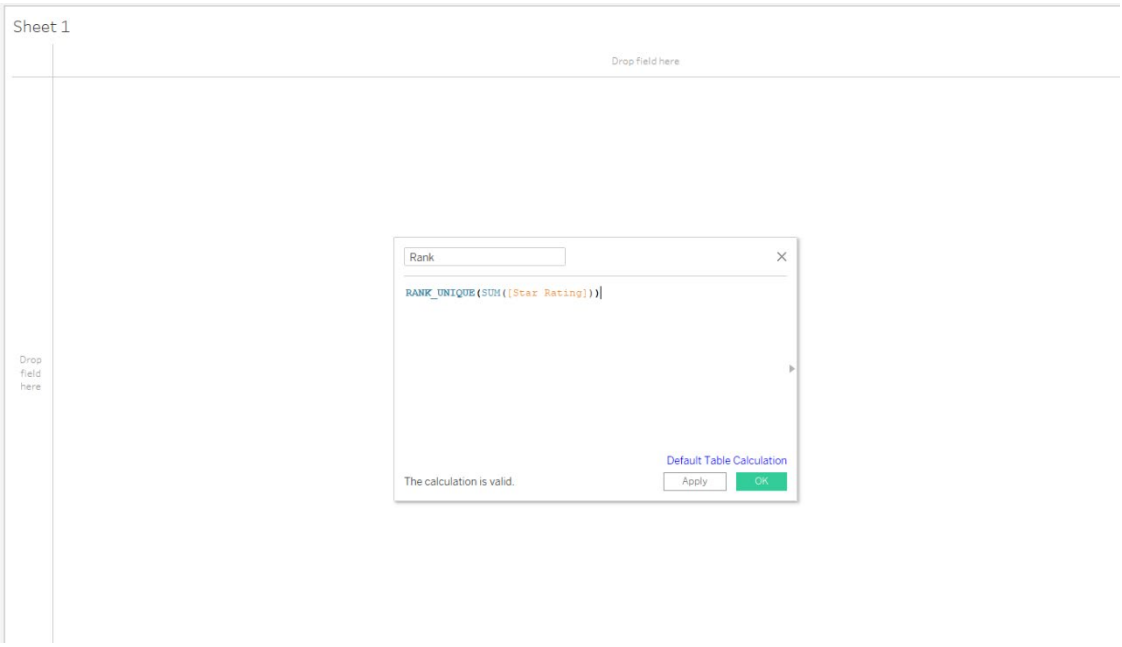


Step 3:
Right click on the empty area under tables to create a calculated field for “Rank.”



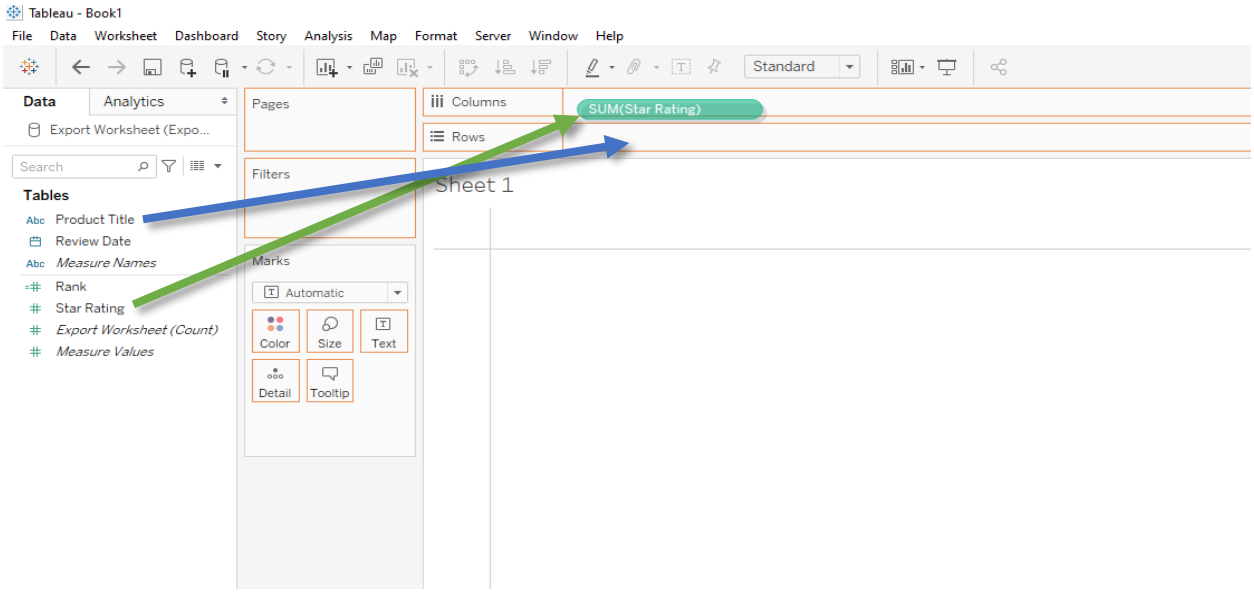
Step 4:

