Biostat 203B Homework 1

Due Jan 24, 2025 @ 11:59PM

Emma Mo and 906542365

Table of contents

Display machine information for reproducibility:

```
sessionInfo()
```

```
R version 4.4.1 (2024-06-14)
Platform: x86_64-apple-darwin20
Running under: macOS Monterey 12.4

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.4-x86_64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.4-x86_64/Resources/lib/libRlapack.dylib;

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

time zone: Asia/Shanghai
tzcode source: internal

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

loaded via a namespace (and not attached):
 [1] compiler_4.4.1    fastmap_1.2.0     cli_3.6.3         tools_4.4.1
 [5] htmltools_0.5.8.1 rstudioapi_0.16.0 yaml_2.3.10       rmarkdown_2.29
 [9] knitr_1.49        jsonlite_1.8.9    xfun_0.50         digest_0.6.37
[13] rlang_1.1.4       evaluate_1.0.3
```

Q1. Git/GitHub

No handwritten homework reports are accepted for this course. We work with Git and GitHub. Efficient and abundant use of Git, e.g., frequent and well-documented commits, is an important criterion for grading your homework.

1. Apply for the Student Developer Pack at GitHub using your UCLA email. You'll get GitHub Pro account for free (unlimited public and private repositories).
2. Create a **private** repository `biostat-203b-2025-winter` and add `Hua-Zhou` and TA team (`Tomoki-Okuno` for Lec 1; `parsajamshidian` and `BowenZhang2001` for Lec 82) as your collaborators with write permission.
3. Top directories of the repository should be `hw1`, `hw2`, … Maintain two branches `main` and `develop`. The `develop` branch will be your main playground, the place where you develop solution (code) to homework problems and write up report. The `main` branch will be your presentation area. Submit your homework files (Quarto file `qmd`, `html` file converted by Quarto, all code and extra data sets to reproduce results) in the `main` branch.
4. After each homework due date, course reader and instructor will check out your `main` branch for grading. Tag each of your homework submissions with tag names `hw1`, `hw2`, … Tagging time will be used as your submission time. That means if you tag your `hw1` submission after deadline, penalty points will be deducted for late submission.
5. After this course, you can make this repository public and use it to demonstrate your skill sets on job market.

Solution: Done. https://github.com/Emma-Mo-0625/biostat-203b-2025-winter

Q2. Data ethics training

This exercise (and later in this course) uses the MIMIC-IV data v3.1, a freely accessible critical care database developed by the MIT Lab for Computational Physiology. Follow the instructions at https://mimic.mit.edu/docs/gettingstarted/ to (1) complete the CITI `Data or Specimens Only Research` course and (2) obtain the PhysioNet credential for using the MIMIC-IV data. Display the verification links to your completion report and completion certificate here. **You must complete Q2 before working on the remaining questions.** (Hint: The CITI training takes a few hours and the PhysioNet credentialing takes a couple days; do not leave it to the last minute.)

Solution: Here is the link to my completion report. Here is the link to my completion certificate.

Q3. Linux Shell Commands

1. Make the MIMIC-IV v3.1 data available at location `~/mimic`. The output of the `ls -l ~/mimic` command should be similar to the below (from my laptop).

```
# content of mimic folder
ls ~/mimic/mimic-iv-3.1
```

```
CHANGELOG.txt
LICENSE.txt
SHA256SUMS.txt
hosp
icu
```

Refer to the documentation https://physionet.org/content/mimiciv/3.1/ for details of data files. Do **not** put these data files into Git; they are big. Do **not** copy them into your directory. Do **not** decompress the gz data files. These create unnecessary big files and are not big-data-friendly practices. Read from the data folder `~/mimic` directly in following exercises.

Use Bash commands to answer following questions.

Solution I downloaded MIMIC IV v3.1 data to my computer and made it available at `~/mimic`.

2. Display the contents in the folders `hosp` and `icu` using Bash command `ls -l`. Why are these data files distributed as `.csv.gz` files instead of `.csv` (comma separated values) files? Read the page https://mimic.mit.edu/docs/iv/ to understand what's in each folder.

