CS-119 (Big Data) **Instructor:** *Dr. Jitendra Singh*
Quiz 2: The Opioid Files **Student:** *Samay Pashine*
September 15, 2022

1. Find the column names in the Opioid dataset. The "normal" way would have been to gunzip the .gz file and run head -1 on the result, but you likely don't have enough disk space. Conveniently, zcat can read the file and write the unzipped contents into stdout, which you can pipe into head -1.

   Ans:

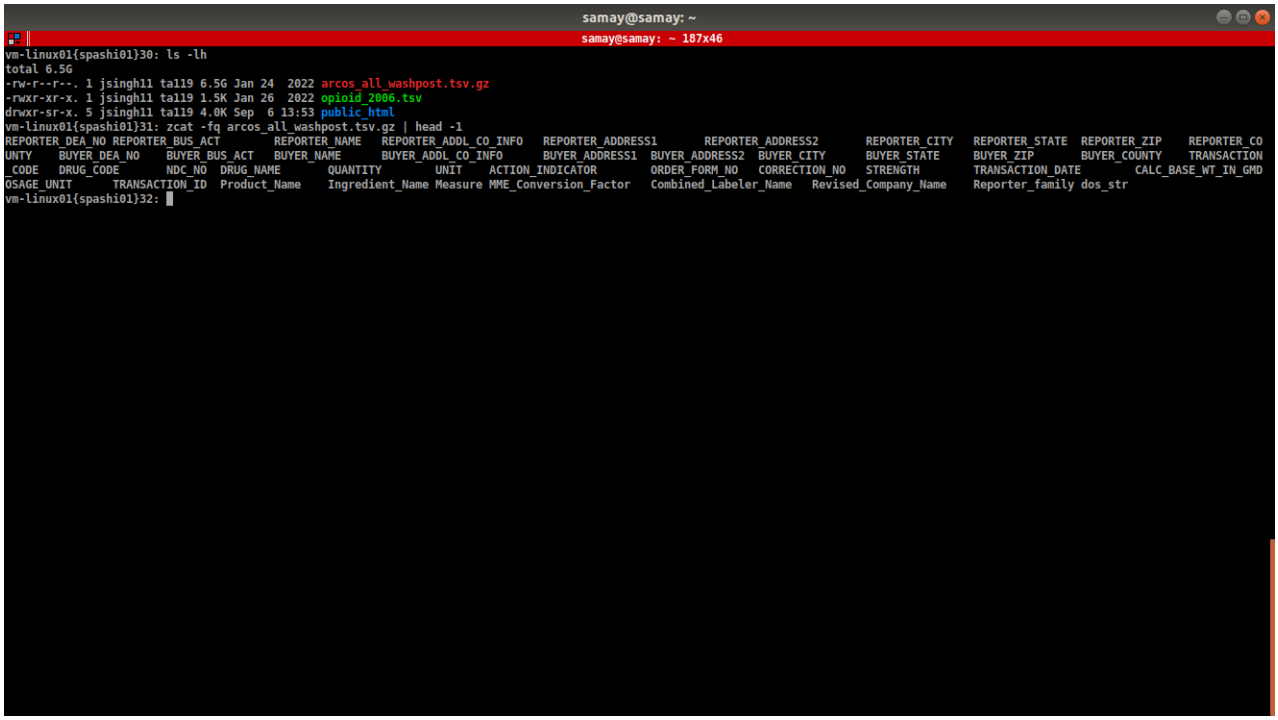

Figure 1: Columns printed in the terminal using zcat and head command

2. Find the number of rows in the Opioid dataset, using zcat output piped into wc. Make sure to take the header row into account.

   Ans:
   Total Number of rows (including the column names): 178598027
   Total Number of rows (excluding the column names): 178598026

