Biostat 203B Homework 3

Due Feb 21 @ 11:59PM

Kiana Mohammadinik and 205928003

Table of contents

Display machine information for reproducibility:

```r
sessionInfo()
```

```
R version 4.3.0 (2023-04-21)
Platform: aarch64-apple-darwin20 (64-bit)
Running under: macOS 14.4.1

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
```

```
LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/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: America/Los_Angeles
tzcode source: internal

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

loaded via a namespace (and not attached):
 [1] compiler_4.3.0    fastmap_1.1.1     cli_3.6.3         tools_4.3.0
 [5] htmltools_0.5.8.1 rstudioapi_0.14   yaml_2.3.8        rmarkdown_2.29
 [9] knitr_1.45        jsonlite_1.8.8    xfun_0.50         digest_0.6.34
[13] rlang_1.1.4       evaluate_0.23
```

Load necessary libraries (you can add more as needed).

```r
library(arrow)
```

```
Warning: package 'arrow' was built under R version 4.3.3


Attaching package: 'arrow'

The following object is masked from 'package:utils':

    timestamp
```

```r
library(gtsummary)
```

```
Warning: package 'gtsummary' was built under R version 4.3.3
```

```r
library(memuse)
```

```
Warning: package 'memuse' was built under R version 4.3.3
```

```r
library(pryr)
```

```

Attaching package: 'pryr'

The following object is masked from 'package:gtsummary':

    where
```

```r
library(R.utils)
```

```
Warning: package 'R.utils' was built under R version 4.3.1

Loading required package: R.oo

Warning: package 'R.oo' was built under R version 4.3.1

Loading required package: R.methodsS3

R.methodsS3 v1.8.2 (2022-06-13 22:00:14 UTC) successfully loaded. See ?R.methodsS3 for help.

R.oo v1.26.0 (2024-01-24 05:12:50 UTC) successfully loaded. See ?R.oo for help.


Attaching package: 'R.oo'

The following object is masked from 'package:R.methodsS3':

    throw

The following objects are masked from 'package:methods':

    getClasses, getMethods

The following objects are masked from 'package:base':

    attach, detach, load, save
```

```
R.utils v2.12.3 (2023-11-18 01:00:02 UTC) successfully loaded. See ?R.utils for help.


Attaching package: 'R.utils'

The following object is masked from 'package:arrow':

    timestamp

The following object is masked from 'package:utils':

    timestamp

The following objects are masked from 'package:base':

    cat, commandArgs, getOption, isOpen, nullfile, parse, warnings
```

```r
library(tidyverse)
```

```
Warning: package 'ggplot2' was built under R version 4.3.1

Warning: package 'tidyr' was built under R version 4.3.1

Warning: package 'dplyr' was built under R version 4.3.1

Warning: package 'stringr' was built under R version 4.3.1

-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.2     v tidyr     1.3.1
v purrr     1.0.1
```

```
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x purrr::compose()      masks pryr::compose()
x lubridate::duration() masks arrow::duration()
x tidyr::extract()      masks R.utils::extract()
x dplyr::filter()       masks stats::filter()
x dplyr::lag()          masks stats::lag()
x purrr::partial()      masks pryr::partial()
x dplyr::where()        masks pryr::where(), gtsummary::where()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```
library(ggplot2)
library(dplyr)
library(lubridate)
library(stringr)
```

Display your machine memory.

```
memuse::Sys.meminfo()
```

```
Totalram:  16.000 GiB
Freeram:    4.693 GiB
```

In this exercise, we use tidyverse (ggplot2, dplyr, etc) to explore the MIMIC-IV data introduced in homework 1 and to build a cohort of ICU stays.

0.1 Q1. Visualizing patient trajectory

Visualizing a patient's encounters in a health care system is a common task in clinical data analysis. In this question, we will visualize a patient's ADT (admission-discharge-transfer) history and ICU vitals in the MIMIC-IV data.

0.1.1 Q1.1 ADT history

A patient's ADT history records the time of admission, discharge, and transfer in the hospital. This figure shows the ADT history of the patient with `subject_id` 10001217 in the MIMIC-IV data. The x-axis is the calendar time, and the y-axis is the type of event (ADT, lab, procedure). The color of the line segment represents the care unit. The size of the line segment represents whether the care unit is an ICU/CCU. The crosses represent lab events, and the shape of the dots represents the type of procedure. The title of the figure shows the patient's demographic information and the subtitle shows top 3 diagnoses.

