Northeastern University, Khoury College of Computer Science

CS 6220 Data Mining — Assignment 1

Due: January 18, 2022 (100 points)

Wei Zhang
joungway
zhang.wei12@northeastern.edu

1 Getting Started - 10 points

1.1 Using Docker

Different companies use different tools for development and different work environments. For future assignments, we won't be prescriptive, but in this homework, we're going to familiarize ourselves with some of the most useful and common delivery and development environment tools in industry today.
Docker http://www.Docker.com is a useful mechanism for delivering software or scaling it up. For example, say we want to run a multi-computer job, passing *Docker containers* to each of the nodes in the cluster is one way to have repetitive and predictable behavior when doing large scale compute.
There are two essential Docker units: a **container** and a **container image**.

1. A **container** is a sandboxed process on your machine that is isolated from all other processes on the host machine. That isolation leverages kernel namespaces and cgroups, features that have been in Linux for a long time. Docker has worked to make these capabilities approachable and easy to use. To summarize, a container:
   a) is a runnable instance of an image. You can create, start, stop, move, or delete a container using the DockerAPI or CLI.
   b) can be run on local machines, virtual machines or deployed to the cloud.
   c) is portable (can be run on any OS).
   d) is isolated from other containers and runs its own software, binaries, and configurations.
2. When running a container, it uses an isolated filesystem. This custom filesystem is provided by a **container image**. Since the image contains the container's filesystem, it

must contain everything needed to run an application - all dependencies, configurations, scripts, binaries, etc. The image also contains other configuration for the container, such as environment variables, a default command to run, and other metadata.

Go ahead and download and install Docker. The getting started guide on Docker has detailed instructions for setting up Docker on

- Mac `https://docs.docker.com/desktop/install/mac-install/`,
- Linux `https://docs.docker.com/install/linux/docker-ce/ubuntu`
- Windows `https://docs.docker.com/docker-for-windows/install`.

1.2 Executing Your "Hello World"

For this assignment, we'll start with creating a Dockerfile in your submission folder. Specify the operating system and version of Python in the Dockerfile. You will subsequently need to install Python and libraries that you anticipate importing. Do not add the data into the image; you will need to pass that into the container with the `-v` Docker option.
For example, here's the most basic `Dockerfile`:

```
FROM ubuntu:20.04

RUN apt update && apt install -y sbcl
WORKDIR /usr/src
```

For this assignment, you'll set up your Docker environment and the appropriate versions of Python. Specifically,

1. Download and install Docker
2. Create your Dockerfile
3. Compile your Docker image
4. Screenshot a list of the Docker images available
5. Screenshot a list of the running Docker containers that include one with the image you created
6. Include both screenshots and the command you used in your write up

1.3 Github

Software version control at companies is essential for every software company in the industry. There are several types, including *Subversion/SVN* (which Google uses its in-house version branched from SVN). The most popular tool of choice is Github, which Microsoft recently bought.
At the end of this assignment, your submission will point to a repository, where the following files will be reviewed and subsequently graded:

- `Dockerfile` specifying what packages that you've used

- assignment1.tex file with your homework writeup
- assignment1.pdf file of the compiled version of your *.tex file
- assignment1.py file of your working code

None of the other files in that repository will be reviewed. We've provided a LaTeXtemplate that you can use for submission, provided here:

- https://github.com/kni-neu/homework-1/blob/main/assignment1_template.tex

Do *NOT* include data into your Git repository. If you need help with LaTeX, the program that creates a PDF file from a coded text file (with extension ⋆.tex), you may wish to use the online site overleaf.com. There is a helpful guide at this url:
https://www.overleaf.com/learn

2 Identifying All Sets - 40 points

In subsequent lectures, you'll learn about frequent item sets, where relationships between items are learned by observing how often they co-occur in a set of data. This information is useful for making recommendations in a rule based manner. Before looking at frequent item sets, it is worth understanding the space of all possible sets and get a sense for how quickly the number of sets with unique items grows.
Suppose that we've received only a hundred records of items bought by customers at a market. Each line in the file represents the items an individual customer bought, i.e. their basket. For example, consider the following rows.

```
ham, cheese, bread
dates, bananas
celery, chocolate bars
```

Customer 1 has a basket of ham, cheese, and bread. Customer 2 has a basket of dates and bananas. Customer 3 has a basket of celery and chocolate bars. Each of these records is the receipt of a given customer, identifying what they bought.

