Biostat 203B Homework 4

Due Mar 9 @ 11:59PM

Wenjing Zhou and 806542441

Display machine information:

```
sessionInfo()
```

```
R version 4.4.3 (2025-02-28)
Platform: x86_64-apple-darwin20
Running under: macOS Sequoia 15.3.1

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;  LAPACK version 3.12.0

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_4.4.3    fastmap_1.2.0     cli_3.6.4         tools_4.4.3
 [5] htmltools_0.5.8.1 rstudioapi_0.17.1 yaml_2.3.10       rmarkdown_2.29
 [9] knitr_1.49        jsonlite_1.8.9    xfun_0.48         digest_0.6.37
[13] rlang_1.1.5       evaluate_1.0.1
```

Display my machine memory.

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

```
Totalram:   16.000 GiB
Freeram:   551.477 MiB
```

Load database libraries and the tidyverse frontend:

```
library(bigrquery)
library(dbplyr)
library(DBI)
library(gt)
library(gtsummary)
library(tidyverse)
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0
──
✔ dplyr     1.1.4     ✔ readr     2.1.5
✔ forcats   1.0.0     ✔ stringr   1.5.1
✔ ggplot2   3.5.1     ✔ tibble    3.2.1
✔ lubridate 1.9.3     ✔ tidyr     1.3.1
✔ purrr     1.0.4
── Conflicts ────────────────────────────────────────── tidyverse_conflicts()
──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::ident()  masks dbplyr::ident()
✖ dplyr::lag()    masks stats::lag()
✖ dplyr::sql()    masks dbplyr::sql()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

Q1. Compile the ICU cohort in HW3 from the Google BigQuery database

Below is an outline of steps. In this homework, we exclusively work with the BigQuery database and should not use any MIMIC data files stored on our local computer. Transform data as much as possible in BigQuery database and `collect()` the tibble **only at the end of Q1.7**.

Q1.1 Connect to BigQuery

Authenticate with BigQuery using the service account token. Please place the service account token (shared via BruinLearn) in the working directory (same folder as your qmd file). Do **not** ever add this token to your Git repository. If you do so, you will lose 50 points.

```
# path to the service account token
satoken <- "biostat-203b-2025-winter-4e58ec6e5579.json"
# BigQuery authentication using service account
bq_auth(path = satoken)
```

Connect to BigQuery database `mimiciv_3_1` in GCP (Google Cloud Platform), using the project billing account `biostat-203b-2025-winter`.

```
# connect to the BigQuery database `biostat-203b-2025-mimiciv_3_1`
con_bq <- dbConnect(
```

```
    bigrquery::bigquery(),
    project = "biostat-203b-2025-winter",
    dataset = "mimiciv_3_1",
    billing = "biostat-203b-2025-winter"
)
con_bq
```

```
<BigQueryConnection>
  Dataset: biostat-203b-2025-winter.mimiciv_3_1
  Billing: biostat-203b-2025-winter
```

List all tables in the `mimiciv_3_1` database.

```
dbListTables(con_bq)
```

```
 [1] "admissions"         "caregiver"          "chartevents"
 [4] "d_hcpcs"            "d_icd_diagnoses"    "d_icd_procedures"
 [7] "d_items"            "d_labitems"         "datetimeevents"
[10] "diagnoses_icd"      "drgcodes"           "emar"
[13] "emar_detail"        "hcpcsevents"        "icustays"
[16] "ingredientevents"   "inputevents"        "labevents"
[19] "microbiologyevents" "omr"                "outputevents"
[22] "patients"           "pharmacy"           "poe"
[25] "poe_detail"         "prescriptions"      "procedureevents"
[28] "procedures_icd"     "provider"           "services"
[31] "transfers"
```

Q1.2 `icustays` data

Connect to the `icustays` table.

```
# full ICU stays table
icustays_tble <- tbl(con_bq, "icustays") |>
  arrange(subject_id, hadm_id, stay_id) |>
  # show_query() |>
  print(width = Inf)
```

```
# Source:     SQL [?? x 8]
# Database:   BigQueryConnection
# Ordered by: subject_id, hadm_id, stay_id
   subject_id  hadm_id  stay_id first_careunit
        <int>    <int>    <int> <chr>
 1   10000032 29079034 39553978 Medical Intensive Care Unit (MICU)
 2   10000690 25860671 37081114 Medical Intensive Care Unit (MICU)
 3   10000980 26913865 39765666 Medical Intensive Care Unit (MICU)
```