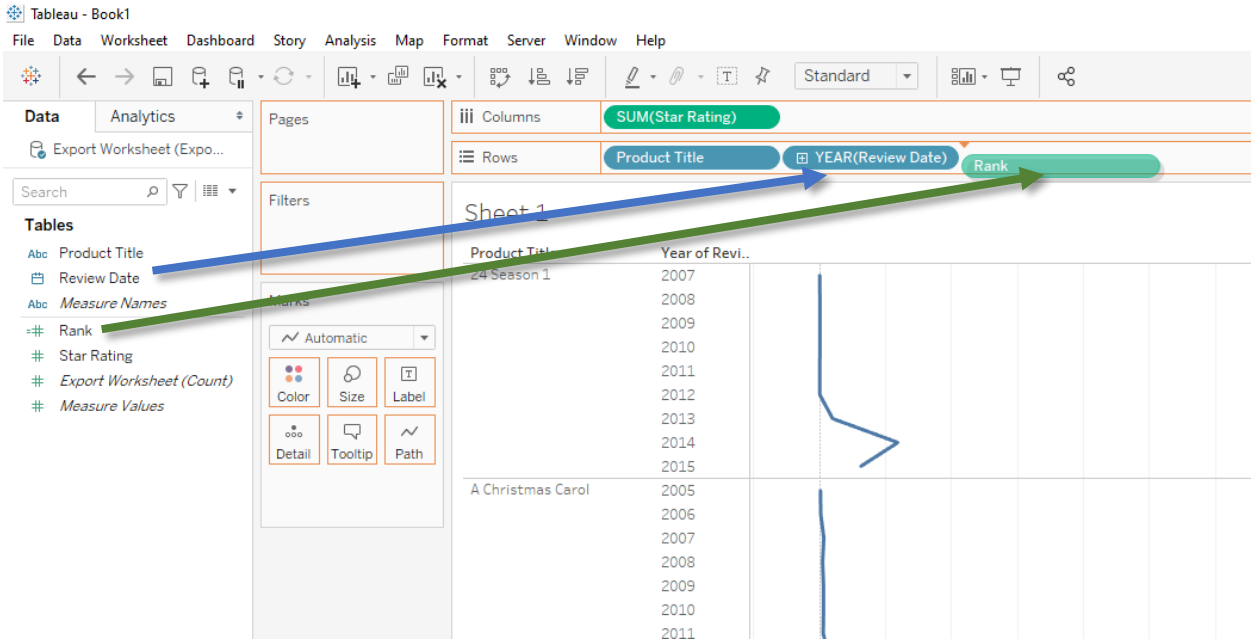
You must make sure to label your field as “Rank” to differentiate your fields. The code RANK_UNIQUE(SUM([Insert_Your_Field_Here])), allows you to have ranking for your rows or columns.