1. What is the cardinality of the full set of unique items? Write a function called cardinality_items that takes a .csv text string file as input, where the format is as the above, and calculates the cardinality of the dataset.
2. Taking any .csv file as a sample of a larger dataset, we'd occasionally like to understand the space of all possible subsets comprised of unique items. If there are N unique items (i.e., the cardinality of the entire dataset is N), how many sets with unique items can there possibly be? (Ignore the null set.)
3. Write a module called all_itemsets() with the following input/output:
   a) Input: $filename$ = the .csv text string file, where the format is as the above.
   b) Output: $L = [S_1, S_2, \cdots S_N]$ = A list of all possible sets of size N

4. Let's take the small sample `.csv` provided as reflective of the distribution of the receipts writ large. So, for example, if the set $S = \{\text{bread}, \text{oatmeal}\}$ occurs twice in a dataset with 100 records, then the probability of item set $\{\text{bread}, \text{oatmeal}\}$ occurring is 0.02. Write a module called `prob_S` with the following input/output:

   a) Input:
      S = the set in question
      D = the entire Dataset (which if it's in memory, Python will pass by reference)

   b) Output: $P(S)$ = the probability that S occurs

3 The Netflix Challenge - 50 points

One of the most famous challenges in data science and machine learning is Netflix's Grand Prize Challenge, where Netflix held an open competition for the best algorithm to predict user ratings for films. The grand prize was $1,000,000 and was won by BellKor's Pragmatic Chaos team. This is the dataset that was used in that competition.

- https://www.kaggle.com/datasets/netflix-inc/netflix-prize-data

In this exercise, we're going to do a bit of exploring in the Netflix Data. Start by downloading the data. If all worked out well, you should have the files in Fig. 3.3. The Kaggle dataset is close to 700MB large, and may take a long time to download. Do *not* include this data in your Docker container, but rather, mount the folder with the data.

3.1 Data Verification

Data integrity tends to be a problem in large scale processing, especially if there is little to no support. Therefore, it's important to verify the quality of the file download.

1. A large part of machine learning and data science is about getting data in the right format. Verify that the schema is the same as the Kaggle Dataset's description. Add screenshots to your assignment.

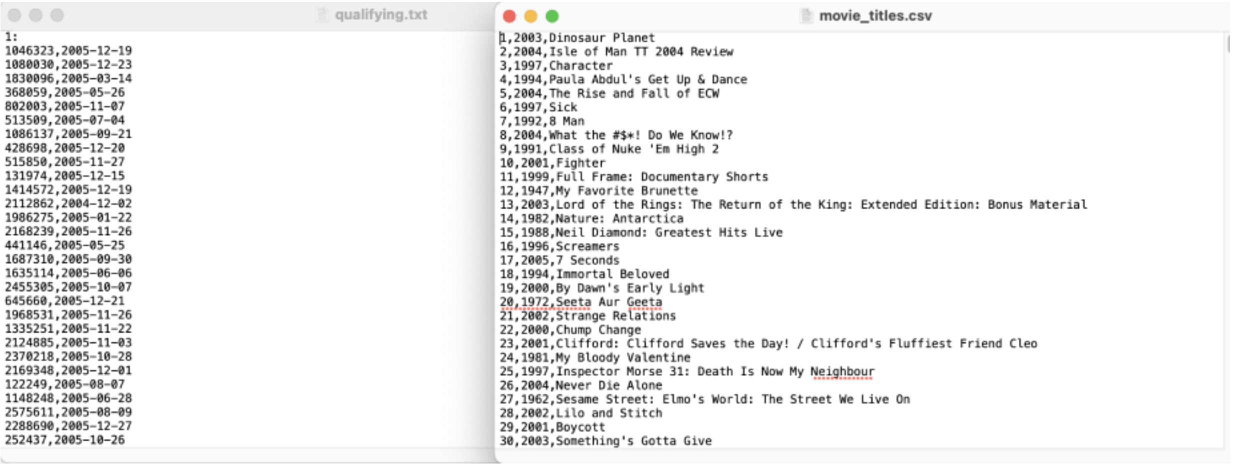
Answer: for "movie-titles.txt" the format that the author use is usable enough to be direct import as dataset. However, the "qualifying.txt" format is not usable enough. Because the placement of MovieID variable, which is place in the same column ad the customer ID, it is easy to be messed up if directly use. Screenshot has included in Fig. 3.1.

3.2 Data Analysis

Let's answer the following questions in your writeup:

1. How many total records are there? - Answer: the total record of movies is: 17770; the total record of users is: 480189.