```
 4   10001217 24597018 37067082 Surgical Intensive Care Unit (SICU)
 5   10001217 27703517 34592300 Surgical Intensive Care Unit (SICU)
 6   10001725 25563031 31205490 Medical/Surgical Intensive Care Unit (MICU/SICU)
 7   10001843 26133978 39698942 Medical/Surgical Intensive Care Unit (MICU/SICU)
 8   10001884 26184834 37510196 Medical Intensive Care Unit (MICU)
 9   10002013 23581541 39060235 Cardiac Vascular Intensive Care Unit (CVICU)
10   10002114 27793700 34672098 Coronary Care Unit (CCU)
   last_careunit                                       intime
   <chr>                                               <dttm>
 1 Medical Intensive Care Unit (MICU)                  2180-07-23 14:00:00
 2 Medical Intensive Care Unit (MICU)                  2150-11-02 19:37:00
 3 Medical Intensive Care Unit (MICU)                  2189-06-27 08:42:00
 4 Surgical Intensive Care Unit (SICU)                 2157-11-20 19:18:02
 5 Surgical Intensive Care Unit (SICU)                 2157-12-19 15:42:24
 6 Medical/Surgical Intensive Care Unit (MICU/SICU)    2110-04-11 15:52:22
 7 Medical/Surgical Intensive Care Unit (MICU/SICU)    2134-12-05 18:50:03
 8 Medical Intensive Care Unit (MICU)                  2131-01-11 04:20:05
 9 Cardiac Vascular Intensive Care Unit (CVICU)        2160-05-18 10:00:53
10 Coronary Care Unit (CCU)                            2162-02-17 23:30:00
   outtime              los
   <dttm>             <dbl>
 1 2180-07-23 23:50:47 0.410
 2 2150-11-06 17:03:17 3.89
 3 2189-06-27 20:38:27 0.498
 4 2157-11-21 22:08:00 1.12
 5 2157-12-20 14:27:41 0.948
 6 2110-04-12 23:59:56 1.34
 7 2134-12-06 14:38:26 0.825
 8 2131-01-20 08:27:30 9.17
 9 2160-05-19 17:33:33 1.31
10 2162-02-20 21:16:27 2.91
# ℹ more rows
```

Q1.3 `admissions` data

Connect to the `admissions` table.

```{r}
# # TODO
# admissions_tble <-

# full admissions table
admissions_tble <- tbl(con_bq, "admissions") |>
  arrange(subject_id, hadm_id) |>
  # show_query() |>
  print(width = Inf)
```

```
# Source:     SQL [?? x 16]
# Database:   BigQueryConnection
# Ordered by: subject_id, hadm_id
   subject_id  hadm_id admittime           dischtime           deathtime
        <int>    <int> <dttm>              <dttm>              <dttm>
 1   10000032 22595853 2180-05-06 22:23:00 2180-05-07 17:15:00 NA
 2   10000032 22841357 2180-06-26 18:27:00 2180-06-27 18:49:00 NA
 3   10000032 25742920 2180-08-05 23:44:00 2180-08-07 17:50:00 NA
 4   10000032 29079034 2180-07-23 12:35:00 2180-07-25 17:55:00 NA
 5   10000068 25022803 2160-03-03 23:16:00 2160-03-04 06:26:00 NA
 6   10000084 23052089 2160-11-21 01:56:00 2160-11-25 14:52:00 NA
 7   10000084 29888819 2160-12-28 05:11:00 2160-12-28 16:07:00 NA
 8   10000108 27250926 2163-09-27 23:17:00 2163-09-28 09:04:00 NA
 9   10000117 22927623 2181-11-15 02:05:00 2181-11-15 14:52:00 NA
10   10000117 27988844 2183-09-18 18:10:00 2183-09-21 16:30:00 NA
   admission_type    admit_provider_id admission_location    discharge_location
   <chr>             <chr>             <chr>                 <chr>
 1 URGENT            P49AFC            TRANSFER FROM HOSPITAL HOME
 2 EW EMER.          P784FA            EMERGENCY ROOM        HOME
 3 EW EMER.          P19UTS            EMERGENCY ROOM        HOSPICE
 4 EW EMER.          P06OTX            EMERGENCY ROOM        HOME
 5 EU OBSERVATION    P39NWO            EMERGENCY ROOM        <NA>
 6 EW EMER.          P42H7G            WALK-IN/SELF REFERRAL HOME HEALTH CARE
 7 EU OBSERVATION    P35NE4            PHYSICIAN REFERRAL    <NA>
 8 EU OBSERVATION    P40JML            EMERGENCY ROOM        <NA>
 9 EU OBSERVATION    P47EY8            EMERGENCY ROOM        <NA>
10 OBSERVATION ADMIT P13ACE            WALK-IN/SELF REFERRAL HOME HEALTH CARE
   insurance language marital_status race  edregtime
   <chr>     <chr>    <chr>          <chr> <dttm>
 1 Medicaid  English  WIDOWED        WHITE 2180-05-06 19:17:00
 2 Medicaid  English  WIDOWED        WHITE 2180-06-26 15:54:00
 3 Medicaid  English  WIDOWED        WHITE 2180-08-05 20:58:00
 4 Medicaid  English  WIDOWED        WHITE 2180-07-23 05:54:00
 5 <NA>      English  SINGLE         WHITE 2160-03-03 21:55:00
 6 Medicare  English  MARRIED        WHITE 2160-11-20 20:36:00
 7 Medicare  English  MARRIED        WHITE 2160-12-27 18:32:00
 8 <NA>      English  SINGLE         WHITE 2163-09-27 16:18:00
 9 Medicaid  English  DIVORCED       WHITE 2181-11-14 21:51:00
10 Medicaid  English  DIVORCED       WHITE 2183-09-18 08:41:00
   edouttime           hospital_expire_flag
   <dttm>                             <int>
 1 2180-05-06 23:30:00                    0
 2 2180-06-26 21:31:00                    0
 3 2180-08-06 01:44:00                    0
 4 2180-07-23 14:00:00                    0
 5 2160-03-04 06:26:00                    0
 6 2160-11-21 03:20:00                    0
 7 2160-12-28 16:07:00                    0
```

```
 8 2163-09-28 09:04:00                    0
 9 2181-11-15 09:57:00                    0
10 2183-09-18 20:20:00                    0
# ℹ more rows
```

Q1.4 `patients` data

Connect to the `patients` table.