Step 5:

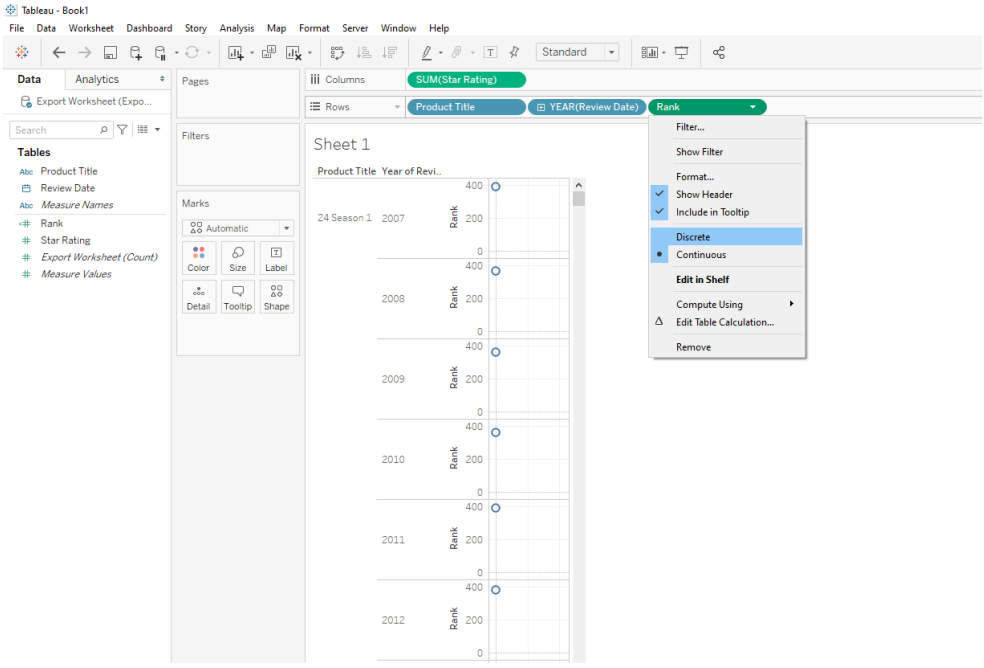
Drag the fields (Under Tables), “Star Rating” to Columns, “Product Title,” “Review Date,” and “Rank” to Rows.

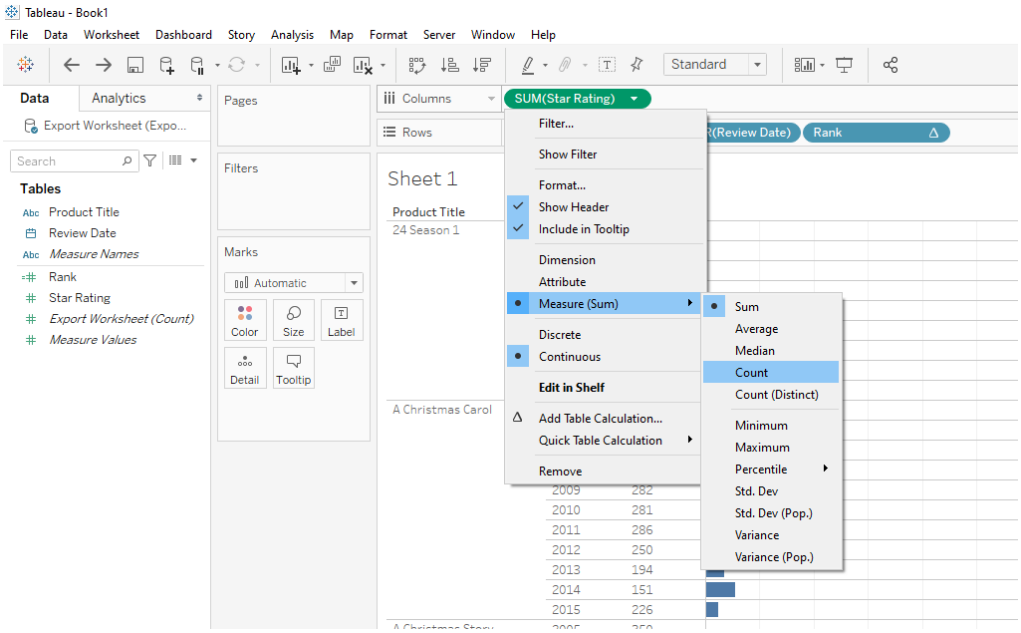




Step 6:

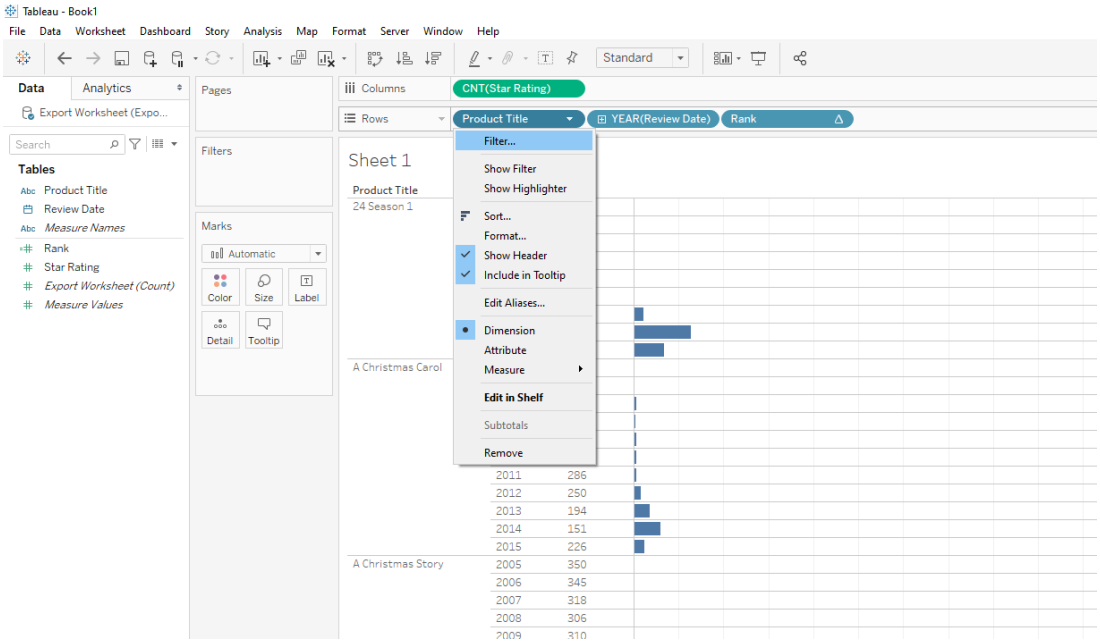
To change the layout of the chart to a Horizontal Bar Chart, move your mouse to rank and click on the arrow. Then click on “Discrete.” After that, move your mouse to Sum(Star Rating) and click on the arrow. Move your mouse down to Measure (Sum), then click on count.





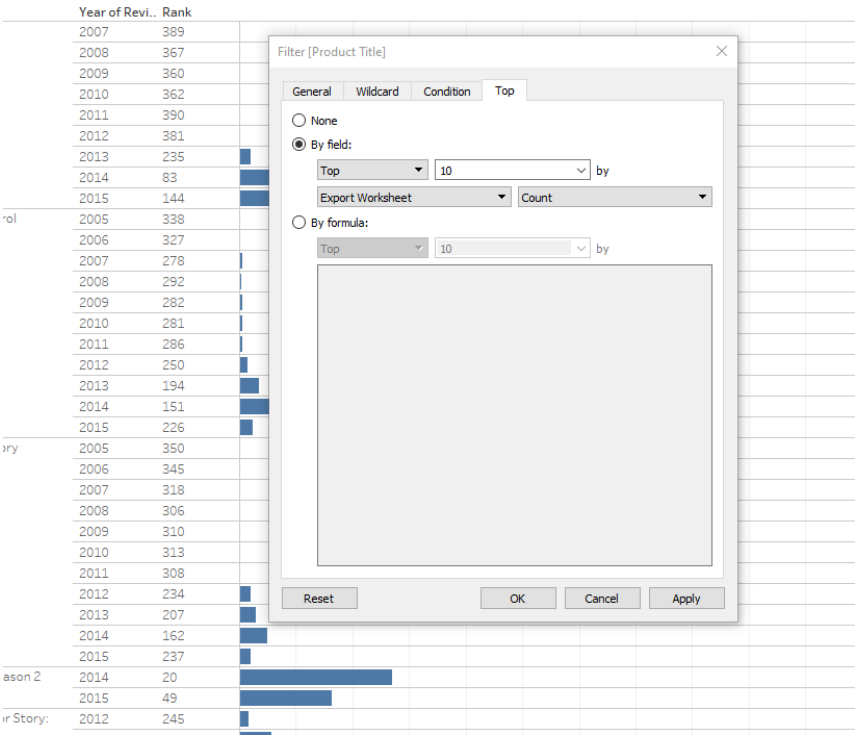
Step 7:

To show only the Top 10 Product Titles, move the mouse to Product Title and click on the arrow. Then click on Filter.



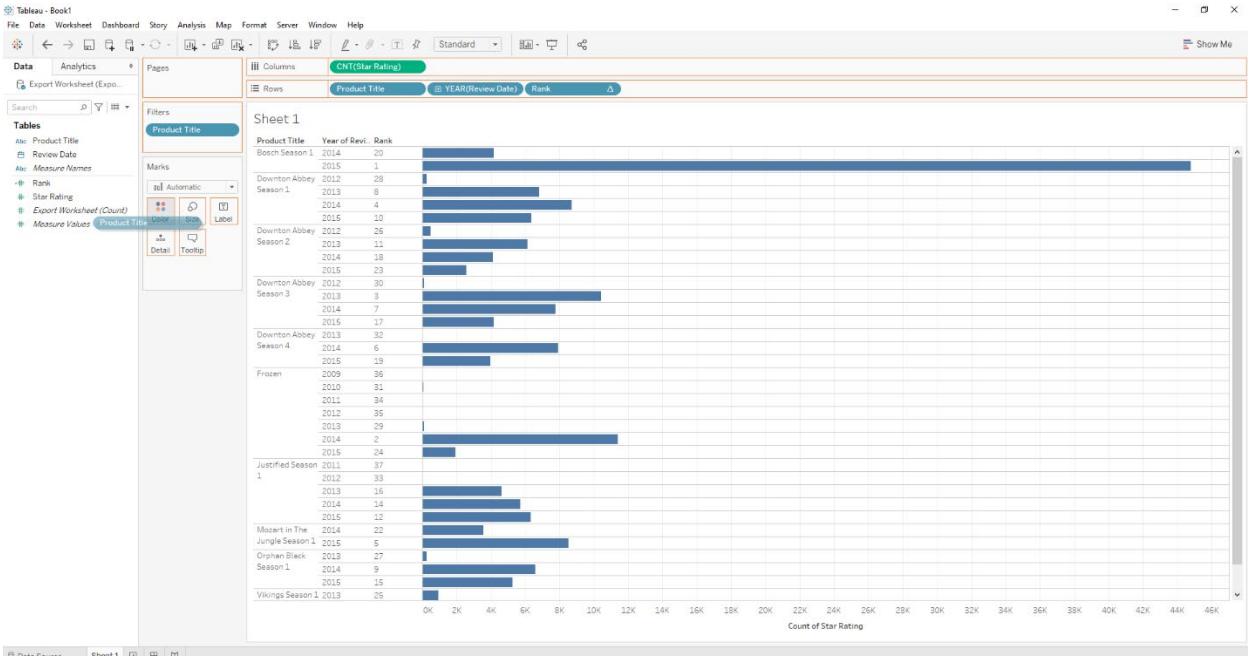
Step 8:

Click on the Top tab and then click on “By Field.” Since it will automatically show “Top 10” like the diagram below, click on Apply and OK.