Do a similar visualization for the patient with `subject_id` 10063848 using ggplot.

Hint: We need to pull information from data files `patients.csv.gz`, `admissions.csv.gz`, `transfers.csv.gz`, `labevents.csv.gz`, `procedures_icd.csv.gz`, `diagnoses_icd.csv.gz`, `d_icd_procedures.csv.gz`, and `d_icd_diagnoses.csv.gz`. For the big file `labevents.csv.gz`, use the Parquet format you generated in Homework 2. For reproducibility, make the Parquet folder `labevents_pq` available at the current working directory `hw3`, for example, by a symbolic link. Make your code reproducible.

```
patients <- read_csv("~/mimic/hosp/patients.csv.gz")
```

```
Rows: 364627 Columns: 6
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (2): gender, anchor_year_group
dbl  (3): subject_id, anchor_age, anchor_year
date (1): dod

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
admissions <- read_csv("~/mimic/hosp/admissions.csv.gz")
```

```
Rows: 546028 Columns: 16
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (8): admission_type, admit_provider_id, admission_location, discharge_l...
dbl  (3): subject_id, hadm_id, hospital_expire_flag
dttm (5): admittime, dischtime, deathtime, edregtime, edouttime

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
transfers <- read_csv("~/mimic/hosp/transfers.csv.gz")
```

```
Rows: 2413581 Columns: 7
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (2): eventtype, careunit
dbl  (3): subject_id, hadm_id, transfer_id
dttm (2): intime, outtime
```

```
i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
labevents <- read_parquet("labevents.parquet/part-0.parquet")
procedures <- read_csv("~/mimic/hosp/procedures_icd.csv.gz")
```

```
Rows: 859655 Columns: 6
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (1): icd_code
dbl  (4): subject_id, hadm_id, seq_num, icd_version
date (1): chartdate

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
diagnoses <- read_csv("~/mimic/hosp/diagnoses_icd.csv.gz")
```

```
Rows: 6364488 Columns: 5
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): icd_code
dbl (4): subject_id, hadm_id, seq_num, icd_version

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
d_icd_procedures <- read_csv("~/mimic/hosp/d_icd_procedures.csv.gz")
```

```
Rows: 86423 Columns: 3
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): icd_code, long_title
dbl (1): icd_version

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
d_icd_diagnoses <- read_csv("~/mimic/hosp/d_icd_diagnoses.csv.gz")
```

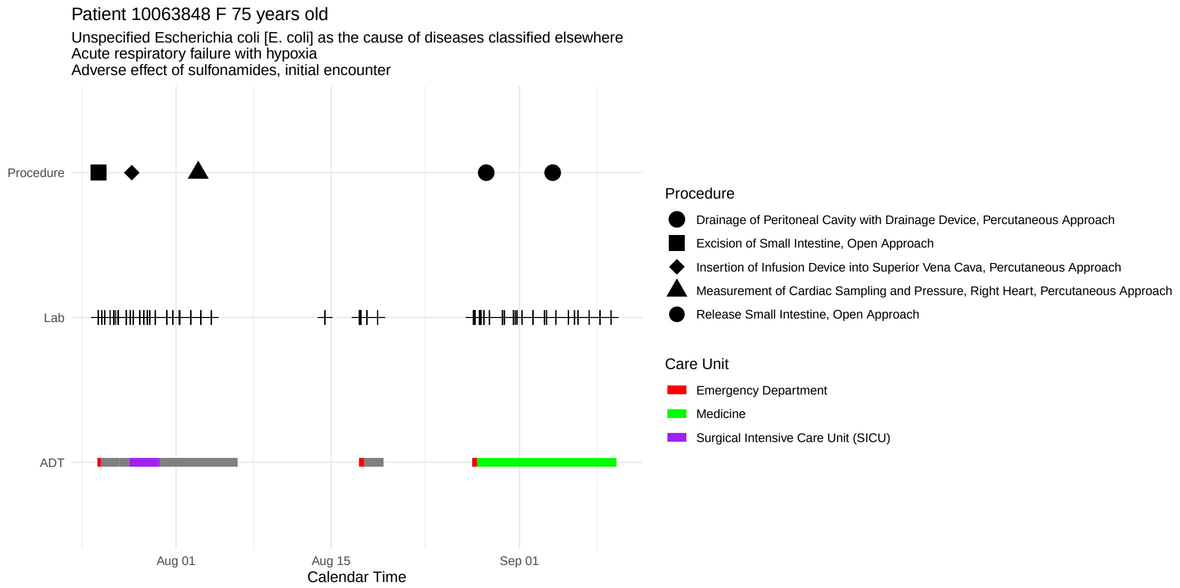
```
Rows: 112107 Columns: 3
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): icd_code, long_title
dbl (1): icd_version

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Filter data for patient 10063848
subject_id <- 10063848
patient_info <- patients %>% filter(subject_id == !!subject_id)
admissions_info <- admissions %>% filter(subject_id == !!subject_id)
transfers_info <- transfers %>% filter(subject_id == !!subject_id)
labevents_info <- labevents %>% filter(subject_id == !!subject_id)
procedures_info <- procedures %>% filter(subject_id == !!subject_id)
diagnoses_info <- diagnoses %>% filter(subject_id == !!subject_id)
```