```
# # TODO
patients_tble <- tbl(con_bq, "patients") |>
  arrange(subject_id) |>
  # show_query() |>
  print(width = Inf)
```

```
# Source:     SQL [?? x 6]
# Database:   BigQueryConnection
# Ordered by: subject_id
   subject_id gender anchor_age anchor_year anchor_year_group dod
        <int> <chr>       <int>       <int> <chr>             <date>
 1   10000032 F              52        2180 2014 - 2016       2180-09-09
 2   10000048 F              23        2126 2008 - 2010       NA
 3   10000058 F              33        2168 2020 - 2022       NA
 4   10000068 F              19        2160 2008 - 2010       NA
 5   10000084 M              72        2160 2017 - 2019       2161-02-13
 6   10000102 F              27        2136 2008 - 2010       NA
 7   10000108 M              25        2163 2014 - 2016       NA
 8   10000115 M              24        2154 2017 - 2019       NA
 9   10000117 F              48        2174 2008 - 2010       NA
10   10000161 M              60        2163 2020 - 2022       NA
# ℹ more rows
```

Q1.5 `labevents` data

Connect to the `labevents` table and retrieve a subset that only contain subjects who appear in `icustays_tble` and the lab items listed in HW3. Only keep the last lab measurements (by `storetime`) before the ICU stay and pivot lab items to become variables/columns. Write all steps in *one* chain of pipes.

```
column_order    <-    c("bicarbonate",    "chloride",    "creatinine","glucose",
"potassium","sodium",           "hematocrit", "white blood cells")

labevents_tble <- tbl(con_bq, "labevents") |>
  inner_join(tbl(con_bq, "icustays") |>
      select(subject_id, stay_id, intime),
      by = c("subject_id"),
      copy = TRUE
```

```
  ) |>
  filter(itemid %in% c(50912, 50971, 50983, 50902,
                       50882, 51221, 50931, 51301)) |>
  mutate(
    storetime = as.POSIXct(storetime),
    intime = as.POSIXct(intime)
  ) |>
  filter(storetime < intime) |>
  group_by(subject_id,itemid) |>
  slice_max(order_by = storetime, n = 1, with_ties = FALSE) |>
  ungroup() |>
  select(subject_id, stay_id, itemid, valuenum) |>
  left_join(tbl(con_bq, "d_labitems") |>
              select(itemid, label), by = c("itemid" = "itemid")) |>
  select(-itemid) |>
  pivot_wider(names_from = label, values_from = valuenum) |>
  arrange(subject_id, stay_id) |>
  rename_with(tolower) |>
  select(subject_id, stay_id, all_of(column_order)) |>
  print(width = Inf)
```

```
# Source:     SQL [?? x 10]
# Database:   BigQueryConnection
# Ordered by: subject_id, stay_id
   subject_id  stay_id bicarbonate chloride creatinine glucose potassium sodium
        <int>    <int>       <dbl>    <dbl>      <dbl>   <dbl>     <dbl>  <dbl>
 1   10000032 39553978          25       95        0.7     102       6.7    126
 2   10000690 37081114          26      100        1        85       4.8    137
 3   10000980 39765666          21      109        2.3      89       3.9    144
 4   10001217 34592300          30      104        0.5      87       4.1    142
 5   10001725 31205490          NA       98       NA        NA       4.1    139
 6   10001843 39698942          28       97        1.3     131       3.9    138
 7   10001884 37510196          30       88        1.1     141       4.5    130
 8   10002013 39060235          24      102        0.9     288       3.5    137
 9   10002114 34672098          18       NA        3.1      95       6.5    125
10   10002155 32358465          26       85        1.4     133       5.7    120
   hematocrit `white blood cells`
        <dbl>               <dbl>
 1       41.1                 6.9
 2       36.1                 7.1
 3       27.3                 5.3
 4       37.4                 5.4
 5       NA                  NA
 6       31.4                10.4
 7       39.7                12.2
 8       34.9                 7.2
 9       34.3                16.8
```

```
10       22.4                 9.8
# i more rows
```

Q1.6 `chartevents` data

Connect to `chartevents` table and retrieve a subset that only contain subjects who appear in `icustays_tble` and the chart events listed in HW3. Only keep the first chart events (by `storetime`) during ICU stay and pivot chart events to become variables/columns. Write all steps in *one* chain of pipes. Similary to HW3, if a vital has multiple measurements at the first `storetime`, average them.