2. Can you plot the distribution of star ratings over users and time? The granularity of the sliding window is at your discretion. Are there any trends?- Answer: from the plot in Fig. 3.2, we have known that there are more and more audiences rate the movie over time.



```
1:
1046323,2005-12-19
1080030,2005-12-23
1830096,2005-03-14
368059,2005-05-26
802003,2005-11-07
513509,2005-07-04
1086137,2005-09-21
428698,2005-12-20
515850,2005-11-27
131974,2005-12-15
1414572,2005-12-19
2112862,2004-12-02
1986275,2005-01-22
2168239,2005-11-26
441146,2005-05-25
1687310,2005-09-30
1635114,2005-06-06
2455305,2005-10-07
645660,2005-12-21
1968531,2005-11-26
1335251,2005-11-22
2124885,2005-11-03
2370218,2005-10-28
2169348,2005-12-01
122249,2005-08-07
1148248,2005-06-28
2575611,2005-08-09
2288690,2005-12-27
252437,2005-10-26
```

```
1,2003,Dinosaur Planet
2,2004,Isle of Man TT 2004 Review
3,1997,Character
4,1994,Paula Abdul's Get Up & Dance
5,2004,The Rise and Fall of ECW
6,1997,Sick
7,1992,8 Man
8,2004,What the #$*! Do We Know!?
9,1991,Class of Nuke 'Em High 2
10,2001,Fighter
11,1999,Full Frame: Documentary Shorts
12,1947,My Favorite Brunette
13,2003,Lord of the Rings: The Return of the King: Extended Edition: Bonus Material
14,1982,Nature: Antarctica
15,1988,Neil Diamond: Greatest Hits Live
16,1996,Screamers
17,2005,7 Seconds
18,1994,Immortal Beloved
19,2000,By Dawn's Early Light
20,1972,Seeta Aur Geeta
21,2002,Strange Relations
22,2000,Chump Change
23,2001,Clifford: Clifford Saves the Day! / Clifford's Fluffiest Friend Cleo
24,1981,My Bloody Valentine
25,1997,Inspector Morse 31: Death Is Now My Neighbour
26,2004,Never Die Alone
27,1962,Sesame Street: Elmo's World: The Street We Live On
28,2002,Lilo and Stitch
29,2001,Boycott
30,2003,Something's Gotta Give
```

Figure 3.1: Screenshot for verifying the Kaggle Dataset.

3. What percentage of the films have gotten *more* popular over time? - Answer: 17470 films exist in the ”qualify” record, which refers to this amount of film get user’s reviews and represent popularity. Since the total films in record is 17770, ther percentage of film have gotten more popular over time is: 17470/17770, result is approximate 98

4. How many films have been re-released? How do you know? - Answer: there are approximately 309 films have been re-released. The process is starting with finding the duplicate name in the dataset ”movie-title”. However, since it is hard to find exact match of the duplicate name, I filtered the data down to only use the name that contain space, which will increase the possibility of exact match.

5. What other information might we try to extract to better understand the data? For the questions that you may come up with (especially any time series data), make sure you back up your assertions with plots. Go ahead and play around with the data, and explore. - Answer: We might try to explore the relationship between amount of users that make rate for each film and the average star rate of each film to see the relationship between ”popularity” and rating.

6. What are some interesting problems that we might solve? (No need to actually solve them!) - Answer: we can analysis the dataset and see if the highest-rating movies is affected by time? Are they popular since released or just popular between a specific time range? This can help us to learn more about audience’s taste towards a movie.

4 Grading Criterion

A significant portion of the grading rubric is the presentation of your report. We’ll review:

1. the answers to questions.