0.2 plot

```r
# Standardize icd_code format before merging
diagnoses_info <- diagnoses_info %>%
  mutate(icd_code = str_pad(icd_code, width = 5, pad = "0"))

# Merge diagnoses with descriptions
diagnoses_info <- diagnoses_info %>%
  left_join(d_icd_diagnoses, by = c("icd_code", "icd_version"))

# Identify the correct long_title column for diagnoses
long_title_col <- grep("long_title", colnames(diagnoses_info), value = TRUE)

if ("long_title" %in% long_title_col) {
  diagnoses_info <- diagnoses_info %>% rename(diagnosis_name = long_title)
} else if ("long_title.x" %in% long_title_col) {
  diagnoses_info <- diagnoses_info %>% rename(diagnosis_name = long_title.x)
} else if ("long_title.y" %in% long_title_col) {
  diagnoses_info <- diagnoses_info %>% rename(diagnosis_name = long_title.y)
} else {
  stop("long_title column not found in diagnoses_info")
}

# Extract top 3 diagnoses, removing NA values
top_diagnoses <- diagnoses_info %>%
  filter(!is.na(diagnosis_name)) %>%
  count(diagnosis_name, sort = TRUE) %>%
  head(3) %>%
  pull(diagnosis_name)

# Only concatenate if there are valid diagnoses
top_diagnoses_text <- ifelse(length(top_diagnoses) > 0,
  paste(top_diagnoses, collapse = "\n"), "")

# Extract demographics, omitting NA values
patient_summary <- paste(
  "Patient",
  subject_id,
  ifelse(is.na(patient_info$gender), "", patient_info$gender),
  ifelse(is.na(patient_info$anchor_age), "",
  paste(patient_info$anchor_age, "years old"))
) %>% str_squish()

# Convert timestamps
transfers_info <- transfers_info %>%
  mutate(intime = as.POSIXct(intime, format="%Y-%m-%d %H:%M:%S"),
         outtime = as.POSIXct(outtime, format="%Y-%m-%d %H:%M:%S")) %>%
  filter(!is.na(outtime))  # Remove missing outtimes

labevents_info <- labevents_info %>%
  mutate(chartdate = as.POSIXct(charttime, format="%Y-%m-%d %H:%M:%S"))

# Join and resolve duplicate columns in procedures_info
procedures_info <- procedures_info %>%
  mutate(chartdate = as.POSIXct(chartdate, format="%Y-%m-%d %H:%M:%S")) %>%
  left_join(d_icd_procedures, by = c("icd_code", "icd_version"))

# Identify procedure name columns
procedure_name_cols <- grep("long_title", colnames(procedures_info),
  value = TRUE)
```

```r
# Ensure only one unique procedure_name remains
if (length(procedure_name_cols) > 1) {
  procedures_info <- procedures_info %>%
  select(-one_of(procedure_name_cols[-1])) %>%
  rename(procedure_name = procedure_name_cols[1])
} else if (length(procedure_name_cols) == 1) {
  procedures_info <- procedures_info %>%
  rename(procedure_name = procedure_name_cols[1])
} else {
  stop("procedure_name column not found in procedures_info")
}

# Remove rows where procedure_name is missing
procedures_info <- procedures_info %>% filter(!is.na(procedure_name))

# Care unit colors
care_unit_colors <- c("Emergency Department" = "red",
                      "Medicine" = "green",
                      "Neurology" = "cyan",
                      "Surgical Intensive Care Unit (SICU)" = "purple")

# Define procedure shapes
procedure_shapes <- setNames(seq(15, 15 +
  length(unique(procedures_info$procedure_name)) - 1),
  unique(procedures_info$procedure_name))

# Plot patient trajectory
plot <- ggplot() +
  geom_segment(data = transfers_info,
  aes(x = intime, xend = outtime,
  y = "ADT", yend = "ADT", color = careunit),
  linewidth = 3) +
  geom_point(data = labevents_info,
  aes(x = chartdate, y = "Lab"), shape = 3, size = 3) +
  geom_point(data = procedures_info,
  aes(x = chartdate, y = "Procedure", shape = procedure_name), size = 5) +
  scale_color_manual(values = care_unit_colors) +
  scale_shape_manual(values = procedure_shapes, drop = FALSE) +
  theme_minimal() +
  labs(title = patient_summary,
  subtitle = top_diagnoses_text,
  x = "Calendar Time",
  y = NULL,
  color = "Care Unit",
  shape = "Procedure")

print(plot)
```



0.2.1 Q1.2 ICU stays

ICU stays are a subset of ADT history. This figure shows the vitals of the patient `10001217` during ICU stays. The x-axis is the calendar time, and the y-axis is the value of the vital. The color of the line represents the type of vital. The facet grid shows the abbreviation of the vital and the stay ID.

Do a similar visualization for the patient `10063848`.

0.3 Q2. ICU stays

`icustays.csv.gz` (https://mimic.mit.edu/docs/iv/modules/icu/icustays/) contains data about Intensive Care Units (ICU) stays. The first 10 lines are

```
zcat < ~/mimic/icu/icustays.csv.gz | head
```

```
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
10001217,27703517,34592300,Surgical Intensive Care Unit (SICU),Surgical Intensive Care Unit
10001725,25563031,31205490,Medical/Surgical Intensive Care Unit (MICU/SICU),Medical/Surgical
10001843,26133978,39698942,Medical/Surgical Intensive Care Unit (MICU/SICU),Medical/Surgical
10001884,26184834,37510196,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit (MI
10002013,23581541,39060235,Cardiac Vascular Intensive Care Unit (CVICU),Cardiac Vascular Inte
```

0.3.1 Q2.1 Ingestion

Import `icustays.csv.gz` as a tibble `icustays_tble`.

0.3.2 Q2.2 Summary and visualization

How many unique `subject_id`? Can a `subject_id` have multiple ICU stays? Summarize the number of ICU stays per `subject_id` by graphs.

0.4 Q3. `admissions` data

Information of the patients admitted into hospital is available in `admissions.csv.gz`. See https://mimic.mit.edu/docs/iv/modules/hosp/admissions/ for details of each field in this file. The first 10 lines are

```
zcat < ~/mimic/hosp/admissions.csv.gz | head
```

```
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 HOSPIT
10000032,22841357,2180-06-26 18:27:00,2180-06-27 18:49:00,,EW EMER.,P784FA,EMERGENCY ROOM,HOM
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,HOM
10000068,25022803,2160-03-03 23:16:00,2160-03-04 06:26:00,,EU OBSERVATION,P39NWO,EMERGENCY RO
10000084,23052089,2160-11-21 01:56:00,2160-11-25 14:52:00,,EW EMER.,P42H7G,WALK-IN/SELF REFER
10000084,29888819,2160-12-28 05:11:00,2160-12-28 16:07:00,,EU OBSERVATION,P35NE4,PHYSICIAN RE
10000108,27250926,2163-09-27 23:17:00,2163-09-28 09:04:00,,EU OBSERVATION,P40JML,EMERGENCY RO
10000117,22927623,2181-11-15 02:05:00,2181-11-15 14:52:00,,EU OBSERVATION,P47EY8,EMERGENCY RO
```

0.4.1 Q3.1 Ingestion

Import `admissions.csv.gz` as a tibble `admissions_tble`.

0.4.2 Q3.2 Summary and visualization

Summarize the following information by graphics and explain any patterns you see.

- number of admissions per patient
- admission hour (anything unusual?)
- admission minute (anything unusual?)
- length of hospital stay (from admission to discharge) (anything unusual?)

According to the MIMIC-IV documentation,

> All dates in the database have been shifted to protect patient confidentiality. Dates will be internally consistent for the same patient, but randomly distributed in the future. Dates of birth which occur in the present time are not true dates of birth. Furthermore, dates of birth which occur before the year 1900 occur if the patient is older than 89. In these cases, the patient's age at their first admission has been fixed to 300.

0.5 Q4. `patients` data

Patient information is available in `patients.csv.gz`. See https://mimic.mit.edu/docs/iv/modules/hosp/patients/ for details of each field in this file. The first 10 lines are

```
zcat < ~/mimic/hosp/patients.csv.gz | head
```

```
subject_id,gender,anchor_age,anchor_year,anchor_year_group,dod
10000032,F,52,2180,2014 - 2016,2180-09-09
10000048,F,23,2126,2008 - 2010,
10000058,F,33,2168,2020 - 2022,
10000068,F,19,2160,2008 - 2010,
10000084,M,72,2160,2017 - 2019,2161-02-13
10000102,F,27,2136,2008 - 2010,
10000108,M,25,2163,2014 - 2016,
10000115,M,24,2154,2017 - 2019,
10000117,F,48,2174,2008 - 2010,
```

0.5.1 Q4.1 Ingestion

Import `patients.csv.gz` (https://mimic.mit.edu/docs/iv/modules/hosp/patients/) as a tibble `patients_tble`.

0.5.2 Q4.2 Summary and visualization

Summarize variables `gender` and `anchor_age` by graphics, and explain any patterns you see.

0.6 Q5. Lab results

`labevents.csv.gz` (https://mimic.mit.edu/docs/iv/modules/hosp/labevents/) contains all laboratory measurements for patients. The first 10 lines are

```
zcat < ~/mimic/hosp/labevents.csv.gz | head
```

```
labevent_id,subject_id,hadm_id,specimen_id,itemid,order_provider_id,charttime,storetime,value
1,10000032,,2704548,50931,P69FQC,2180-03-23 11:51:00,2180-03-23 15:56:00,___,95,mg/dL,70,100
2,10000032,,36092842,51071,P69FQC,2180-03-23 11:51:00,2180-03-23 16:00:00,NEG,,,,,,ROUTINE,
3,10000032,,36092842,51074,P69FQC,2180-03-23 11:51:00,2180-03-23 16:00:00,NEG,,,,,,ROUTINE,
4,10000032,,36092842,51075,P69FQC,2180-03-23 11:51:00,2180-03-23 16:00:00,NEG,,,,,,ROUTINE,"
5,10000032,,36092842,51079,P69FQC,2180-03-23 11:51:00,2180-03-23 16:00:00,NEG,,,,,,ROUTINE,
6,10000032,,36092842,51087,P69FQC,2180-03-23 11:51:00,,,,,,,,ROUTINE,RANDOM.
7,10000032,,36092842,51089,P69FQC,2180-03-23 11:51:00,2180-03-23 16:15:00,,,,,,,,ROUTINE,PRESU
8,10000032,,36092842,51090,P69FQC,2180-03-23 11:51:00,2180-03-23 16:00:00,NEG,,,,,,ROUTINE,MI
9,10000032,,36092842,51092,P69FQC,2180-03-23 11:51:00,2180-03-23 16:00:00,NEG,,,,,,ROUTINE,"
```

`d_labitems.csv.gz` (https://mimic.mit.edu/docs/iv/modules/hosp/d_labitems/) is the dictionary of lab measurements.

```
zcat < ~/mimic/hosp/d_labitems.csv.gz | head
```

```
itemid,label,fluid,category
50801,Alveolar-arterial Gradient,Blood,Blood Gas
50802,Base Excess,Blood,Blood Gas
50803,"Calculated Bicarbonate, Whole Blood",Blood,Blood Gas
50804,Calculated Total CO2,Blood,Blood Gas
50805,Carboxyhemoglobin,Blood,Blood Gas
50806,"Chloride, Whole Blood",Blood,Blood Gas
50808,Free Calcium,Blood,Blood Gas
```

```
50809,Glucose,Blood,Blood Gas
50810,"Hematocrit, Calculated",Blood,Blood Gas
```

We are interested in the lab measurements of creatinine (50912), potassium (50971), sodium (50983), chloride (50902), bicarbonate (50882), hematocrit (51221), white blood cell count (51301), and glucose (50931). Retrieve a subset of `labevents.csv.gz` that only containing these items for the patients in `icustays_tble`. Further restrict to the last available measurement (by `storetime`) before the ICU stay. The final `labevents_tble` should have one row per ICU stay and columns for each lab measurement.

Hint: Use the Parquet format you generated in Homework 2. For reproducibility, make `labevents_pq` folder available at the current working directory `hw3`, for example, by a symbolic link.

0.7 Q6. Vitals from charted events

`chartevents.csv.gz` (https://mimic.mit.edu/docs/iv/modules/icu/chartevents/) contains all the charted data available for a patient. During their ICU stay, the primary repository of a patient's information is their electronic chart. The `itemid` variable indicates a single measurement type in the database. The `value` variable is the value measured for `itemid`. The first 10 lines of `chartevents.csv.gz` are

```
zcat < ~/mimic/icu/chartevents.csv.gz | head
```

```
subject_id,hadm_id,stay_id,caregiver_id,charttime,storetime,itemid,value,valuenum,valueuom,wa
10000032,29079034,39553978,18704,2180-07-23 12:36:00,2180-07-23 14:45:00,226512,39.4,39.4,kg
10000032,29079034,39553978,18704,2180-07-23 12:36:00,2180-07-23 14:45:00,226707,60,60,Inch,0
10000032,29079034,39553978,18704,2180-07-23 12:36:00,2180-07-23 14:45:00,226730,152,152,cm,0
10000032,29079034,39553978,18704,2180-07-23 14:00:00,2180-07-23 14:18:00,220048,SR (Sinus Rhy
10000032,29079034,39553978,18704,2180-07-23 14:00:00,2180-07-23 14:18:00,224642,Oral,,,0
10000032,29079034,39553978,18704,2180-07-23 14:00:00,2180-07-23 14:18:00,224650,None,,,0
10000032,29079034,39553978,18704,2180-07-23 14:00:00,2180-07-23 14:20:00,223761,98.7,98.7,°F
10000032,29079034,39553978,18704,2180-07-23 14:11:00,2180-07-23 14:17:00,220179,84,84,mmHg,0
10000032,29079034,39553978,18704,2180-07-23 14:11:00,2180-07-23 14:17:00,220180,48,48,mmHg,0
```

`d_items.csv.gz` (https://mimic.mit.edu/docs/iv/modules/icu/d_items/) is the dictionary for the `itemid` in `chartevents.csv.gz`.

```
zcat < ~/mimic/icu/d_items.csv.gz | head
```

```
itemid,label,abbreviation,linksto,category,unitname,param_type,lownormalvalue,highnormalvalu
220001,Problem List,Problem List,chartevents,General,,Text,,
220003,ICU Admission date,ICU Admission date,datetimeevents,ADT,,Date and time,,
220045,Heart Rate,HR,chartevents,Routine Vital Signs,bpm,Numeric,,
220046,Heart rate Alarm - High,HR Alarm - High,chartevents,Alarms,bpm,Numeric,,
220047,Heart Rate Alarm - Low,HR Alarm - Low,chartevents,Alarms,bpm,Numeric,,
220048,Heart Rhythm,Heart Rhythm,chartevents,Routine Vital Signs,,Text,,
220050,Arterial Blood Pressure systolic,ABPs,chartevents,Routine Vital Signs,mmHg,Numeric,90
220051,Arterial Blood Pressure diastolic,ABPd,chartevents,Routine Vital Signs,mmHg,Numeric,6
220052,Arterial Blood Pressure mean,ABPm,chartevents,Routine Vital Signs,mmHg,Numeric,,
```

We are interested in the vitals for ICU patients: heart rate (220045), systolic non-invasive blood pressure (220179), diastolic non-invasive blood pressure (220180), body temperature in Fahrenheit (223761), and respiratory rate (220210). Retrieve a subset of `chartevents.csv.gz` only containing these items for the patients in `icustays_tble`. Further restrict to the first vital measurement within the ICU stay. The final `chartevents_tble` should have one row per ICU stay and columns for each vital measurement.

Hint: Use the Parquet format you generated in Homework 2. For reproducibility, make `chartevents_pq` folder available at the current working directory, for example, by a symbolic link.

0.8 Q7. Putting things together

Let us create a tibble `mimic_icu_cohort` for all ICU stays, where rows are all ICU stays of adults (age at `intime` >= 18) and columns contain at least following variables

- all variables in `icustays_tble`

- all variables in `admissions_tble`

- all variables in `patients_tble`
- the last lab measurements before the ICU stay in `labevents_tble`
- the first vital measurements during the ICU stay in `chartevents_tble`

The final `mimic_icu_cohort` should have one row per ICU stay and columns for each variable.

0.9 Q8. Exploratory data analysis (EDA)

Summarize the following information about the ICU stay cohort `mimic_icu_cohort` using appropriate numerics or graphs:

- Length of ICU stay `los` vs demographic variables (race, insurance, marital_status, gender, age at intime)
- Length of ICU stay `los` vs the last available lab measurements before ICU stay
- Length of ICU stay `los` vs the first vital measurements within the ICU stay
- Length of ICU stay `los` vs first ICU unit