Solution Here is the content of the `hosp` folder:

```
ls -l ~/mimic/mimic-iv-3.1/hosp
```

```
total 12306248
-rw-r--r--  1 emma  staff    19928140 Jun 25  2024 admissions.csv.gz
-rw-r--r--  1 emma  staff      427554 Apr 13  2024 d_hcpcs.csv.gz
-rw-r--r--  1 emma  staff      876360 Apr 13  2024 d_icd_diagnoses.csv.gz
-rw-r--r--  1 emma  staff      589186 Apr 13  2024 d_icd_procedures.csv.gz
-rw-r--r--  1 emma  staff       13169 Oct  4 00:07 d_labitems.csv.gz
-rw-r--r--  1 emma  staff    33564802 Oct  4 00:07 diagnoses_icd.csv.gz
-rw-r--r--  1 emma  staff     9743908 Oct  4 00:07 drgcodes.csv.gz
-rw-r--r--  1 emma  staff   811305629 Apr 13  2024 emar.csv.gz
-rw-r--r--  1 emma  staff   748158322 Apr 13  2024 emar_detail.csv.gz
```

```
-rw-r--r--  1 emma  staff     2162335 Apr 13  2024 hcpcsevents.csv.gz
-rw-r--r--  1 emma  staff  2592909134 Oct  4 00:08 labevents.csv.gz
-rw-r--r--  1 emma  staff   117644075 Oct  4 00:08 microbiologyevents.csv.gz
-rw-r--r--  1 emma  staff    44069351 Oct  4 00:08 omr.csv.gz
-rw-r--r--  1 emma  staff     2835586 Apr 13  2024 patients.csv.gz
-rw-r--r--  1 emma  staff   525708076 Apr 13  2024 pharmacy.csv.gz
-rw-r--r--  1 emma  staff   666594177 Apr 13  2024 poe.csv.gz
-rw-r--r--  1 emma  staff    55267894 Apr 13  2024 poe_detail.csv.gz
-rw-r--r--  1 emma  staff   606298611 Apr 13  2024 prescriptions.csv.gz
-rw-r--r--  1 emma  staff     7777324 Apr 13  2024 procedures_icd.csv.gz
-rw-r--r--  1 emma  staff      127330 Apr 13  2024 provider.csv.gz
-rw-r--r--  1 emma  staff     8569241 Apr 13  2024 services.csv.gz
-rw-r--r--  1 emma  staff    46185771 Oct  4 00:08 transfers.csv.gz
```

3. Briefly describe what Bash commands `zcat`, `zless`, `zmore`, and `zgrep` do.

Solution `zcat`: decompresses a .gz file and outputs the content to standard output (stdout) without actually creating a decompressed file on disk. `zless`: view the contents of a compressed file by scrolling. `zmore`: view the contents of a compressed file page by page. `zgrep`: search for specific text patterns inside a .gz file using regular expressions

4. (Looping in Bash) What's the output of the following bash script?

```
for datafile in ~/mimic/mimic-iv-3.1/hosp/{a,l,pa}*.gz
do
  ls -l $datafile
done
```

Display the number of lines in each data file using a similar loop. (Hint: combine linux commands `zcat <` and `wc -l`.)

```
for datafile in ~/mimic/mimic-iv-3.1/hosp/{a,l,pa}*.gz
do
  echo "File: $datafile"
  zcat < "$datafile" | wc -l
done
```

```
File: /Users/emma/mimic/mimic-iv-3.1/hosp/admissions.csv.gz
  546029
File: /Users/emma/mimic/mimic-iv-3.1/hosp/labevents.csv.gz
 158374765
File: /Users/emma/mimic/mimic-iv-3.1/hosp/patients.csv.gz
  364628
```

5. Display the first few lines of `admissions.csv.gz`. How many rows are in this data file, excluding the header line? Each `hadm_id` identifies a hospitalization. How many hospitalizations are in this data file? How many unique patients (identified by `subject_id`) are in this data file? Do they match the number of patients listed in the `patients.csv.gz` file? (Hint: combine Linux commands `zcat <`, `head`/`tail`, `awk`, `sort`, `uniq`, `wc`, and so on.)

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz | head -5  # first five rows
```

```
subject_id,hadm_id,admittime,dischtime,deathtime,admission_type,admit_provider_id,admission_l
10000032,22595853,2180-05-06 22:23:00,2180-05-07 17:15:00,,URGENT,P49AFC,TRANSFER FROM HOSPI
10000032,22841357,2180-06-26 18:27:00,2180-06-27 18:49:00,,EW EMER.,P784FA,EMERGENCY ROOM,HO
10000032,25742920,2180-08-05 23:44:00,2180-08-07 17:50:00,,EW EMER.,P19UTS,EMERGENCY ROOM,HOS
10000032,29079034,2180-07-23 12:35:00,2180-07-25 17:55:00,,EW EMER.,P06OTX,EMERGENCY ROOM,HO
```

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz | tail -n +2 | head -5  # exclude header r
```

```
10000032,22595853,2180-05-06 22:23:00,2180-05-07 17:15:00,,URGENT,P49AFC,TRANSFER FROM HOSPI
10000032,22841357,2180-06-26 18:27:00,2180-06-27 18:49:00,,EW EMER.,P784FA,EMERGENCY ROOM,HO
10000032,25742920,2180-08-05 23:44:00,2180-08-07 17:50:00,,EW EMER.,P19UTS,EMERGENCY ROOM,HOS
10000032,29079034,2180-07-23 12:35:00,2180-07-25 17:55:00,,EW EMER.,P06OTX,EMERGENCY ROOM,HO
10000068,25022803,2160-03-03 23:16:00,2160-03-04 06:26:00,,EU OBSERVATION,P39NWO,EMERGENCY RO
```

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz | tail -n +2 | wc -l
```

```
546028
```

There are 546028 rows in this data file, excluding the header line. There are 546028 hospitalizations.

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz | tail -n +2 | awk -F',' '{print $1}' | so
```

```
223452
```

There are 223452 unique patients in the `admissions.csv.gz` file.

```
zcat < ~/mimic/mimic-iv-3.1/hosp/patients.csv.gz | tail -n +2 | awk -F',' '{print $1}' | sor
```

```
364627
```

There are 364627 unique patients in the `patients.csv.gz` file, which doesn't match.

6. What are the possible values taken by each of the variable `admission_type`, `admission_location`, `insurance`, and `ethnicity`? Also report the count for each unique value of these variables in decreasing order. (Hint: combine Linux commands `zcat`, `head`/`tail`, `awk`, `uniq -c`, `wc`, `sort`, and so on; skip the header line.)

Solution The possible values taken by `admission_type` is

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz |
awk -F, '{print $6}' | # prints the 6th column (admission_type)
tail -n +2 | #skip the header row & start from the second row
sort | #sort the values
uniq -c | #get unique values (eliminate consecutive duplicates) and count them
sort -nr #sort the count in descending order
```

```
177459 EW EMER.
119456 EU OBSERVATION
84437 OBSERVATION ADMIT
54929 URGENT
42898 SURGICAL SAME DAY ADMISSION
24551 DIRECT OBSERVATION
21973 DIRECT EMER.
13130 ELECTIVE
7195 AMBULATORY OBSERVATION
```

The possible values taken by `admission_location` is

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz |
awk -F, '{print $8}' | tail -n +2 | sort | uniq -c | sort -nr
```

```
244179 EMERGENCY ROOM
163228 PHYSICIAN REFERRAL
56227 TRANSFER FROM HOSPITAL
42365 WALK-IN/SELF REFERRAL
12965 CLINIC REFERRAL
```

```
8518 PROCEDURE SITE
6317 TRANSFER FROM SKILLED NURSING FACILITY
5837 INTERNAL TRANSFER TO OR FROM PSYCH
5734 PACU
 402 INFORMATION NOT AVAILABLE
 255 AMBULATORY SURGERY TRANSFER
   1
```

The possible values taken by `insurance` is

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz |
awk -F, '{print $10}' | tail -n +2 | sort | uniq -c | sort -nr
```

```
244576 Medicare
173399 Private
104229 Medicaid
14006 Other
9355
 463 No charge
```

The possible values taken by `ethnicity` is

```
zcat < ~/mimic/mimic-iv-3.1/hosp/admissions.csv.gz |
awk -F, '{print $13}' | tail -n +2 | sort | uniq -c | sort -nr
```

```
336538 WHITE
75482 BLACK/AFRICAN AMERICAN
19788 OTHER
13972 WHITE - OTHER EUROPEAN
13870 UNKNOWN
10903 HISPANIC/LATINO - PUERTO RICAN
8287 HISPANIC OR LATINO
7809 ASIAN
7644 ASIAN - CHINESE
6597 WHITE - RUSSIAN
6205 BLACK/CAPE VERDEAN
6070 HISPANIC/LATINO - DOMINICAN
3875 BLACK/CARIBBEAN ISLAND
3495 BLACK/AFRICAN
3478 UNABLE TO OBTAIN
2162 PATIENT DECLINED TO ANSWER
```

```
2082 PORTUGUESE
1973 ASIAN - SOUTH EAST ASIAN
1886 WHITE - EASTERN EUROPEAN
1858 HISPANIC/LATINO - GUATEMALAN
1661 ASIAN - ASIAN INDIAN
1526 WHITE - BRAZILIAN
1320 HISPANIC/LATINO - SALVADORAN
1247 AMERICAN INDIAN/ALASKA NATIVE
 920 HISPANIC/LATINO - COLUMBIAN
 883 HISPANIC/LATINO - MEXICAN
 774 SOUTH AMERICAN
 725 HISPANIC/LATINO - HONDURAN
 664 ASIAN - KOREAN
 641 HISPANIC/LATINO - CUBAN
 603 HISPANIC/LATINO - CENTRAL AMERICAN
 596 MULTIPLE RACE/ETHNICITY
 494 NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER
```

7. The `icustays.csv.gz` file contains all the ICU stays during the study period. How many ICU stays, identified by `stay_id`, are in this data file? How many unique patients, identified by `subject_id`, are in this data file?

```
zcat < ~/mimic/mimic-iv-3.1/icu/icustays.csv.gz | head -5
```

```
subject_id,hadm_id,stay_id,first_careunit,last_careunit,intime,outtime,los
10000032,29079034,39553978,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit (MI
10000690,25860671,37081114,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit (MI
10000980,26913865,39765666,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit (MI
10001217,24597018,37067082,Surgical Intensive Care Unit (SICU),Surgical Intensive Care Unit
```

```
zcat < ~/mimic/mimic-iv-3.1/icu/icustays.csv.gz | tail -n +2 | wc -l
```

```
  94458
```

Total number of ICU stays: 94458

```
zcat < ~/mimic/mimic-iv-3.1/icu/icustays.csv.gz | tail -n +2 | awk -F',' '{print $3}' | sort
```

```
  94458
```

Number of unique ICU stays: 94458

```
zcat < ~/mimic/mimic-iv-3.1/icu/icustays.csv.gz | tail -n +2 | awk -F',' '{print $1}' | sort
```

```
65366
```

Number of unique patients, identified by `subject_id`, is 65366

8. *To compress, or not to compress. That's the question.* Let's focus on the big data file `labevents.csv.gz`. Compare compressed gz file size to the uncompressed file size. Compare the run times of `zcat < ~/mimic/labevents.csv.gz | wc -l` versus `wc -l labevents.csv`. Discuss the trade off between storage and speed for big data files. (Hint: `gzip -dk < FILENAME.gz > ./FILENAME`. Remember to delete the large `labevents.csv` file after the exercise.)

```
ls -lh ~/mimic/mimic-iv-3.1/hosp/labevents.csv.gz # size of the compressed file
```

```
-rw-r--r--  1 emma  staff   2.4G Oct  4 00:08 /Users/emma/mimic/mimic-iv-3.1/hosp/labevents.
```

size of the compressed file: 2.4G

```
time zcat < ~/mimic/mimic-iv-3.1/hosp/labevents.csv.gz | wc -l # run time of compressed file
```

```
 158374765

real    0m35.844s
user    0m45.385s
sys 0m3.900s
```

```
gzip -dk < ~/mimic/mimic-iv-3.1/hosp/labevents.csv.gz > ~/mimic/mimic-iv-3.1/hosp/labevents.
```

```
time wc -l ~/mimic/mimic-iv-3.1/hosp/labevents.csv # run time of uncompressed file
```

```
 158374765 /Users/emma/mimic/mimic-iv-3.1/hosp/labevents.csv

real    0m17.161s
user    0m11.913s
sys 0m3.979s
```

```
ls -lh ~/mimic/mimic-iv-3.1/hosp/labevents.csv # size of the uncompressed file
```

```
-rw-r--r--  1 emma  staff    17G Jan 25 08:19 /Users/emma/mimic/mimic-iv-3.1/hosp/labevents.c
```

Size of the uncompressed file: 17G

```
rm ~/mimic/mimic-iv-3.1/hosp/labevents.csv # delete the large labevents.csv file
```

The compressed file takes smaller storage, which saves significant disk space. However, the compressed file takes longer to read and process, which may slow down the analysis. The uncompressed file is faster to read and process, but it takes up more disk space.

Q4. Who's popular in Price and Prejudice

1. You and your friend just have finished reading *Pride and Prejudice* by Jane Austen. Among the four main characters in the book, Elizabeth, Jane, Lydia, and Darcy, your friend thinks that Darcy was the most mentioned. You, however, are certain it was Elizabeth. Obtain the full text of the novel from http://www.gutenberg.org/cache/epub/42671/pg42671.txt and save to your local folder.

```
wget -nc http://www.gutenberg.org/cache/epub/42671/pg42671.txt
```

Explain what `wget -nc` does. Do **not** put this text file `pg42671.txt` in Git. Complete the following loop to tabulate the number of times each of the four characters is mentioned using Linux commands.

Solution `wget -nc` downloads the file only if it doesn't exist in the current directory.

```
for char in Elizabeth Jane Lydia Darcy
do
  echo $char:
  grep -o "$char" pg42671.txt | wc -l  # count the number of times each character is mentione
done
```

2. What's the difference between the following two commands?

```
echo 'hello, world' > test1.txt
```

and