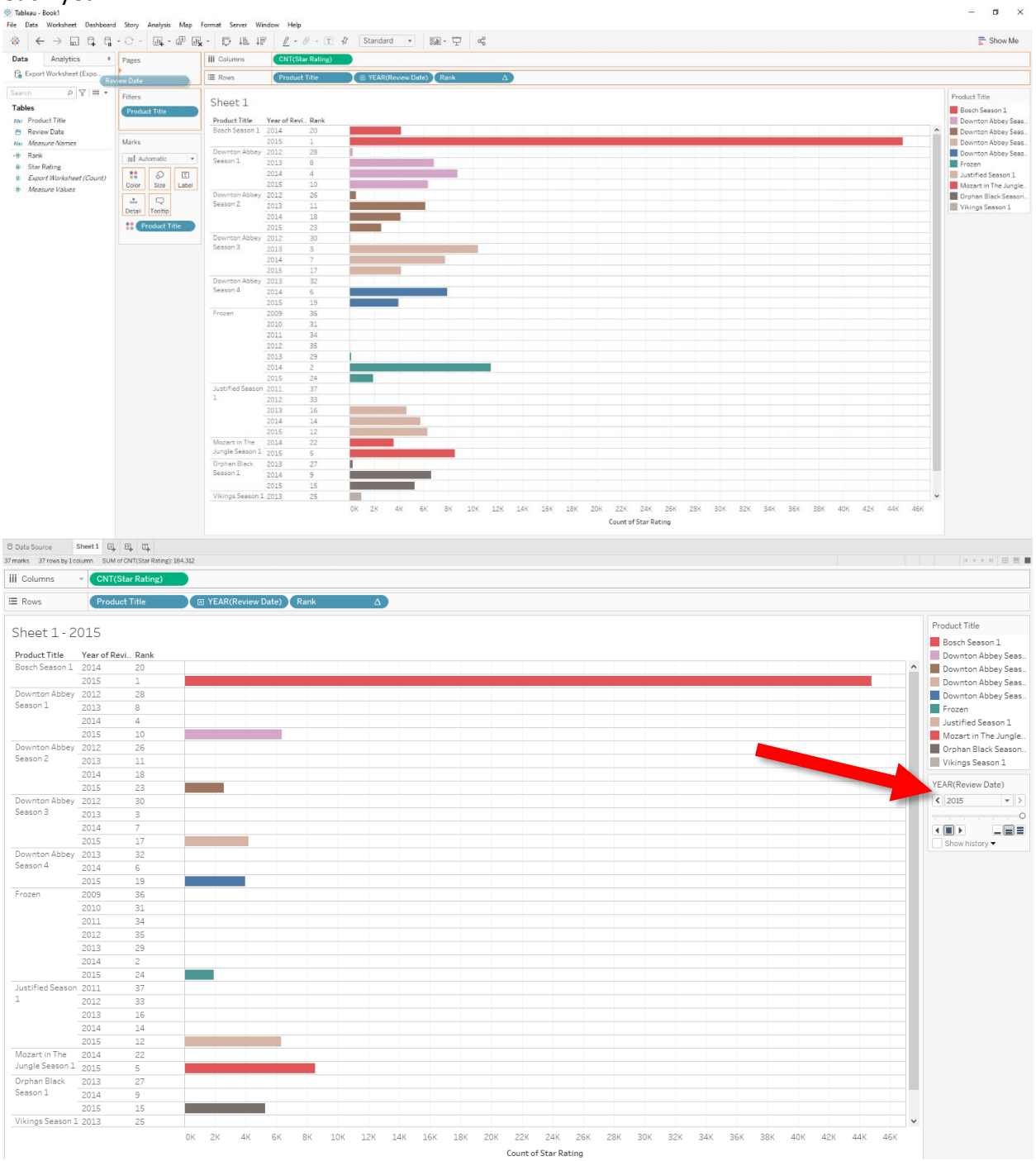
Step 9:

To assign different bar colors for each product title, drag the Product Title field to the Color mini box in the Marks box.