```
column_order <- c("Heart Rate", "Non Invasive Blood Pressure systolic",
                  "Non Invasive Blood Pressure diastolic", "Respiratory Rate",
                  "Temperature Fahrenheit")

chartevents_tble <- tbl(con_bq, "chartevents") |>
    select(subject_id, itemid, storetime, valuenum) |>
  inner_join(tbl(con_bq, "icustays") |>
      select(subject_id, stay_id, intime, outtime),
      by = c("subject_id"),
      copy = TRUE
  ) |>
  filter(itemid %in% c(
    220045,
    220179,
    220180,
    223761,
    220210)) |>
  filter(storetime > intime & storetime < outtime) |>
  # average multiple measurements at the same time
  group_by(subject_id, stay_id, itemid, storetime) |>
  summarise(valuenum = round(mean(valuenum, na.rm = TRUE), 2), .groups = "drop")
|>
  group_by(subject_id, stay_id, itemid) |>
  slice_max(order_by = storetime, n = 1, with_ties = FALSE) |>
  select(-storetime) |>
  ungroup() |>
  # select(subject_id, stay_id, itemid, valuenum) |>
  left_join(tbl(con_bq, "d_items") |>
              select(itemid, label), by = c("itemid" = "itemid")) |>
  select(-itemid) |>
  pivot_wider(names_from = label, values_from = valuenum) |>
  arrange(subject_id, stay_id) |>
  relocate(subject_id) |>
  select(subject_id, stay_id, all_of(column_order)) |>
  rename_with(tolower) |>
  print(width = Inf)
```

```
# Source:     SQL [?? x 7]
# Database:   BigQueryConnection
# Ordered by: subject_id, stay_id
   subject_id  stay_id `heart rate` `non invasive blood pressure systolic`
        <int>    <int>        <dbl>                                  <dbl>
 1   10000032 39553978           94                                   83.5
 2   10000690 37081114           84                                   92.5
 3   10000980 39765666           69                                  131
 4   10001217 34592300           80                                  107
 5   10001217 37067082           93                                  144
 6   10001725 31205490           73                                   91
 7   10001843 39698942          126.                                  83.5
 8   10001884 37510196           74                                   86
 9   10002013 39060235           94                                  106
10   10002114 34672098           85                                  122.
   `non invasive blood pressure diastolic` `respiratory rate`
                                     <dbl>              <dbl>
 1                                    57                 20
 2                                    49                 26
 3                                    69                 21
 4                                    78                 22
 5                                    86                 17
 6                                    58                 23
 7                                    52.2               22.5
 8                                    48                 14
 9                                    60                 14
10                                    72                 25
   `temperature fahrenheit`
                      <dbl>
 1                     99.5
 2                     98
 3                     98.7
 4                     98.3
 5                     99.1
 6                     98.4
 7                     97.5
 8                     99.1
 9                     97.8
10                     98.2
# i more rows
```

Q1.7 Put things together

This step is similar to Q7 of HW3. Using *one* chain of pipes `|>` to perform following data wrangling steps: (i) start with the `icustays_tble`, (ii) merge in admissions and patients tables, (iii) keep adults only (age at ICU intime >= 18), (iv) merge in the labevents and chartevents tables, (v) `collect` the tibble, (vi) sort `subject_id`, `hadm_id`, `stay_id` and `print(width = Inf)`.

```r
# # TODO
mimic_icu_cohort <- icustays_tble |>
  left_join(admissions_tble, by = c("subject_id", "hadm_id")) |>
  left_join(patients_tble, by = "subject_id") |>
  filter(anchor_age >= 18) |>
  left_join(labevents_tble, by = c("subject_id", "stay_id")) |>
  left_join(chartevents_tble, by = c("subject_id", "stay_id")) |>
  collect() |>
  arrange(subject_id, hadm_id, stay_id) |>

  print(width = Inf)
```

```
Warning: ORDER BY is ignored in subqueries without LIMIT
ℹ Do you need to move arrange() later in the pipeline or use window_order()
instead?
ORDER BY is ignored in subqueries without LIMIT
ℹ Do you need to move arrange() later in the pipeline or use window_order()
instead?
ORDER BY is ignored in subqueries without LIMIT
ℹ Do you need to move arrange() later in the pipeline or use window_order()
instead?
ORDER BY is ignored in subqueries without LIMIT
ℹ Do you need to move arrange() later in the pipeline or use window_order()
instead?
ORDER BY is ignored in subqueries without LIMIT
ℹ Do you need to move arrange() later in the pipeline or use window_order()
instead?
```