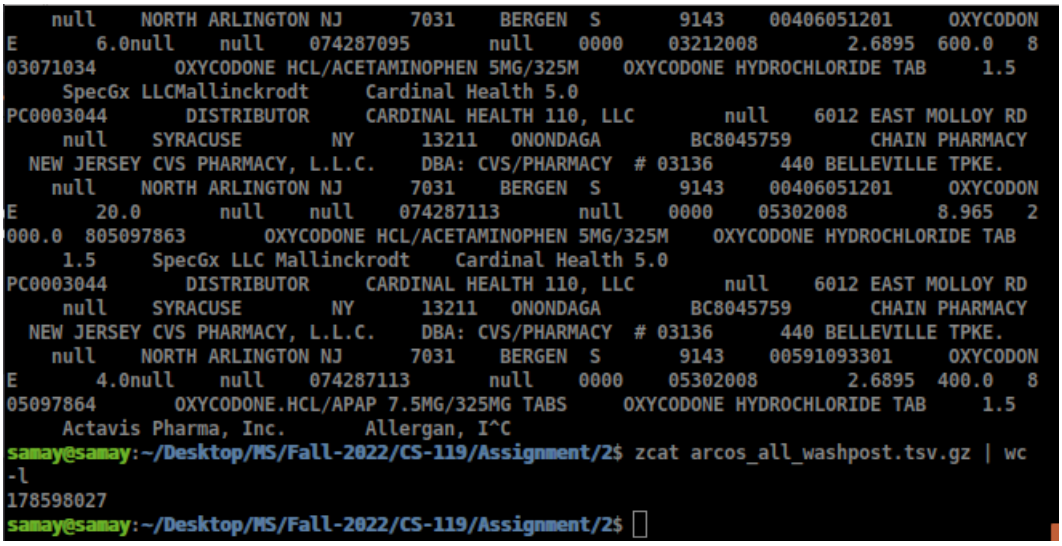

Figure 2: Total rows including the names of the columns printed using zcat and wc.

3. Find the number of rows for each year in the dataset. As above, we don't have enough space to unzip. So here's a potential strategy: Use the shuf command to extract, say, 5000 rows from the output of zcat. Find the proportion of rows for each year in this extract. You won't get an exact count but knowing the total number of rows, can you estimate the number of rows for each year?

   Ans:
   The answer to the question is shown in the screenshot below.

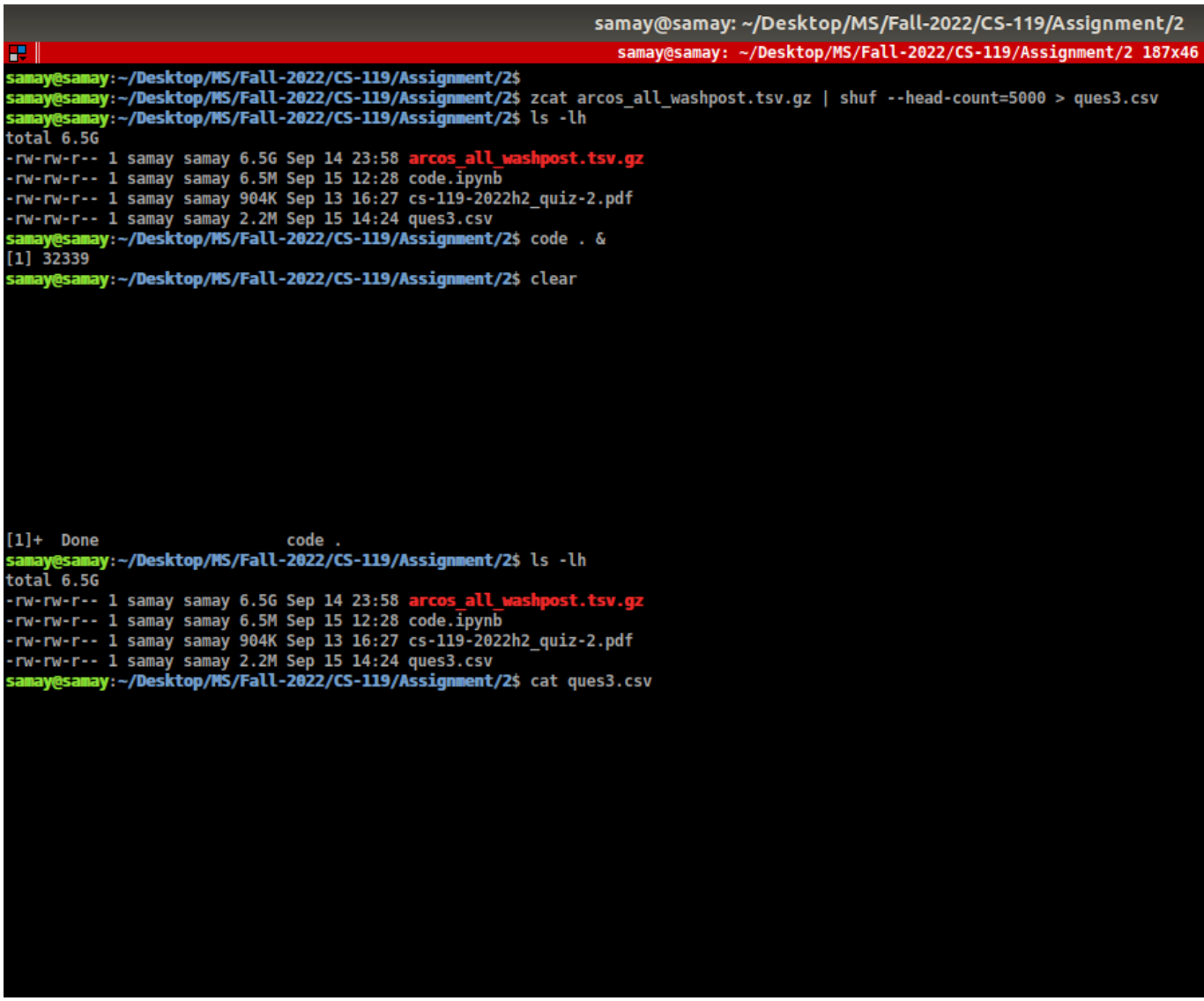

Figure 3: Using zcat and shuf command to extract 5000 rows from the whole dataset and saving it in a separate csv file.

```
import pandas as pd
df = pd.read_csv('5000_rows.csv', sep="\t", error_bad_lines=False)
```
✓ 0.1s

```
year_dict = {}
for i in range(df.shape[0]):
    if str(df.iloc[i, 30])[-4:] not in year_dict:
        year_dict[str(df.iloc[i, 30])[-4:]] = 0
    else:
        year_dict[str(df.iloc[i, 30])[-4:]] += 1
year_dict
```
✓ 0.3s

```
{'2006': 541,
 '2007': 637,
 '2012': 808,
 '2011': 867,
 '2010': 761,
 '2009': 704,
 '2008': 674}
```

```
part_rows = 5000
total_rows = 178598826
rows_2006 = int((float(year_dict['2006'])/part_rows)*total_rows)
rows_2007 = int((float(year_dict['2007'])/part_rows)*total_rows)
rows_2008 = int((float(year_dict['2008'])/part_rows)*total_rows)
rows_2009 = int((float(year_dict['2009'])/part_rows)*total_rows)
rows_2010 = int((float(year_dict['2010'])/part_rows)*total_rows)
rows_2011 = int((float(year_dict['2011'])/part_rows)*total_rows)
rows_2012 = int((float(year_dict['2012'])/part_rows)*total_rows)
print("[CALCULATED]. Number of rows for 2006 : ", rows_2006)
print("[CALCULATED]. Number of rows for 2007 : ", rows_2007)
print("[CALCULATED]. Number of rows for 2008 : ", rows_2008)
print("[CALCULATED]. Number of rows for 2009 : ", rows_2009)
print("[CALCULATED]. Number of rows for 2010 : ", rows_2010)
print("[CALCULATED]. Number of rows for 2011 : ", rows_2011)
print("[CALCULATED]. Number of rows for 2012 : ", rows_2012)
print("\n[CALCULATED]. Total number of rows : ", rows_2006 + rows_2007 + rows_2008 + rows_2009 + rows_2010 + rows_2011 + rows_2012)
print("[ACTUAL]. Total number of rows : ", total_rows)
```
✓ 0.4s

```
[CALCULATED]. Number of rows for 2006 :  19324306
[CALCULATED]. Number of rows for 2007 :  22753388
[CALCULATED]. Number of rows for 2008 :  24075013
[CALCULATED]. Number of rows for 2009 :  25146602
[CALCULATED]. Number of rows for 2010 :  27182619
[CALCULATED]. Number of rows for 2011 :  30968897
[CALCULATED]. Number of rows for 2012 :  28861441

[CALCULATED]. Total number of rows :  178312266
[ACTUAL]. Total number of rows :  178598826
```

Figure 4: Code to count the number of rows for each year using pandas library.

NOTE: The 5000 rows were saved in the file named ques3.csv but the file was renamed later to 5000_rows.csv