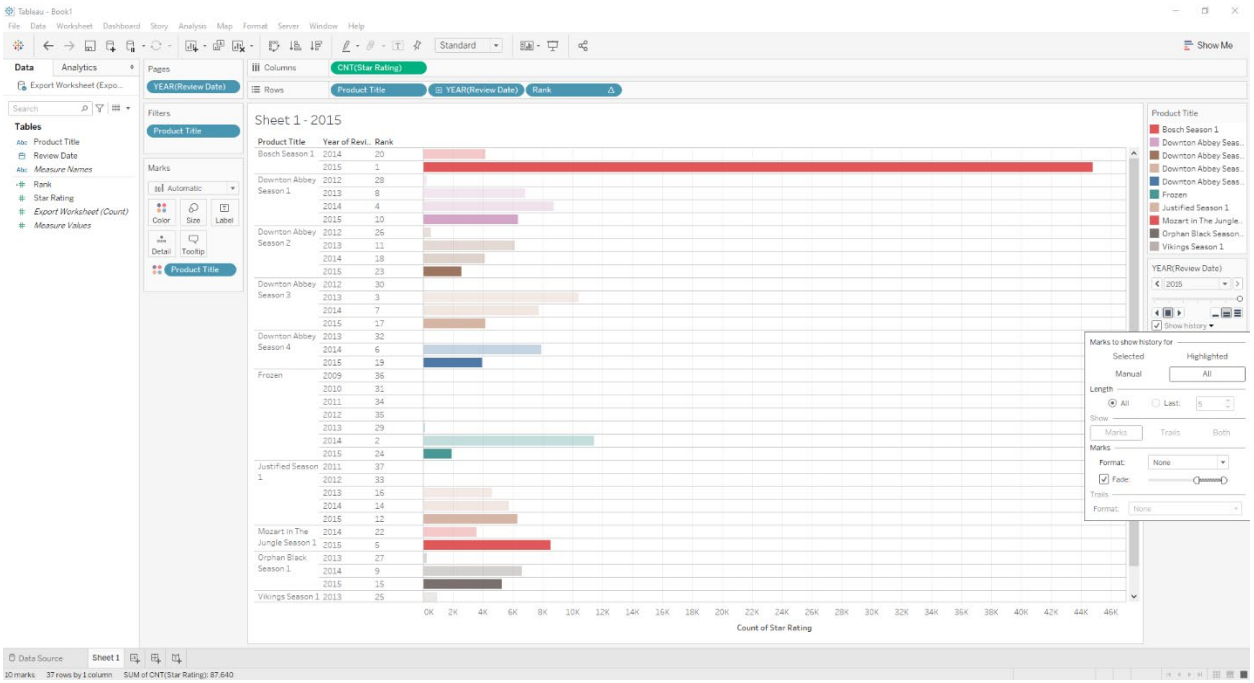


Step 10:

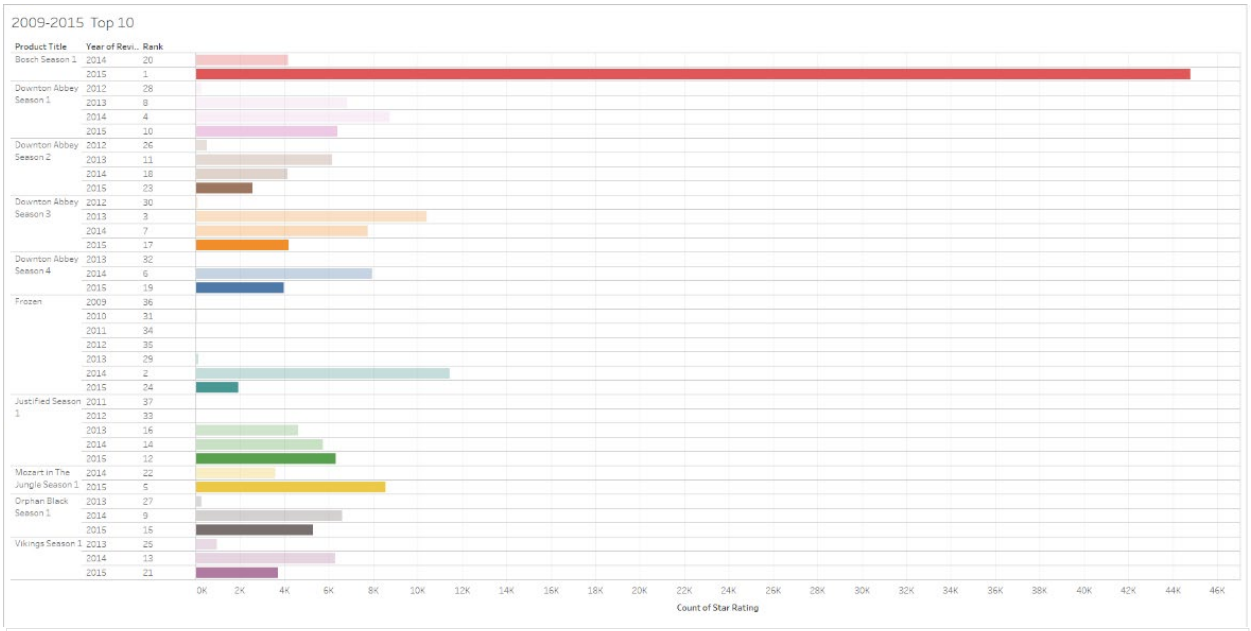
To begin with the animation, drag the Review Date field to the Pages box. Then the Animation box will appear on the right side of Tableau. To see the chart animation, click on the right arrow (Play Button) to see progress year after year. Left arrow goes backward (Reverse Button), and the middle button stops the animation. You can also drag the circle button both ways to see each year.

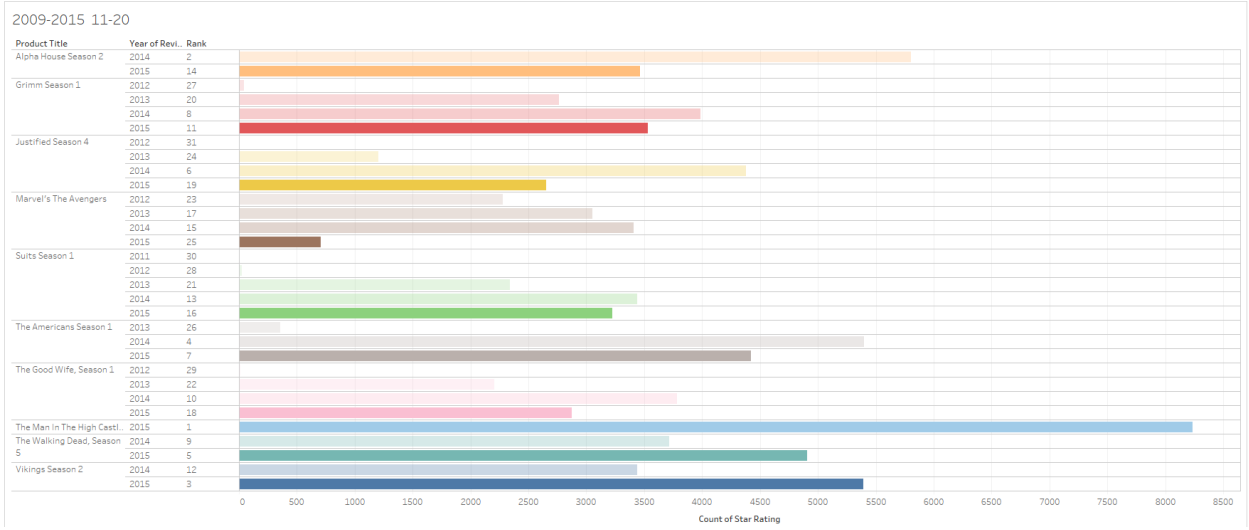


- Also, when clicking on the Show History Button, click on All to see bars faded from previous years as the animation is being played. You can drag the fade to your preference.



Final Graphs from Tableau:





References

1. URL of Data Source, URL of Data Source, https://s3.amazonaws.com/amazon-reviews-pds/tsv/index.txt

2. URL of Project Github: https://github.com/NinaRo2/CIS_5200

3. Other reference: https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF