HOMEWORK IN PIG

Case study: film classification

Debora Russo
Big Data Course

Summary

Introduction

We want to process data related to a set of films.

The first dataset (https://grouplens.org/datasets/movielens/), updated in September 2018, consists of the following attributes:

- MovieId
- Title: movie title and release year
- Genres: genres that describe the film

The second dataset instead consists of:

- UserId
- MovieId
- Rating
- Timestamp

Target analysis

We want to get the list of films in ascending alphabetical order whose genres are "Comedy" and "Drama", and the list of films whose title begins with the letter T.

We then want to get a ranking of films whose genre is "Drama" and the rating is 5.

Finally, we want to get the average rating for the movie "The Outlaw Josey Wales".

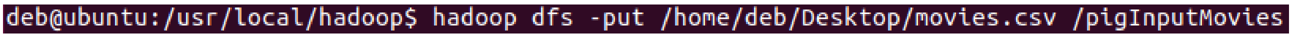
Everything was done thanks to the Linux Ubuntu operating system.

Execution (Interactive Mode)

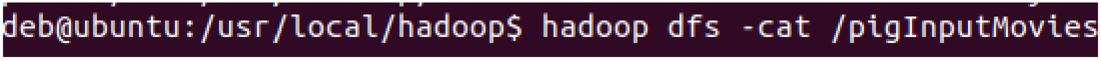
1. Let’s start Hadoop:

   bin/hdfs namenome –format
   sbin/start-dfs.sh

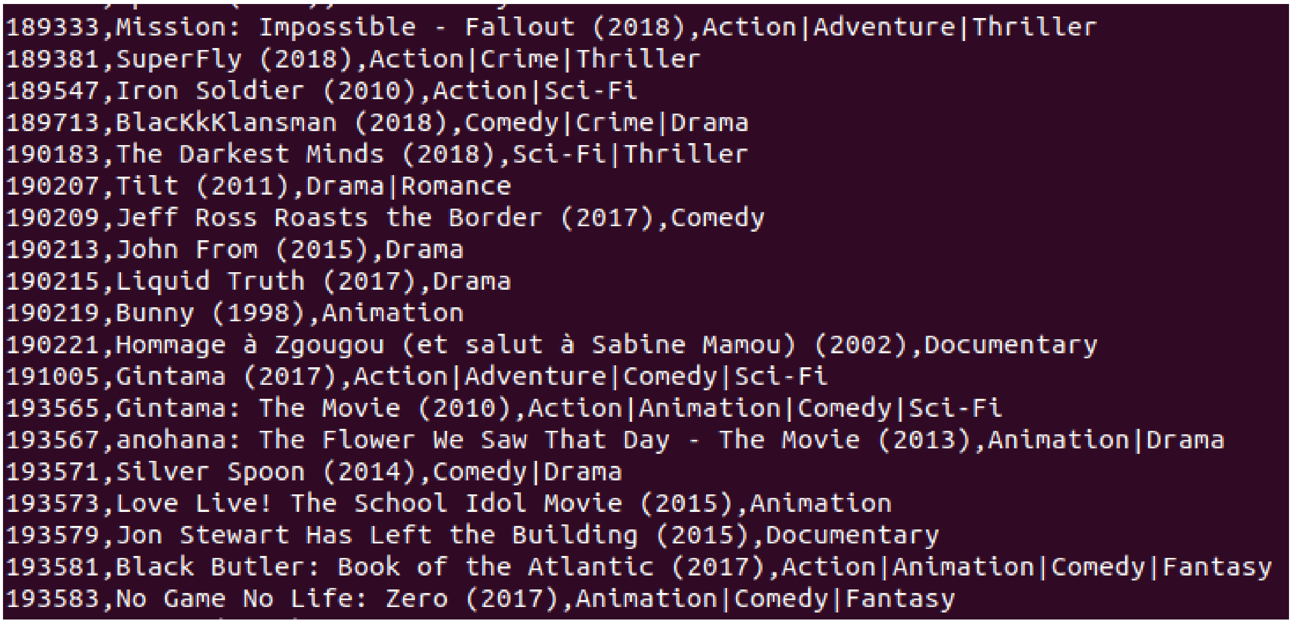
2. We move the .csv file from Desktop into HDFS, creating a folder called pigInputMovies.
   After that, we make sure that the file is present:



```
deb@ubuntu:/usr/local/hadoop$ hadoop dfs -put /home/deb/Desktop/movies.csv /pigInputMovies
```

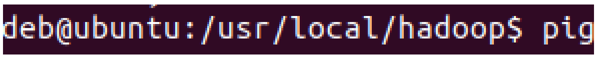


```
deb@ubuntu:/usr/local/hadoop$ hadoop dfs -cat /pigInputMovies
```

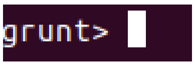


```
189333,Mission: Impossible - Fallout (2018),Action|Adventure|Thriller
189381,SuperFly (2018),Action|Crime|Thriller
189547,Iron Soldier (2010),Action|Sci-Fi
189713,BlacKkKlansman (2018),Comedy|Crime|Drama
190183,The Darkest Minds (2018),Sci-Fi|Thriller
190207,Tilt (2011),Drama|Romance
190209,Jeff Ross Roasts the Border (2017),Comedy
190213,John From (2015),Drama
190215,Liquid Truth (2017),Drama
190219,Bunny (1998),Animation
190221,Hommage à Zgougou (et salut à Sabine Mamou) (2002),Documentary
191005,Gintama (2017),Action|Adventure|Comedy|Sci-Fi
193565,Gintama: The Movie (2010),Action|Animation|Comedy|Sci-Fi
193567,anohana: The Flower We Saw That Day - The Movie (2013),Animation|Drama
193571,Silver Spoon (2014),Comedy|Drama
193573,Love Live! The School Idol Movie (2015),Animation
193579,Jon Stewart Has Left the Building (2015),Documentary
193581,Black Butler: Book of the Atlantic (2017),Action|Animation|Comedy|Fantasy
193583,No Game No Life: Zero (2017),Animation|Comedy|Fantasy
```

3. Let’s start Pig in MapReduce mode:

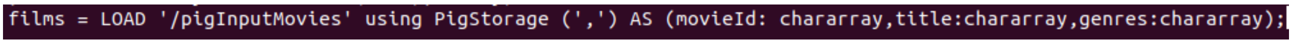


```
deb@ubuntu:/usr/local/hadoop$ pig
```

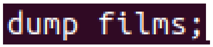


```
grunt>
```

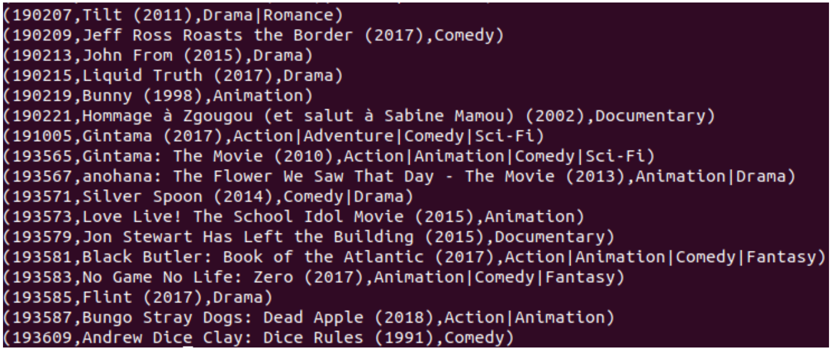
4. We load the data found in HDFS inside Apache Pig, and check if they are present:



```
films = LOAD '/pigInputMovies' using PigStorage (',') AS (movieId: chararray,title:chararray,genres:chararray);
```

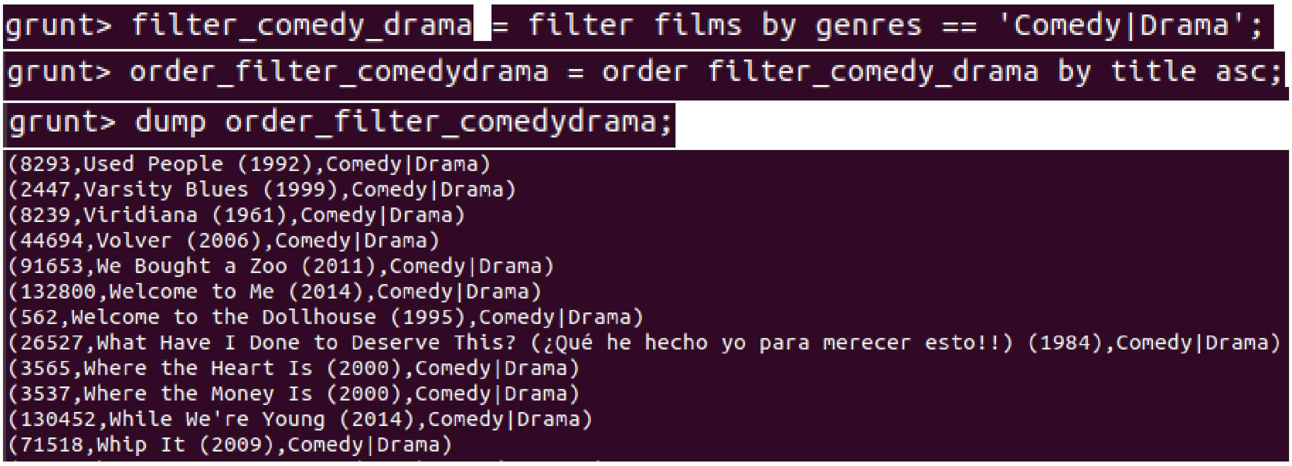


```
dump films;
```



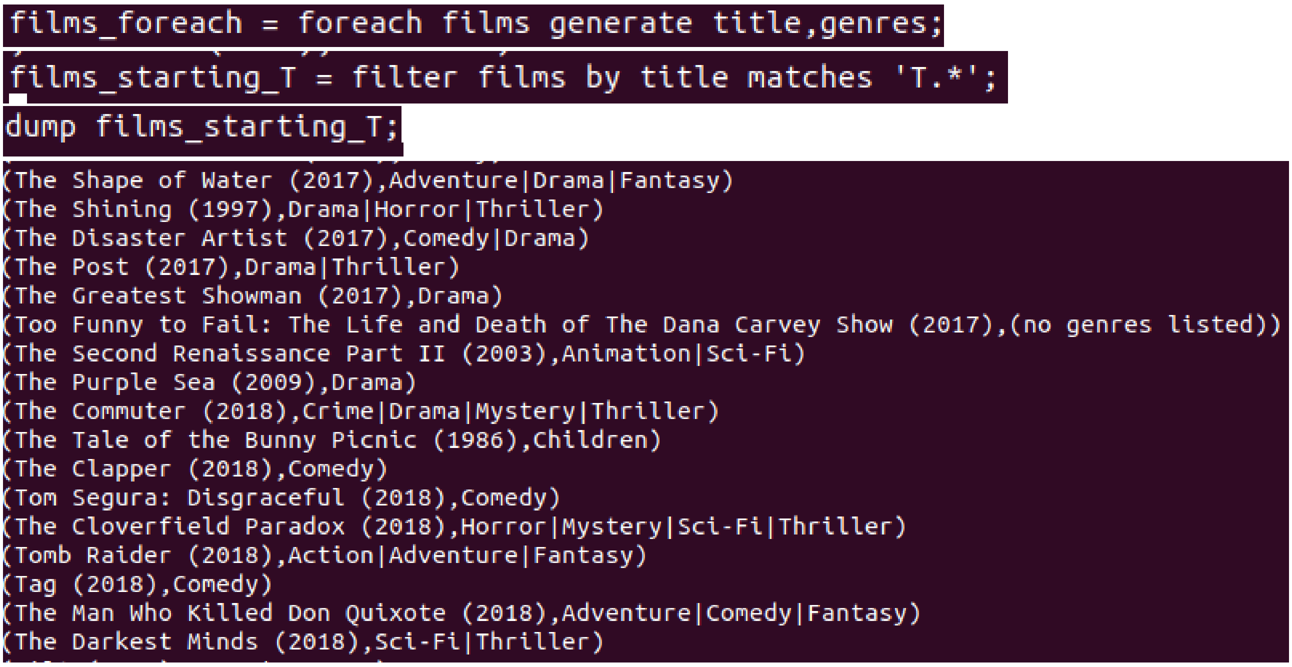
```
(190207,Tilt (2011),Drama|Romance)
(190209,Jeff Ross Roasts the Border (2017),Comedy)
(190213,John From (2015),Drama)
(190215,Liquid Truth (2017),Drama)
(190219,Bunny (1998),Animation)
(190221,Hommage à Zgougou (et salut à Sabine Mamou) (2002),Documentary)
(191005,Gintama (2017),Action|Adventure|Comedy|Sci-Fi)
(193565,Gintama: The Movie (2010),Action|Animation|Comedy|Sci-Fi)
(193567,anohana: The Flower We Saw That Day - The Movie (2013),Animation|Drama)
(193571,Silver Spoon (2014),Comedy|Drama)
(193573,Love Live! The School Idol Movie (2015),Animation)
(193579,Jon Stewart Has Left the Building (2015),Documentary)
(193581,Black Butler: Book of the Atlantic (2017),Action|Animation|Comedy|Fantasy)
(193583,No Game No Life: Zero (2017),Animation|Comedy|Fantasy)
(193585,Flint (2017),Drama)
(193587,Bungo Stray Dogs: Dead Apple (2018),Action|Animation)
(193609,Andrew Dice Clay: Dice Rules (1991),Comedy)
```

5. We order the list of comedy and drama films in ascending alphabetical order:



```
grunt> filter_comedy_drama = filter films by genres == 'Comedy|Drama';
grunt> order_filter_comedydrama = order filter_comedy_drama by title asc;
grunt> dump order_filter_comedydrama;
(8293,Used People (1992),Comedy|Drama)
(2447,Varsity Blues (1999),Comedy|Drama)
(8239,Viridiana (1961),Comedy|Drama)
(44694,Volver (2006),Comedy|Drama)
(91653,We Bought a Zoo (2011),Comedy|Drama)
(132800,Welcome to Me (2014),Comedy|Drama)
(562,Welcome to the Dollhouse (1995),Comedy|Drama)
(26527,What Have I Done to Deserve This? (¿Qué he hecho yo para merecer esto!!) (1984),Comedy|Drama)
(3565,Where the Heart Is (2000),Comedy|Drama)
(3537,Where the Money Is (2000),Comedy|Drama)
(130452,While We're Young (2014),Comedy|Drama)
(71518,Whip It (2009),Comedy|Drama)
```

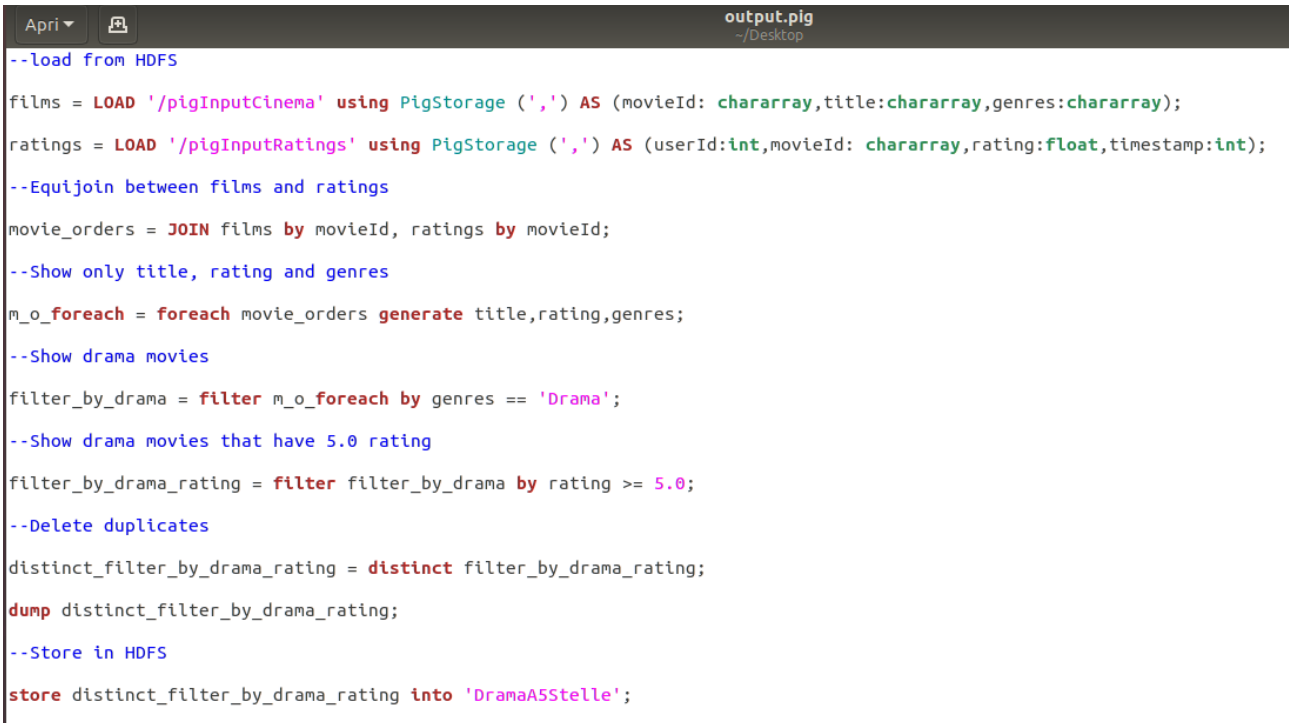
6. We find the list of films starting with the letter T. This time we only show the title and genres:



```
films_foreach = foreach films generate title,genres;
films_starting_T = filter films by title matches 'T.*';
dump films_starting_T;
(The Shape of Water (2017),Adventure|Drama|Fantasy)
(The Shining (1997),Drama|Horror|Thriller)
(The Disaster Artist (2017),Comedy|Drama)
(The Post (2017),Drama|Thriller)
(The Greatest Showman (2017),Drama)
(Too Funny to Fail: The Life and Death of The Dana Carvey Show (2017),(no genres listed))
(The Second Renaissance Part II (2003),Animation|Sci-Fi)
(The Purple Sea (2009),Drama)
(The Commuter (2018),Crime|Drama|Mystery|Thriller)
(The Tale of the Bunny Picnic (1986),Children)
(The Clapper (2018),Comedy)
(Tom Segura: Disgraceful (2018),Comedy)
(The Cloverfield Paradox (2018),Horror|Mystery|Sci-Fi|Thriller)
(Tomb Raider (2018),Action|Adventure|Fantasy)
(Tag (2018),Comedy)
(The Man Who Killed Don Quixote (2018),Adventure|Comedy|Fantasy)
(The Darkest Minds (2018),Sci-Fi|Thriller)
```

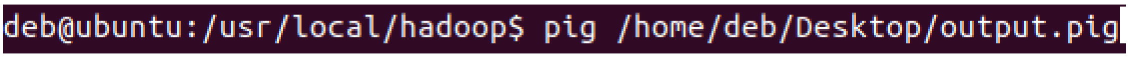
Execution (Batch mode)

Now let's write a script that allows us to get a ranking of the films whose genre is "Drama" (Drama) and the rating is 5.

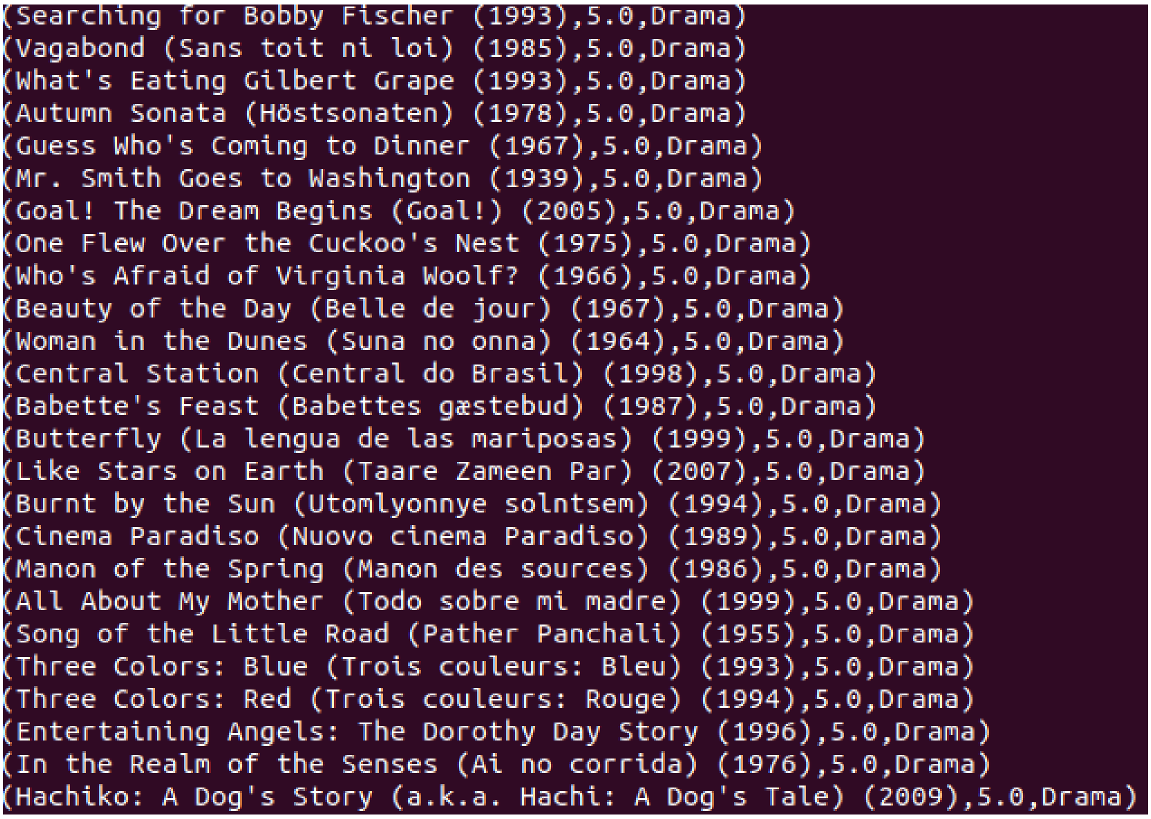


```
--load from HDFS

films = LOAD '/pigInputCinema' using PigStorage (',') AS (movieId: chararray,title:chararray,genres:chararray);

ratings = LOAD '/pigInputRatings' using PigStorage (',') AS (userId:int,movieId: chararray,rating:float,timestamp:int);

--Equijoin between films and ratings

movie_orders = JOIN films by movieId, ratings by movieId;

--Show only title, rating and genres

m_o_foreach = foreach movie_orders generate title,rating,genres;

--Show drama movies

filter_by_drama = filter m_o_foreach by genres == 'Drama';

--Show drama movies that have 5.0 rating

filter_by_drama_rating = filter filter_by_drama by rating >= 5.0;

--Delete duplicates

distinct_filter_by_drama_rating = distinct filter_by_drama_rating;

dump distinct_filter_by_drama_rating;

--Store in HDFS

store distinct_filter_by_drama_rating into 'DramaA5Stelle';
```

Let’s execute the script:

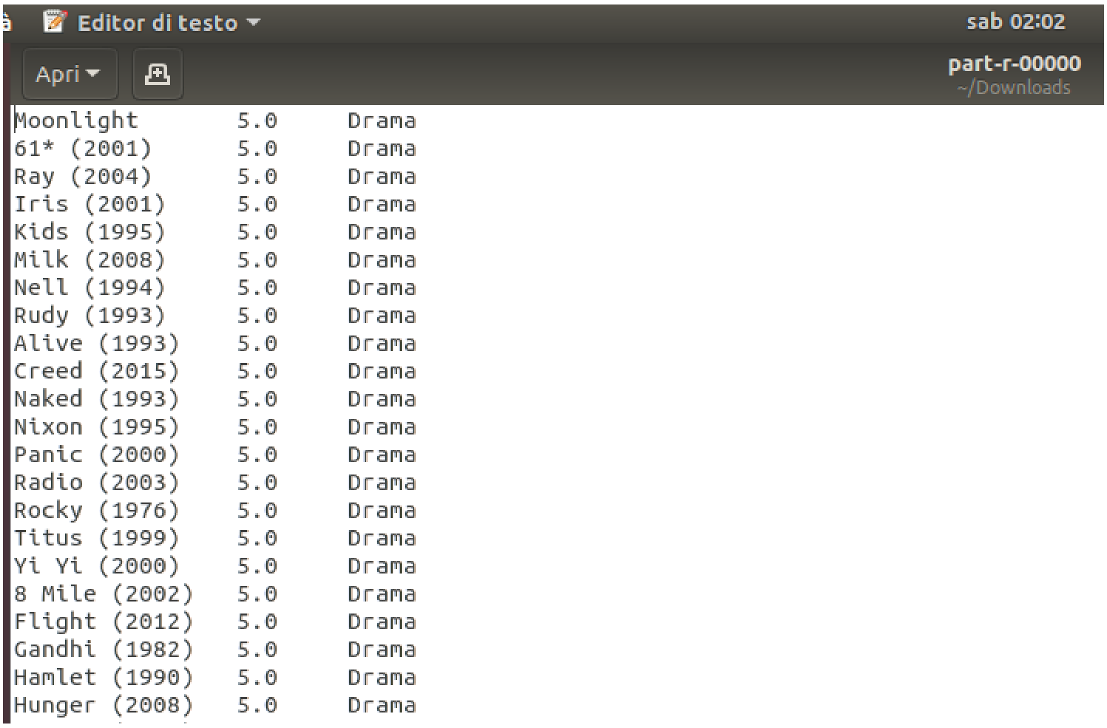


```
deb@ubuntu:/usr/local/hadoop$ pig /home/deb/Desktop/output.pig
```



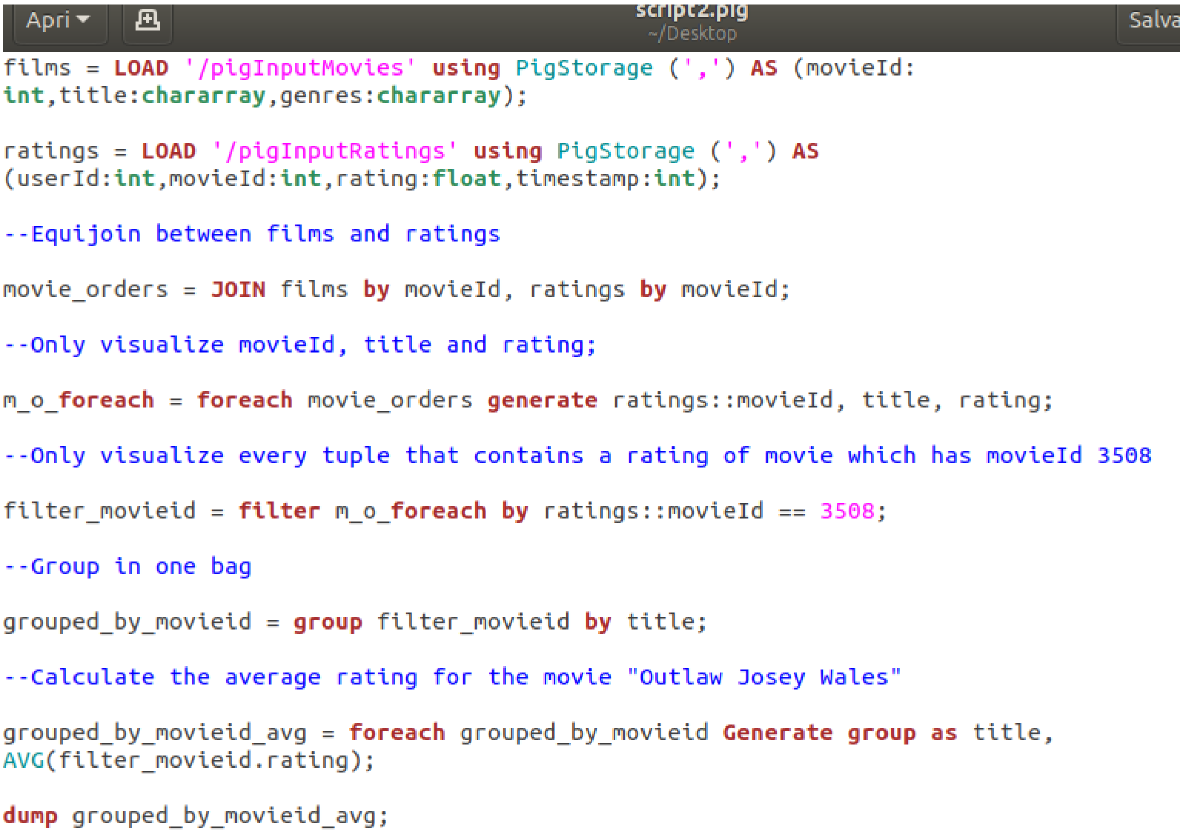
```
(Searching for Bobby Fischer (1993),5.0,Drama)
(Vagabond (Sans toit ni loi) (1985),5.0,Drama)
(What's Eating Gilbert Grape (1993),5.0,Drama)
(Autumn Sonata (Höstsonaten) (1978),5.0,Drama)
(Guess Who's Coming to Dinner (1967),5.0,Drama)
(Mr. Smith Goes to Washington (1939),5.0,Drama)
(Goal! The Dream Begins (Goal!) (2005),5.0,Drama)
(One Flew Over the Cuckoo's Nest (1975),5.0,Drama)
(Who's Afraid of Virginia Woolf? (1966),5.0,Drama)
(Beauty of the Day (Belle de jour) (1967),5.0,Drama)
(Woman in the Dunes (Suna no onna) (1964),5.0,Drama)
(Central Station (Central do Brasil) (1998),5.0,Drama)
(Babette's Feast (Babettes gæstebud) (1987),5.0,Drama)
(Butterfly (La lengua de las mariposas) (1999),5.0,Drama)
(Like Stars on Earth (Taare Zameen Par) (2007),5.0,Drama)
(Burnt by the Sun (Utomlyonnye solntsem) (1994),5.0,Drama)
(Cinema Paradiso (Nuovo cinema Paradiso) (1989),5.0,Drama)
(Manon of the Spring (Manon des sources) (1986),5.0,Drama)
(All About My Mother (Todo sobre mi madre) (1999),5.0,Drama)
(Song of the Little Road (Pather Panchali) (1955),5.0,Drama)
(Three Colors: Blue (Trois couleurs: Bleu) (1993),5.0,Drama)
(Three Colors: Red (Trois couleurs: Rouge) (1994),5.0,Drama)
(Entertaining Angels: The Dorothy Day Story (1996),5.0,Drama)
(In the Realm of the Senses (Ai no corrida) (1976),5.0,Drama)
(Hachiko: A Dog's Story (a.k.a. Hachi: A Dog's Tale) (2009),5.0,Drama)
```

The data is subsequently saved within the HDFS with the store command:



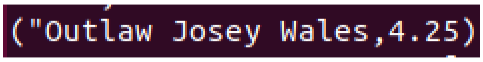
```
Moonlight       5.0     Drama
61* (2001)      5.0     Drama
Ray (2004)      5.0     Drama
Iris (2001)     5.0     Drama
Kids (1995)     5.0     Drama
Milk (2008)     5.0     Drama
Nell (1994)     5.0     Drama
Rudy (1993)     5.0     Drama
Alive (1993)    5.0     Drama
Creed (2015)    5.0     Drama
Naked (1993)    5.0     Drama
Nixon (1995)    5.0     Drama
Panic (2000)    5.0     Drama
Radio (2003)    5.0     Drama
Rocky (1976)    5.0     Drama
Titus (1999)    5.0     Drama
Yi Yi (2000)    5.0     Drama
8 Mile (2002)   5.0     Drama
Flight (2012)   5.0     Drama
Gandhi (1982)   5.0     Drama
Hamlet (1990)   5.0     Drama
Hunger (2008)   5.0     Drama
```

As for the second script instead:



```
films = LOAD '/pigInputMovies' using PigStorage (',') AS (movieId:
int,title:chararray,genres:chararray);

ratings = LOAD '/pigInputRatings' using PigStorage (',') AS
(userId:int,movieId:int,rating:float,timestamp:int);

--Equijoin between films and ratings

movie_orders = JOIN films by movieId, ratings by movieId;

--Only visualize movieId, title and rating;

m_o_foreach = foreach movie_orders generate ratings::movieId, title, rating;

--Only visualize every tuple that contains a rating of movie which has movieId 3508

filter_movieid = filter m_o_foreach by ratings::movieId == 3508;

--Group in one bag

grouped_by_movieid = group filter_movieid by title;

--Calculate the average rating for the movie "Outlaw Josey Wales"

grouped_by_movieid_avg = foreach grouped_by_movieid Generate group as title,
AVG(filter_movieid.rating);

dump grouped_by_movieid_avg;
```

We can see how the average rating for the movie "The Outlaw Josey Wales" is 4.25:



```
("Outlaw Josey Wales,4.25)
```