2. your code and its legibility

3. the clarity of your write-up, including
   a) pipeline and code decisions,

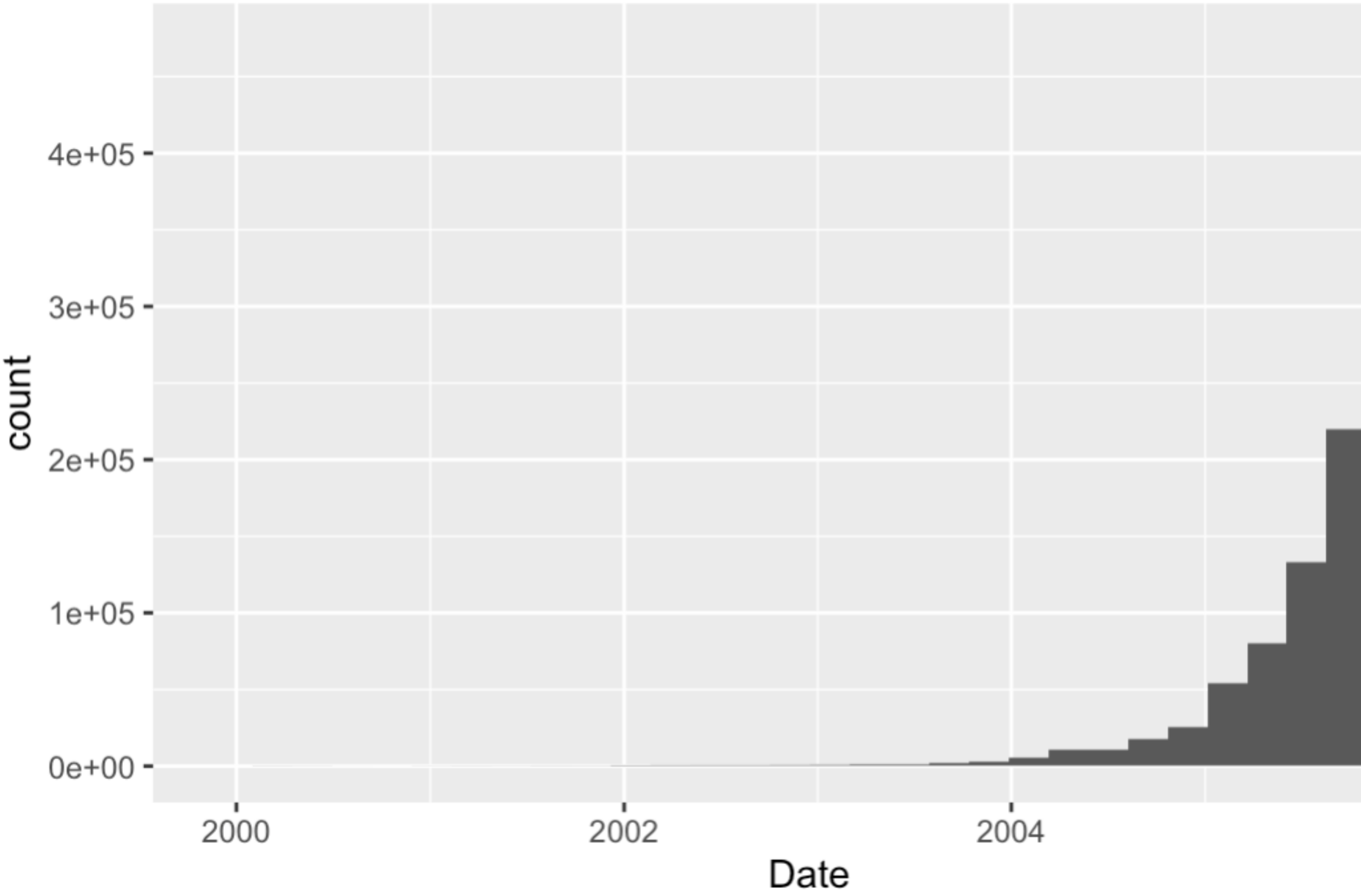
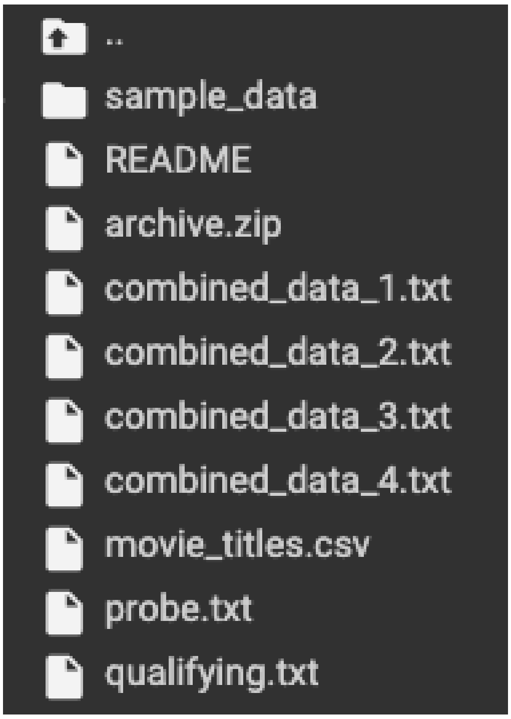


Figure 3.2: Distribution plot of star ratings over users and time.



```
..
sample_data
README
archive.zip
combined_data_1.txt
combined_data_2.txt
combined_data_3.txt
combined_data_4.txt
movie_titles.csv
probe.txt
qualifying.txt
```

Figure 3.3: If everything worked out well, you should have the above files available to browse and process.

b) perspectives on the solution,

c) and algorithmic rationale.