```
# A tibble: 94,458 × 40
   subject_id  hadm_id  stay_id first_careunit
        <int>    <int>    <int> <chr>
 1   10000032 29079034 39553978 Medical Intensive Care Unit (MICU)
 2   10000690 25860671 37081114 Medical Intensive Care Unit (MICU)
 3   10000980 26913865 39765666 Medical Intensive Care Unit (MICU)
 4   10001217 24597018 37067082 Surgical Intensive Care Unit (SICU)
 5   10001217 27703517 34592300 Surgical Intensive Care Unit (SICU)
 6   10001725 25563031 31205490 Medical/Surgical Intensive Care Unit (MICU/SICU)
 7   10001843 26133978 39698942 Medical/Surgical Intensive Care Unit (MICU/SICU)
 8   10001884 26184834 37510196 Medical Intensive Care Unit (MICU)
 9   10002013 23581541 39060235 Cardiac Vascular Intensive Care Unit (CVICU)
10   10002114 27793700 34672098 Coronary Care Unit (CCU)
   last_careunit                                    intime
   <chr>                                            <dttm>
 1 Medical Intensive Care Unit (MICU)               2180-07-23 14:00:00
 2 Medical Intensive Care Unit (MICU)               2150-11-02 19:37:00
 3 Medical Intensive Care Unit (MICU)               2189-06-27 08:42:00
```

```
 4 Surgical Intensive Care Unit (SICU)              2157-11-20 19:18:02
 5 Surgical Intensive Care Unit (SICU)              2157-12-19 15:42:24
 6 Medical/Surgical Intensive Care Unit (MICU/SICU) 2110-04-11 15:52:22
 7 Medical/Surgical Intensive Care Unit (MICU/SICU) 2134-12-05 18:50:03
 8 Medical Intensive Care Unit (MICU)               2131-01-11 04:20:05
 9 Cardiac Vascular Intensive Care Unit (CVICU)     2160-05-18 10:00:53
10 Coronary Care Unit (CCU)                         2162-02-17 23:30:00
   outtime               los admittime           dischtime
   <dttm>              <dbl> <dttm>              <dttm>
 1 2180-07-23 23:50:47 0.410 2180-07-23 12:35:00 2180-07-25 17:55:00
 2 2150-11-06 17:03:17 3.89  2150-11-02 18:02:00 2150-11-12 13:45:00
 3 2189-06-27 20:38:27 0.498 2189-06-27 07:38:00 2189-07-03 03:00:00
 4 2157-11-21 22:08:00 1.12  2157-11-18 22:56:00 2157-11-25 18:00:00
 5 2157-12-20 14:27:41 0.948 2157-12-18 16:58:00 2157-12-24 14:55:00
 6 2110-04-12 23:59:56 1.34  2110-04-11 15:08:00 2110-04-14 15:00:00
 7 2134-12-06 14:38:26 0.825 2134-12-05 00:10:00 2134-12-06 12:54:00
 8 2131-01-20 08:27:30 9.17  2131-01-07 20:39:00 2131-01-20 05:15:00
 9 2160-05-19 17:33:33 1.31  2160-05-18 07:45:00 2160-05-23 13:30:00
10 2162-02-20 21:16:27 2.91  2162-02-17 22:32:00 2162-03-04 15:16:00
   deathtime           admission_type              admit_provider_id
   <dttm>              <chr>                       <chr>
 1 NA                  EW EMER.                    P06OTX
 2 NA                  EW EMER.                    P26QQ4
 3 NA                  EW EMER.                    P06OTX
 4 NA                  EW EMER.                    P3610N
 5 NA                  DIRECT EMER.                P276OU
 6 NA                  EW EMER.                    P32W56
 7 2134-12-06 12:54:00 URGENT                      P67ATB
 8 2131-01-20 05:15:00 OBSERVATION ADMIT           P49AFC
 9 NA                  SURGICAL SAME DAY ADMISSION P8286C
10 NA                  OBSERVATION ADMIT           P46834
   admission_location     discharge_location insurance language marital_status
   <chr>                  <chr>              <chr>     <chr>    <chr>
 1 EMERGENCY ROOM         HOME               Medicaid  English  WIDOWED
 2 EMERGENCY ROOM         REHAB              Medicare  English  WIDOWED
 3 EMERGENCY ROOM         HOME HEALTH CARE   Medicare  English  MARRIED
 4 EMERGENCY ROOM         HOME HEALTH CARE   Private   Other    MARRIED
 5 PHYSICIAN REFERRAL     HOME HEALTH CARE   Private   Other    MARRIED
 6 PACU                   HOME               Private   English  MARRIED
 7 TRANSFER FROM HOSPITAL DIED               Medicare  English  SINGLE
 8 EMERGENCY ROOM         DIED               Medicare  English  MARRIED
 9 PHYSICIAN REFERRAL     HOME HEALTH CARE   Medicare  English  SINGLE
10 PHYSICIAN REFERRAL     HOME HEALTH CARE   Medicaid  English  <NA>
   race                   edregtime           edouttime
   <chr>                  <dttm>              <dttm>
 1 WHITE                  2180-07-23 05:54:00 2180-07-23 14:00:00
 2 WHITE                  2150-11-02 11:41:00 2150-11-02 19:37:00
 3 BLACK/AFRICAN AMERICAN 2189-06-27 06:25:00 2189-06-27 08:42:00
```

```
 4 WHITE                   2157-11-18 17:38:00 2157-11-19 01:24:00
 5 WHITE                   NA                  NA
 6 WHITE                   NA                  NA
 7 WHITE                   NA                  NA
 8 BLACK/AFRICAN AMERICAN 2131-01-07 13:36:00 2131-01-07 22:13:00
 9 OTHER                   NA                  NA
10 UNKNOWN                 2162-02-17 19:35:00 2162-02-17 23:30:00
   hospital_expire_flag gender anchor_age anchor_year anchor_year_group
                  <int> <chr>       <int>       <int> <chr>
 1                    0 F              52        2180 2014 - 2016
 2                    0 F              86        2150 2008 - 2010
 3                    0 F              73        2186 2008 - 2010
 4                    0 F              55        2157 2011 - 2013
 5                    0 F              55        2157 2011 - 2013
 6                    0 F              46        2110 2011 - 2013
 7                    1 M              73        2131 2017 - 2019
 8                    1 F              68        2122 2008 - 2010
 9                    0 F              53        2156 2008 - 2010
10                    0 M              56        2162 2020 - 2022
   dod        bicarbonate chloride creatinine glucose potassium sodium
   <date>           <dbl>    <dbl>      <dbl>   <dbl>     <dbl>  <dbl>
 1 2180-09-09          25       95        0.7     102       6.7    126
 2 2152-01-30          26      100        1        85       4.8    137
 3 2193-08-26          21      109        2.3      89       3.9    144
 4 NA                  NA       NA       NA        NA      NA       NA
 5 NA                  30      104        0.5      87       4.1    142
 6 NA                  NA       98       NA        NA       4.1    139
 7 2134-12-06          28       97        1.3     131       3.9    138
 8 2131-01-20          30       88        1.1     141       4.5    130
 9 NA                  24      102        0.9     288       3.5    137
10 2162-12-11          18       NA        3.1      95       6.5    125
   hematocrit `white blood cells` `heart rate`
        <dbl>               <dbl>        <dbl>
 1       41.1                 6.9           94
 2       36.1                 7.1           84
 3       27.3                 5.3           69
 4       NA                  NA             93
 5       37.4                 5.4           80
 6       NA                  NA             73
 7       31.4                10.4          126.
 8       39.7                12.2           74
 9       34.9                 7.2           94
10       34.3                16.8           85
   `non invasive blood pressure systolic`
                                    <dbl>
 1                                   83.5
 2                                   92.5
 3                                  131
```

```
 4                                  144
 5                                  107
 6                                   91
 7                                   83.5
 8                                   86
 9                                  106
10                                  122.
   `non invasive blood pressure diastolic` `respiratory rate`
                                     <dbl>              <dbl>
 1                                    57                 20
 2                                    49                 26
 3                                    69                 21
 4                                    86                 17
 5                                    78                 22
 6                                    58                 23
 7                                    52.2               22.5
 8                                    48                 14
 9                                    60                 14
10                                    72                 25
   `temperature fahrenheit`
                      <dbl>
 1                     99.5
 2                     98
 3                     98.7
 4                     99.1
 5                     98.3
 6                     98.4
 7                     97.5
 8                     99.1
 9                     97.8
10                     98.2
# i 94,448 more rows
```

Q1.8 Preprocessing

Perform the following preprocessing steps. (i) Lump infrequent levels into "Other" level for `first_careunit`, `last_careunit`, `admission_type`, `admission_location`, and `discharge_location`. (ii) Collapse the levels of `race` into `ASIAN`, `BLACK`, `HISPANIC`, `WHITE`, and `Other`. (iii) Create a new variable `los_long` that is `TRUE` when `los` is greater than or equal to 2 days. (iv) Summarize the data using `tbl_summary()`, stratified by `los_long`. Hint: `fct_lump_n` and `fct_collapse` from the `forcats` package are useful.

Hint: Below is a numerical summary of my tibble after preprocessing:

```
# # TODO
library(forcats)

mimic_icu_cohort_sum <- mimic_icu_cohort |>
```

```
  select(first_careunit, last_careunit, los, admission_type, admission_location,
discharge_location, insurance, language, marital_status, race, gender, dod,
chloride, creatinine, sodium, potassium, glucose, hematocrit, `white blood
cells`, bicarbonate, `non invasive blood pressure systolic`, `non invasive blood
pressure diastolic`, `respiratory rate`, `temperature fahrenheit`, `heart rate`,
`anchor_age`) |>
  mutate(
    first_careunit = fct_lump_n(first_careunit, n = 4, other_level = "Other"),
    last_careunit = fct_lump_n(last_careunit, n = 4, other_level = "Other"),
    admission_type = fct_lump_n(admission_type, n = 4, other_level = "Other"),
     admission_location = fct_lump_n(admission_location, n = 3, other_level =
"Other"),
     discharge_location = fct_lump_n(discharge_location, n = 5, other_level =
"Other"),
    race = fct_collapse(
      race,
       ASIAN = c("ASIAN", "ASIAN - ASIAN INDIAN", "ASIAN - CHINESE", "ASIAN -
SOUTH EAST ASIAN", "ASIAN - KOREAN"),
     BLACK = c("BLACK/AFRICAN AMERICAN", "BLACK/CAPE VERDEAN", "BLACK/CARIBBEAN
ISLAND", "BLACK/CAPE VERDEAN", "BLACK/AFRICAN"),
         HISPANIC = c("HISPANIC OR LATINO", "HISPANIC/LATINO - SALVADORAN",
"HISPANIC/LATINO - CENTRAL AMERICAN", "HISPANIC/LATINO - COLUMBIAN", "HISPANIC/
LATINO - CUBAN", "HISPANIC/LATINO - GUATEMALAN", "HISPANIC/LATINO - DOMINICAN",
"HISPANIC/LATINO - GUATEMALAN", "HISPANIC/LATINO - HONDURAN", "HISPANIC/LATINO
- MEXICAN", "HISPANIC/LATINO - PUERTO RICAN"),
      WHITE = c("WHITE", "WHITE - RUSSIAN", "WHITE - OTHER EUROPEAN", "WHITE -
EASTERN EUROPEAN", "WHITE - BRAZILIAN", "WHITE - EASTERN EUROPEAN"),
       Other = c("UNKNOWN", "OTHER", "UNABLE TO OBTAIN", "PATIENT DECLINED TO
ANSWER", "MULTIPLE RACE/ETHNICITY", "NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER",
"AMERICAN INDIAN/ALASKA NATIVE", "SOUTH AMERICAN", "PORTUGUESE")
    ),
    los_long = ifelse(los >= 2, TRUE, FALSE)
  )

summary_tbl <- mimic_icu_cohort_sum |>
  tbl_summary(
    by = los_long,
   statistic = list(all_categorical() ~ "{n} ({p}%)", all_continuous() ~ "{mean}
({sd})"),
    missing = "no"
  ) |>
  add_p()
```

```
14 missing rows in the "los_long" column have been removed.
The following errors were returned during `add_p()`:
✖ For variable `dod` (`los_long`) and "statistic" and "p.value" statistics: 'x'
  must be numeric
```

```
✖ For variable `insurance` (`los_long`) and "estimate", "p.value", "conf.low",
  and "conf.high" statistics: FEXACT error 501. The hash table key cannot be
  computed because the largest key is larger than the largest representable
  int. The algorithm cannot proceed. Reduce the workspace, consider using
  'simulate.p.value=TRUE' or another algorithm.
```

```
# Display the summary table
summary_tbl
```

Characteristic	TRUE N = 46,337[1]	FALSE N = 48,107[1]	p-value[2]
first_careunit			<0.001
Cardiac Vascular Intensive Care Unit (CVICU)	7,353 (16%)	7,416 (15%)	
Medical Intensive Care Unit (MICU)	9,837 (21%)	10,862 (23%)	
Medical/Surgical Intensive Care Unit (MICU/SICU)	6,667 (14%)	8,780 (18%)	
Surgical Intensive Care Unit (SICU)	6,434 (14%)	6,574 (14%)	
Other	16,046 (35%)	14,475 (30%)	
last_careunit			<0.001
Cardiac Vascular Intensive Care Unit (CVICU)	7,353 (16%)	7,416 (15%)	
Medical Intensive Care Unit (MICU)	9,837 (21%)	10,862 (23%)	

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Characteristic	TRUE N = 46,337[1]	FALSE N = 48,107[1]	p-value[2]
Medical/Surgical Intensive Care Unit (MICU/SICU)	6,667 (14%)	8,780 (18%)	
Surgical Intensive Care Unit (SICU)	6,434 (14%)	6,574 (14%)	
Other	16,046 (35%)	14,475 (30%)	
los	6.2 (6.8)	1.1 (0.5)	<0.001
admission_type			<0.001
EW EMER.	23,012 (50%)	25,337 (53%)	
OBSERVATION ADMIT	7,393 (16%)	6,638 (14%)	
SURGICAL SAME DAY ADMISSION	4,001 (8.6%)	5,543 (12%)	
URGENT	8,691 (19%)	6,683 (14%)	
Other	3,240 (7.0%)	3,906 (8.1%)	
admission_location			<0.001
EMERGENCY ROOM	17,058 (37%)	20,443 (42%)	
PHYSICIAN REFERRAL	11,013 (24%)	12,684 (26%)	
TRANSFER FROM HOSPITAL	13,904 (30%)	10,400 (22%)	
Other	4,362 (9.4%)	4,580 (9.5%)	

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Characteristic	TRUE N = 46,337[1]	FALSE N = 48,107[1]	p-value[2]
discharge_location			<0.001
DIED	6,884 (15%)	4,436 (9.4%)	
HOME	6,879 (15%)	15,210 (32%)	
HOME HEALTH CARE	10,620 (23%)	13,422 (28%)	
REHAB	5,574 (12%)	2,445 (5.2%)	
SKILLED NURSING FACILITY	8,785 (19%)	7,489 (16%)	
Other	7,518 (16%)	4,334 (9.2%)	
insurance			
Medicaid	6,768 (15%)	7,469 (16%)	
Medicare	26,330 (58%)	25,485 (54%)	
No charge	5 (<0.1%)	3 (<0.1%)	
Other	1,091 (2.4%)	1,237 (2.6%)	
Private	11,515 (25%)	13,018 (28%)	
language			<0.001
American Sign Language	29 (<0.1%)	34 (<0.1%)	
Amharic	14 (<0.1%)	9 (<0.1%)	
Arabic	87 (0.2%)	62 (0.1%)	
Armenian	12 (<0.1%)	13 (<0.1%)	
Bengali	22 (<0.1%)	12 (<0.1%)	
Chinese	550 (1.2%)	611 (1.3%)	
English	41,563 (90%)	43,483 (91%)	

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Characteristic	TRUE N = 46,337[1]	FALSE N = 48,107[1]	p-value[2]
French	18 (<0.1%)	14 (<0.1%)	
Haitian	375 (0.8%)	252 (0.5%)	
Hindi	24 (<0.1%)	21 (<0.1%)	
Italian	101 (0.2%)	107 (0.2%)	
Japanese	5 (<0.1%)	7 (<0.1%)	
Kabuverdianu	301 (0.7%)	345 (0.7%)	
Khmer	50 (0.1%)	37 (<0.1%)	
Korean	40 (<0.1%)	32 (<0.1%)	
Modern Greek (1453-)	102 (0.2%)	88 (0.2%)	
Other	152 (0.3%)	153 (0.3%)	
Persian	42 (<0.1%)	35 (<0.1%)	
Polish	36 (<0.1%)	38 (<0.1%)	
Portuguese	351 (0.8%)	314 (0.7%)	
Russian	601 (1.3%)	659 (1.4%)	
Somali	8 (<0.1%)	15 (<0.1%)	
Spanish	1,472 (3.2%)	1,429 (3.0%)	
Thai	21 (<0.1%)	22 (<0.1%)	
Vietnamese	151 (0.3%)	129 (0.3%)	
marital_status			0.002
DIVORCED	3,377 (8.0%)	3,555 (8.0%)	
MARRIED	20,557 (49%)	21,344 (48%)	
SINGLE	12,745 (30%)	14,039 (31%)	
WIDOWED	5,319 (13%)	5,752 (13%)	
race			<0.001
Other	8,036 (17%)	6,880 (14%)	

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Characteristic	TRUE N = 46,337[1]	FALSE N = 48,107[1]	p-value[2]
ASIAN	1,369 (3.0%)	1,516 (3.2%)	
BLACK	4,933 (11%)	5,452 (11%)	
HISPANIC	1,687 (3.6%)	1,908 (4.0%)	
WHITE	30,312 (65%)	32,351 (67%)	
gender			<0.001
F	20,106 (43%)	21,471 (45%)	
M	26,231 (57%)	26,636 (55%)	
dod	2155-09-17 (8858.35[illegible])	(8884.44648637213)	
chloride	101.3 (6.5)	101.7 (5.9)	<0.001
creatinine	1.46 (1.54)	1.35 (1.57)	<0.001
sodium	138.0 (5.7)	138.2 (5.0)	<0.001
potassium	4.33 (0.80)	4.32 (0.79)	0.082
glucose	143 (87)	140 (87)	<0.001
hematocrit	35 (7)	36 (7)	<0.001
white blood cells	11.5 (9.8)	10.7 (9.6)	<0.001
bicarbonate	23.9 (5.1)	23.9 (4.8)	0.008
non invasive blood pressure systolic	121 (22)	121 (77)	<0.001
non invasive blood pressure diastolic	69 (414)	68 (133)	0.11
respiratory rate	19.7 (6.2)	19.1 (6.0)	<0.001
temperature fahrenheit	98.29 (8.57)	98.27 (11.02)	0.12
heart rate	83 (21)	82 (39)	<0.001

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Characteristic	TRUE N = 46,337[1]	FALSE N = 48,107[1]	p-value[2]
anchor_age	64 (16)	62 (17)	<0.001

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Q1.9 Save the final tibble

Save the final tibble to an R data file `mimic_icu_cohort.rds` in the `mimiciv_shiny` folder.

```
# make a directory mimiciv_shiny
if (!dir.exists("mimiciv_shiny")) {
  dir.create("mimiciv_shiny")
}
# save the final tibble
mimic_icu_cohort <- mimic_icu_cohort |>
  rename_with(~ gsub(" ", "_", .)) |>
  write_rds("mimiciv_shiny/mimic_icu_cohort.rds", compress = "gz")
```

Close database connection and clear workspace.

```
if (exists("con_bq")) {
  dbDisconnect(con_bq)
}
rm(list = ls())
```

Although it is not a good practice to add big data files to Git, for grading purpose, please add `mimic_icu_cohort.rds` to your Git repository.

Q2. Shiny app

Develop a Shiny app for exploring the ICU cohort data created in Q1. The app should reside in the `mimiciv_shiny` folder. The app should contain at least two tabs. One tab provides easy access to the graphical and numerical summaries of variables (demographics, lab measurements, vitals) in the ICU cohort, using the `mimic_icu_cohort.rds` you curated in Q1. The other tab allows user to choose a specific patient in the cohort and display the patient's ADT and ICU stay information as we did in Q1 of HW3, by dynamically retrieving the patient's ADT and ICU stay information from BigQuery database. Again, do **not** ever add the BigQuery token to your Git repository. If you do so, you will lose 50 points.