Biostat 203B Homework 3

Due Feb 21 @ 11:59PM

AUTHOR
Zhiyuan Yu 906405523

Display machine information for reproducibility:

```
sessionInfo()
```

```
R version 4.4.2 (2024-10-31)
Platform: x86_64-pc-linux-gnu
Running under: Ubuntu 24.04.1 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.12.0
LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.12.0

locale:
 [1] LC_CTYPE=C.UTF-8       LC_NUMERIC=C           LC_TIME=C.UTF-8
 [4] LC_COLLATE=C.UTF-8     LC_MONETARY=C.UTF-8    LC_MESSAGES=C.UTF-8
 [7] LC_PAPER=C.UTF-8       LC_NAME=C              LC_ADDRESS=C
[10] LC_TELEPHONE=C         LC_MEASUREMENT=C.UTF-8 LC_IDENTIFICATION=C

time zone: America/Los_Angeles
tzcode source: system (glibc)

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

loaded via a namespace (and not attached):
 [1] htmlwidgets_1.6.4 compiler_4.4.2    fastmap_1.2.0     cli_3.6.3
 [5] tools_4.4.2       htmltools_0.5.8.1 rstudioapi_0.17.1 yaml_2.3.10
 [9] rmarkdown_2.29    knitr_1.49        jsonlite_1.8.9    xfun_0.50
[13] digest_0.6.37     rlang_1.1.4       evaluate_1.0.3
```

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

```
library(arrow)
```

```
Attaching package: 'arrow'

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

    timestamp
```

```
library(gtsummary)
library(memuse)
library(pryr)
```

```
Attaching package: 'pryr'

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

    where
```

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

```
Loading required package: R.oo

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 v1.27.0 (2024-11-01 18:00:02 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, use, warnings
```

```
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 1.9.4     ✓ tidyr     1.3.1
✓ purrr     1.0.2
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
✖ purrr::compose()      masks pryr::compose()
✖ lubridate::duration() masks arrow::duration()
✖ tidyr::extract()      masks R.utils::extract()
✖ dplyr::filter()       masks stats::filter()
✖ dplyr::lag()          masks stats::lag()
✖ purrr::partial()      masks pryr::partial()
✖ dplyr::where()        masks pryr::where(), gtsummary::where()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
errors
```

```
library(readr)
library(dplyr)
library(ggplot2)
library(DescTools)
library(data.table)
```

```
Attaching package: 'data.table'

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

    %like%

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

    hour, isoweek, mday, minute, month, quarter, second, wday, week,
    yday, year

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

    between, first, last

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

    transpose
```

```
The following object is masked from 'package:pryr':

    address
```

```
library(rlang)
```

```
Attaching package: 'rlang'

The following object is masked from 'package:data.table':

    :=

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

    %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
    flatten_raw, invoke, splice

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

    env

The following objects are masked from 'package:R.oo':

    abort, ll

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

    bytes

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

    string
```

Display your machine memory.

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

```
Totalram:  15.463 GiB
Freeram:   11.619 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.

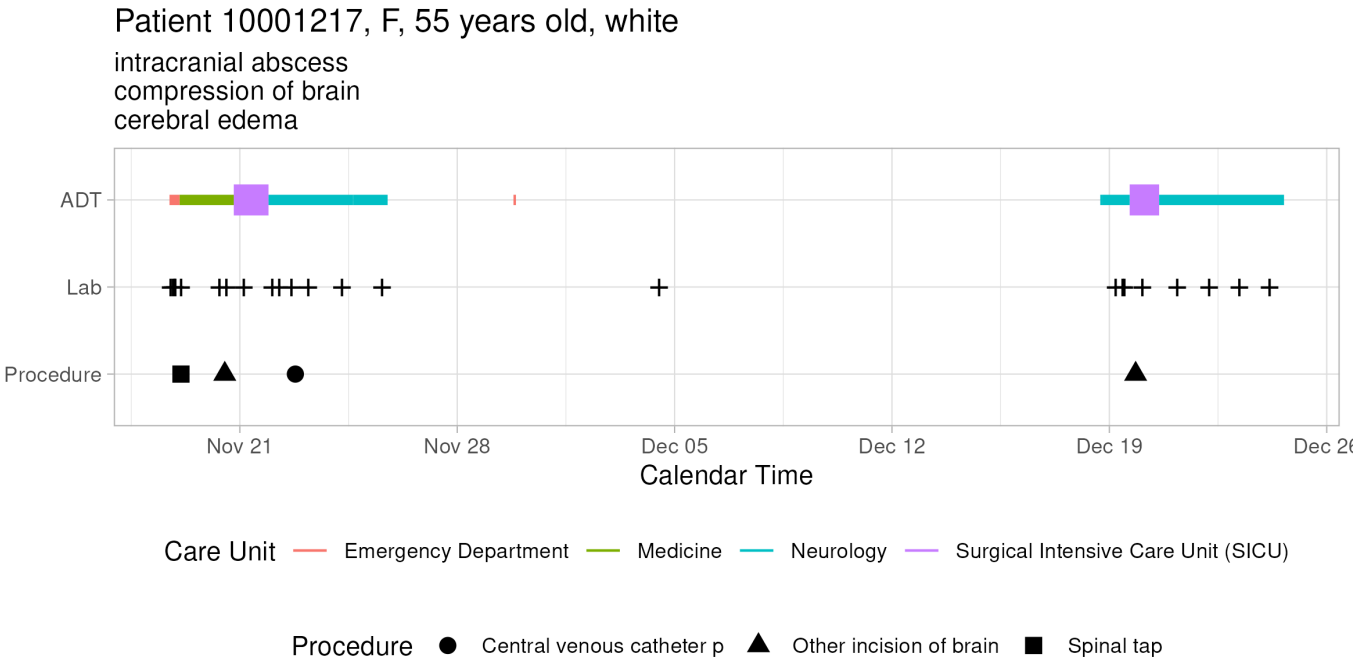
Q1. Visualizing patient trajectory

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.

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.

Solution: Step 1: Load in required datasets

```
patients <- read_csv("~/mimic/hosp/patients.csv.gz")
admissions <- read_csv("~/mimic/hosp/admissions.csv.gz")
transfers <- read_csv("~/mimic/hosp/transfers.csv.gz")
labevents_data <- arrow::open_dataset("~/mimic/hosp/labevents.csv",
                                      format = "csv")
arrow::write_dataset(labevents_data, path = "labevents_pq", format = "parquet")
dataset_parquet <- arrow::open_dataset ("labevents_pq", format = "parquet")
procedures_icd <- read_csv("~/mimic/hosp/procedures_icd.csv.gz")
diagnoses_icd <- read_csv("~/mimic/hosp/diagnoses_icd.csv.gz")
```

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

Step 2: Take a peek of the composition of the data

```
zcat < ~/mimic/hosp/patients.csv.gz | head
zcat < ~/mimic/hosp/admissions.csv.gz | head
zcat < ~/mimic/hosp/transfers.csv.gz | head
zcat < ~/mimic/hosp/labevents.csv.gz | head
zcat < ~/mimic/hosp/procedures_icd.csv.gz | head
zcat < ~/mimic/hosp/diagnoses_icd.csv.gz | head
zcat < ~/mimic/hosp/d_icd_procedures.csv.gz | head
zcat < ~/mimic/hosp/d_icd_diagnoses.csv.gz | head
```

Step 3: Get the required information for patient 10063848

```
subject_id <- 10063848

# Get race info
race <- read_csv("~/mimic/hosp/admissions.csv.gz") %>%
  filter(subject_id == !!subject_id) %>%
  distinct(race)
```

```
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.
```

```
# Get demographic info
demographics <- read_csv("~/mimic/hosp/patients.csv.gz") %>%
  filter(subject_id == !!subject_id) %>%
  mutate(race = tolower(race$race))
```

```
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.
```

```
# Get top 3 diagnoses info
top_3_diagnoses <- read_csv("~/mimic/hosp/diagnoses_icd.csv.gz") %>%
  filter(subject_id == !!subject_id) %>%
  left_join(read_csv("~/mimic/hosp/d_icd_diagnoses.csv.gz"),
            by = c("icd_code" = "icd_code", "icd_version" = "icd_version")) %>%
  group_by(long_title) %>%
  summarise(freq = n()) %>%
  arrange(desc(freq)) %>%
  slice(1:3)
```

```
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.
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.
```

```
# Get ADT info
ADT <- read_csv("~/mimic/hosp/transfers.csv.gz") %>%
  filter(subject_id == !!subject_id) %>%
  filter(!is.na(careunit)) %>%
  filter(!is.na(intime) & !is.na(outtime)) %>%
  mutate(segment_thickness = if_else(str_detect(careunit, "Care Unit"), 10, 8))
```

```
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.
```

```
# Get lab events info
labevents <- arrow::open_dataset('./labevents_pq', format = "parquet") %>%
  dplyr::select(subject_id, charttime) %>%
  dplyr::filter(subject_id == !!subject_id) %>%
  dplyr::distinct(subject_id, charttime) %>%
```

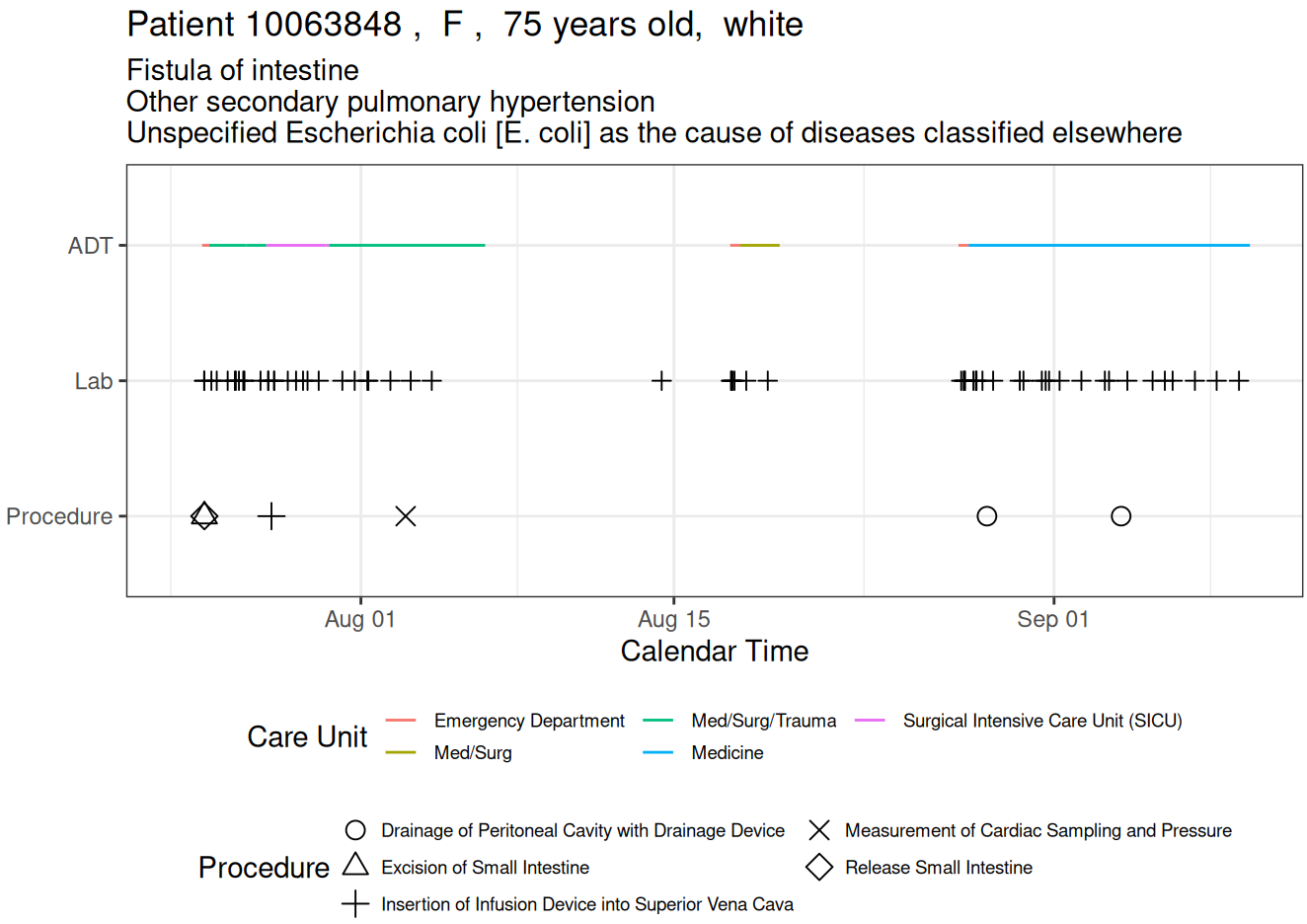
```
  collect()

# Get procedures info
procedures <- read_csv("~/mimic/hosp/procedures_icd.csv.gz") %>%
  filter(subject_id == !!subject_id) %>%
  left_join(read_csv("~/mimic/hosp/d_icd_procedures.csv.gz"),
            by = c("icd_code" = "icd_code", "icd_version" = "icd_version"))
```

```
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.
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.
```

Step 4: Make the required plot

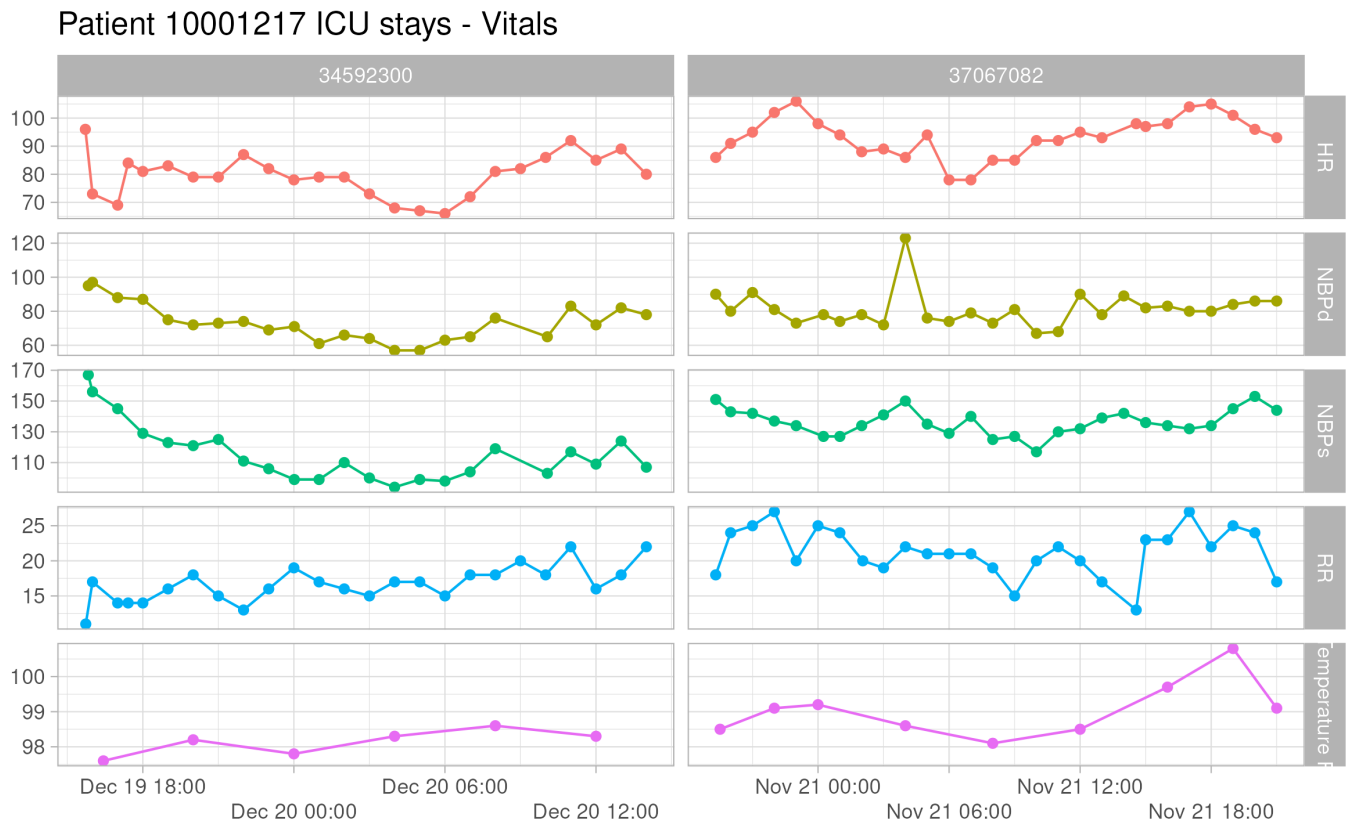
```
# Create the ADT history plot
ADT_history <- ggplot() +

  # Specify x-axis limits dynamically
  scale_x_datetime(name = "Calendar Time",
                   limits = c(min(ADT$intime) - days(1), max(ADT$outtime))) +

  # Specify y-axis with 3 levels
  scale_y_discrete(name = NULL,
                   limits = c("Procedure", "Lab", "ADT")) +

  # Add procedure events
  geom_point(data = procedures,
             aes(x = as.POSIXct(chartdate),
                 y = "Procedure",
                 shape = sub(",.*", "", long_title)),
             size = 3, color = "black") +

  scale_shape_manual(values = c(1:n_distinct(procedures$long_title))) +

  # Add lab events
  geom_point(data = labevents,
             aes(x = charttime, y = "Lab"),
```

```
            shape = 3, size = 2, color = "black") +

  # Add ADT events
  geom_segment(data = ADT,
               aes(x = intime,
                   xend = outtime,
                   y = "ADT",
                   yend = "ADT",
                   color = careunit,
                   )) +

  # Set legend position and arrangement
  theme_bw() +
  theme(legend.position = "bottom",
        legend.box = "vertical",
        legend.key.size = unit(0, "pt"),
        legend.text = element_text(size = 7)) +

  # Set legend titles and arrangement
  guides(color = guide_legend(title = "Care Unit",
                              ncol = 3,
                              keywidth = 1),
         shape = guide_legend(title = "Procedure",
                              ncol = 2),
         size = "none") +

  # Add patient information as title and subtitle
  labs(title = paste("Patient", demographics$subject_id[1], ", ",
                     demographics$gender[1], ", ",
                     demographics$anchor_age[1], "years old, ",
                     demographics$race[1]),
       subtitle = paste(top_3_diagnoses$long_title[1],
                        top_3_diagnoses$long_title[2],
                        top_3_diagnoses$long_title[3],
                        sep = "\n"))

print(ADT_history)
```



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`.

Solution: Step 1: decompress chartevents.csv.gz

```
gzip -d -k ~/mimic/icu/chartevents.csv.gz
```

Step 2: prepare the data

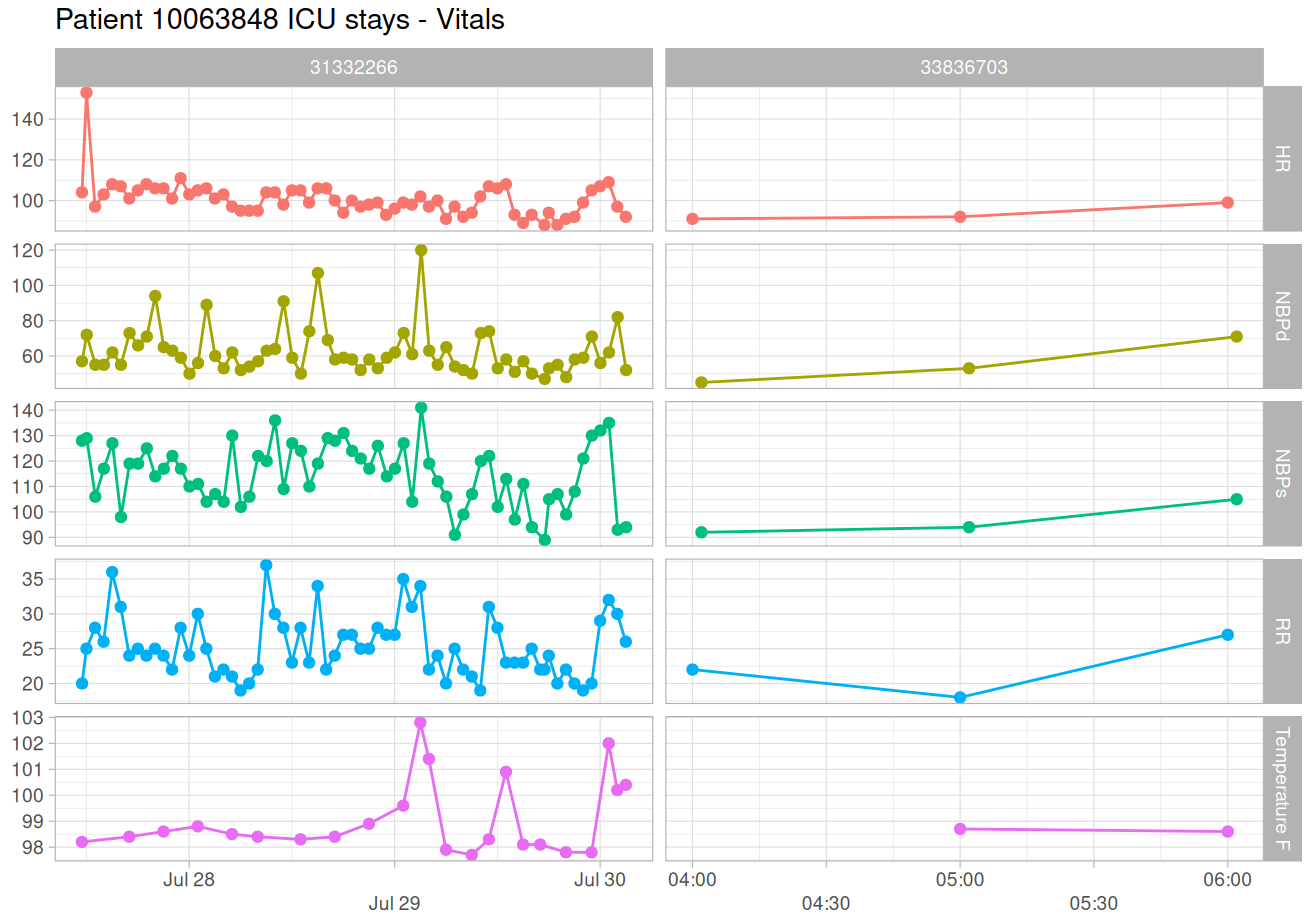
```
target_id <- 10063848

# Read d_items.csv.gz to get vital sign labels
items <- read_csv("~/mimic/icu/d_items.csv.gz") %>%
  dplyr::select(itemid, label, abbreviation) %>%
  dplyr::filter(abbreviation %in% c("HR", "NBPd", "NBPs", "RR",
                                    "Temperature F"))
```

```
Rows: 4095 Columns: 9
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (6): label, abbreviation, linksto, category, unitname, param_type
dbl (3): itemid, lownormalvalue, highnormalvalue

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.
```

```
chartevents_data <- arrow::open_dataset("~/mimic/icu/chartevents.csv",
                                        format = "csv")

chart_tibble <- chartevents_data %>%
  filter(subject_id == target_id,
         itemid %in% c(220045, 220180, 220179, 223761, 220210)) %>%
  select(subject_id, stay_id, itemid, charttime, valuenum) %>%
  collect()

# Merge with items to get vital sign labels
chart_tibble <- chart_tibble %>%
  left_join(items, by = "itemid")
```

Step 4: plot the plots

```
vitals_line_plot <- ggplot(chart_tibble,
                           aes(x = charttime,
                               y = valuenum,
                               color = abbreviation)) +
  geom_point() +
  geom_line() +

  facet_grid(abbreviation ~ stay_id, scales = "free") +
  labs(title = paste("Patient",
                     chart_tibble$subject_id[1],
                     "ICU stays - Vitals"),
       x = "",
       y = "") +

  theme_light(base_size = 9) + theme(legend.position = "none") +
  guides(x = guide_axis(n.dodge = 2))

print(vitals_line_plot)
```



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
(MICU),2180-07-23 14:00:00,2180-07-23 23:50:47,0.4102662037037037
10000690,25860671,37081114,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit
(MICU),2150-11-02 19:37:00,2150-11-06 17:03:17,3.8932523148148146
10000980,26913865,39765666,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit
(MICU),2189-06-27 08:42:00,2189-06-27 20:38:27,0.4975347222222222
10001217,24597018,37067082,Surgical Intensive Care Unit (SICU),Surgical Intensive Care Unit
(SICU),2157-11-20 19:18:02,2157-11-21 22:08:00,1.1180324074074075
10001217,27703517,34592300,Surgical Intensive Care Unit (SICU),Surgical Intensive Care Unit
(SICU),2157-12-19 15:42:24,2157-12-20 14:27:41,0.948113425925926
10001725,25563031,31205490,Medical/Surgical Intensive Care Unit (MICU/SICU),Medical/Surgical
Intensive Care Unit (MICU/SICU),2110-04-11 15:52:22,2110-04-12 23:59:56,1.338587962962963
10001843,26133978,39698942,Medical/Surgical Intensive Care Unit (MICU/SICU),Medical/Surgical
Intensive Care Unit (MICU/SICU),2134-12-05 18:50:03,2134-12-06 14:38:26,0.8252662037037037
```

```
10001884,26184834,37510196,Medical Intensive Care Unit (MICU),Medical Intensive Care Unit
(MICU),2131-01-11 04:20:05,2131-01-20 08:27:30,9.17181712962963
10002013,23581541,39060235,Cardiac Vascular Intensive Care Unit (CVICU),Cardiac Vascular
Intensive Care Unit (CVICU),2160-05-18 10:00:53,2160-05-19 17:33:33,1.314351851851852
```

Q2.1 Ingestion

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

```
icustays_tble <- open_dataset("~/mimic/icu/icustays.csv.gz",
                                    format = "csv") %>% collect()
```

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.

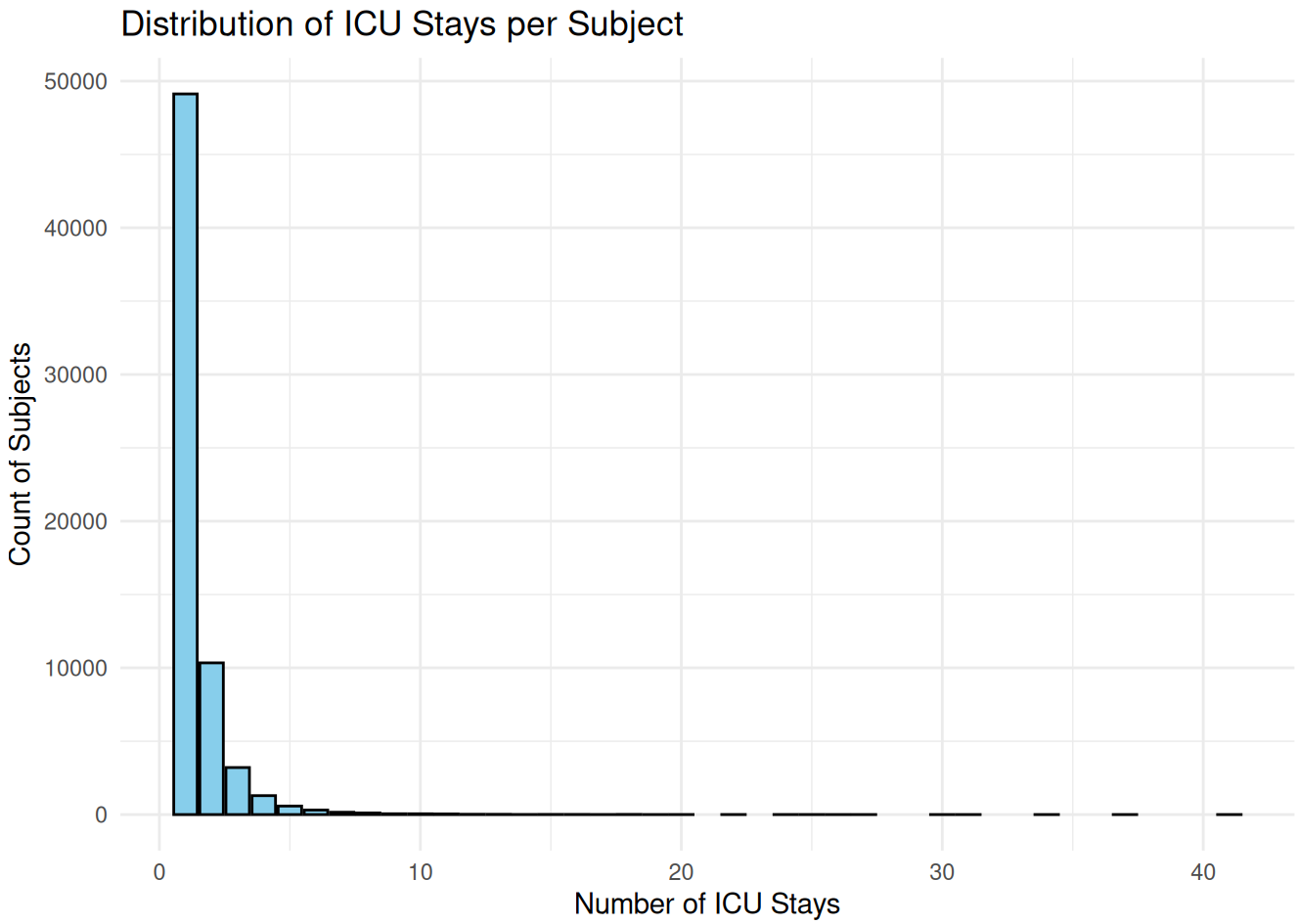
```
# Count the number of unique subject_id
num_unique_subjects <- icustays_tble %>%
  distinct(subject_id) %>%
  nrow()

cat("Number of unique subject_id:", num_unique_subjects, "\n")
```

```
Number of unique subject_id: 65366
```

```
# Count ICU stays per subject_id
icu_stays_per_subject <- icustays_tble %>%
  group_by(subject_id) %>%
  summarise(num_stays = n())

# Check if a subject_id can have multiple stays
max_stays <- max(icu_stays_per_subject$num_stays)
cat("Maximum ICU stays for a single subject_id:", max_stays, "\n")
```

```
Maximum ICU stays for a single subject_id: 41
```

```
# Plot the distribution of ICU stays per subject_id
ggplot(icu_stays_per_subject, aes(x = num_stays)) +
  geom_bar(fill = "skyblue", color = "black") +
  labs(title = "Distribution of ICU Stays per Subject",
       x = "Number of ICU Stays",
       y = "Count of Subjects") +
  theme_minimal()
```



There are 65366 unique subject_id. A single subject_id can have multiple stays and the maximum ICU stays for a single subject_id is 41.

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_locat
ion,discharge_location,insurance,language,marital_status,race,edregtime,edouttime,hospital_expire
_flag
10000032,22595853,2180-05-06 22:23:00,2180-05-07 17:15:00,,URGENT,P49AFC,TRANSFER FROM
HOSPITAL,HOME,Medicaid,English,WIDOWED,WHITE,2180-05-06 19:17:00,2180-05-06 23:30:00,0
10000032,22841357,2180-06-26 18:27:00,2180-06-27 18:49:00,,EW EMER.,P784FA,EMERGENCY
ROOM,HOME,Medicaid,English,WIDOWED,WHITE,2180-06-26 15:54:00,2180-06-26 21:31:00,0
10000032,25742920,2180-08-05 23:44:00,2180-08-07 17:50:00,,EW EMER.,P19UTS,EMERGENCY
ROOM,HOSPICE,Medicaid,English,WIDOWED,WHITE,2180-08-05 20:58:00,2180-08-06 01:44:00,0
10000032,29079034,2180-07-23 12:35:00,2180-07-25 17:55:00,,EW EMER.,P06OTX,EMERGENCY
ROOM,HOME,Medicaid,English,WIDOWED,WHITE,2180-07-23 05:54:00,2180-07-23 14:00:00,0
10000068,25022803,2160-03-03 23:16:00,2160-03-04 06:26:00,,EU OBSERVATION,P39NWO,EMERGENCY
```

```
ROOM,,,English,SINGLE,WHITE,2160-03-03 21:55:00,2160-03-04 06:26:00,0
10000084,23052089,2160-11-21 01:56:00,2160-11-25 14:52:00,,EW EMER.,P42H7G,WALK-IN/SELF
REFERRAL,HOME HEALTH CARE,Medicare,English,MARRIED,WHITE,2160-11-20 20:36:00,2160-11-21
03:20:00,0
10000084,29888819,2160-12-28 05:11:00,2160-12-28 16:07:00,,EU OBSERVATION,P35NE4,PHYSICIAN
REFERRAL,,Medicare,English,MARRIED,WHITE,2160-12-27 18:32:00,2160-12-28 16:07:00,0
10000108,27250926,2163-09-27 23:17:00,2163-09-28 09:04:00,,EU OBSERVATION,P40JML,EMERGENCY
ROOM,,,English,SINGLE,WHITE,2163-09-27 16:18:00,2163-09-28 09:04:00,0
10000117,22927623,2181-11-15 02:05:00,2181-11-15 14:52:00,,EU OBSERVATION,P47EY8,EMERGENCY
ROOM,,Medicaid,English,DIVORCED,WHITE,2181-11-14 21:51:00,2181-11-15 09:57:00,0
```

Q3.1 Ingestion

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

Solution:

```
admissions_tble <- open_dataset("~/mimic/hosp/admissions.csv.gz",
  format = "csv") %>% collect()
```

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?)

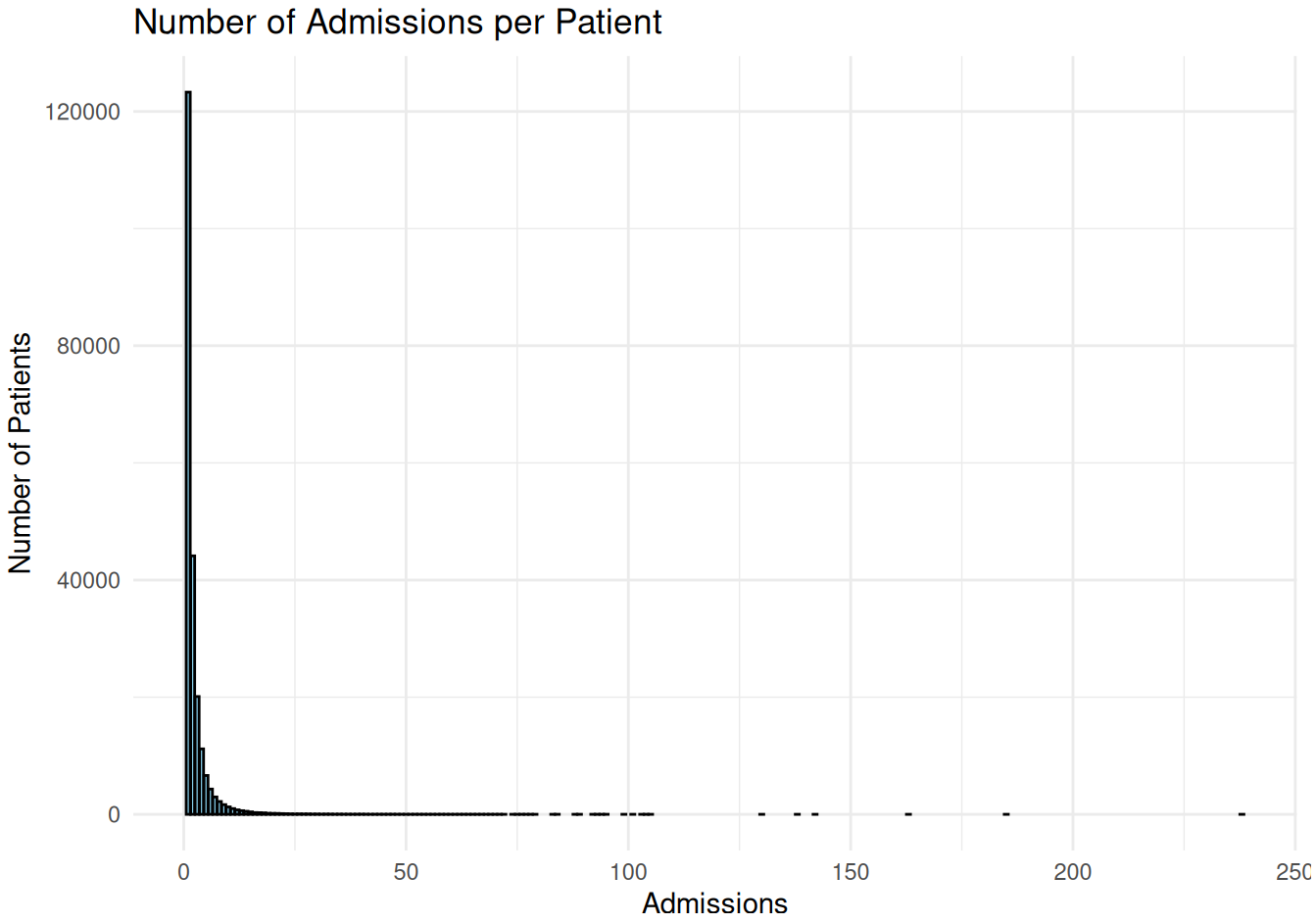
According to the MIMIC-IV documentation,

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.

Solution:

```
# Convert datetime columns to proper format
admissions_tble <- admissions_tble %>%
  mutate(admittime = as.POSIXct(admittime, format="%Y-%m-%d %H:%M:%S"),
         dischtime = as.POSIXct(dischtime, format="%Y-%m-%d %H:%M:%S"))

# 1. Number of Admissions per Patient
admissions_per_patient <- admissions_tble %>%
  count(subject_id)

# Plot the distribution of admissions per patient
ggplot(admissions_per_patient, aes(x = n)) +
```

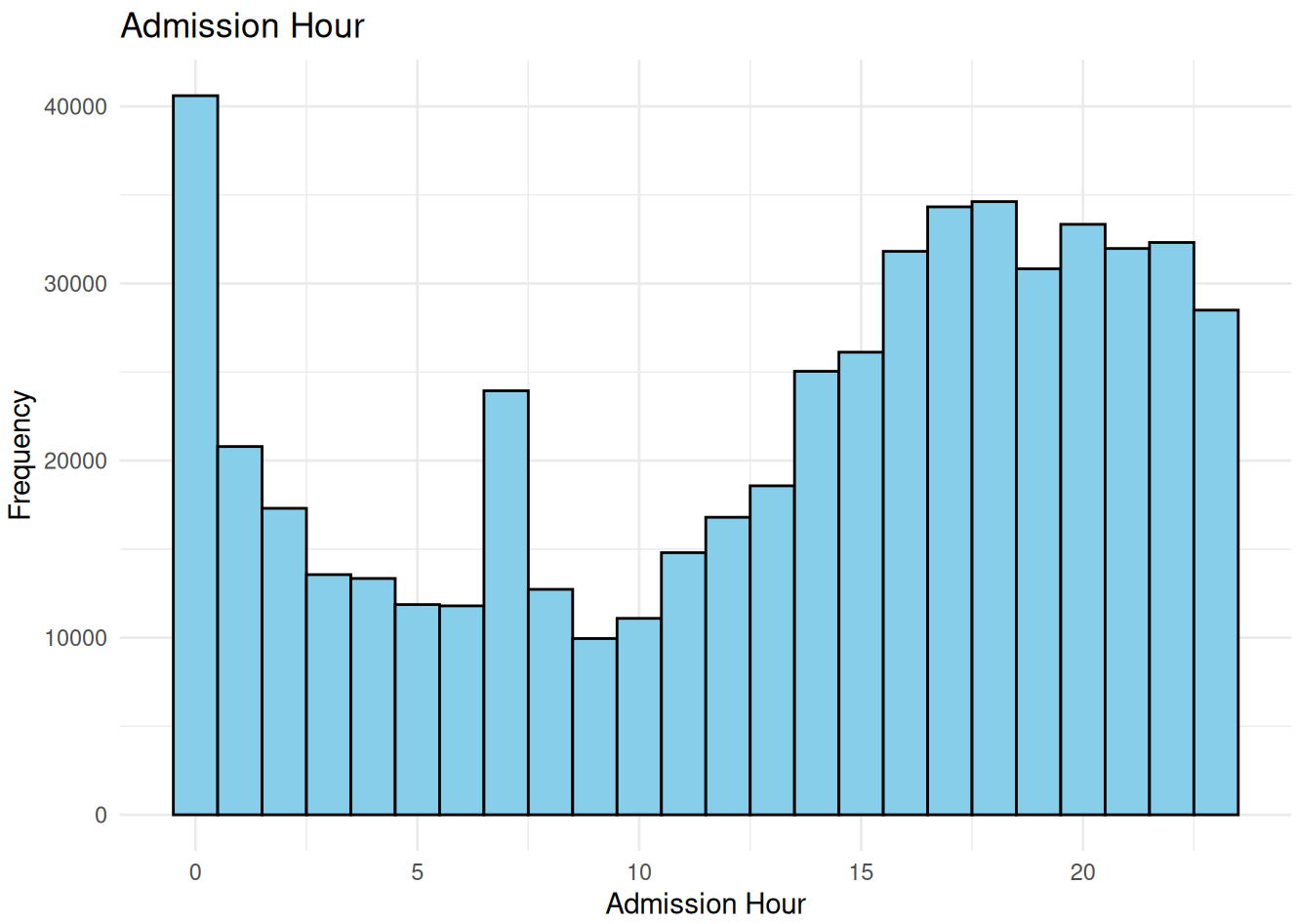
```
  geom_bar(fill = "skyblue", color = "black") +
  labs(title = "Number of Admissions per Patient",
       x = "Admissions",
       y = "Number of Patients") +
  theme_minimal()
```



Based on the graph, we see that most of the patients have only 1 admission but some have more than one admissions and there are some extreme cases of over 125 stays. This situation could be explained by most of the patients have been treated well and recovered so that they won't need to go back to the hospital again, but some patients might experienced more serious diseases that cannot be cured within a few hospital visits, or some patients suffered from chronic disease or disease relapsing and would have to be readmitted to the hospital.

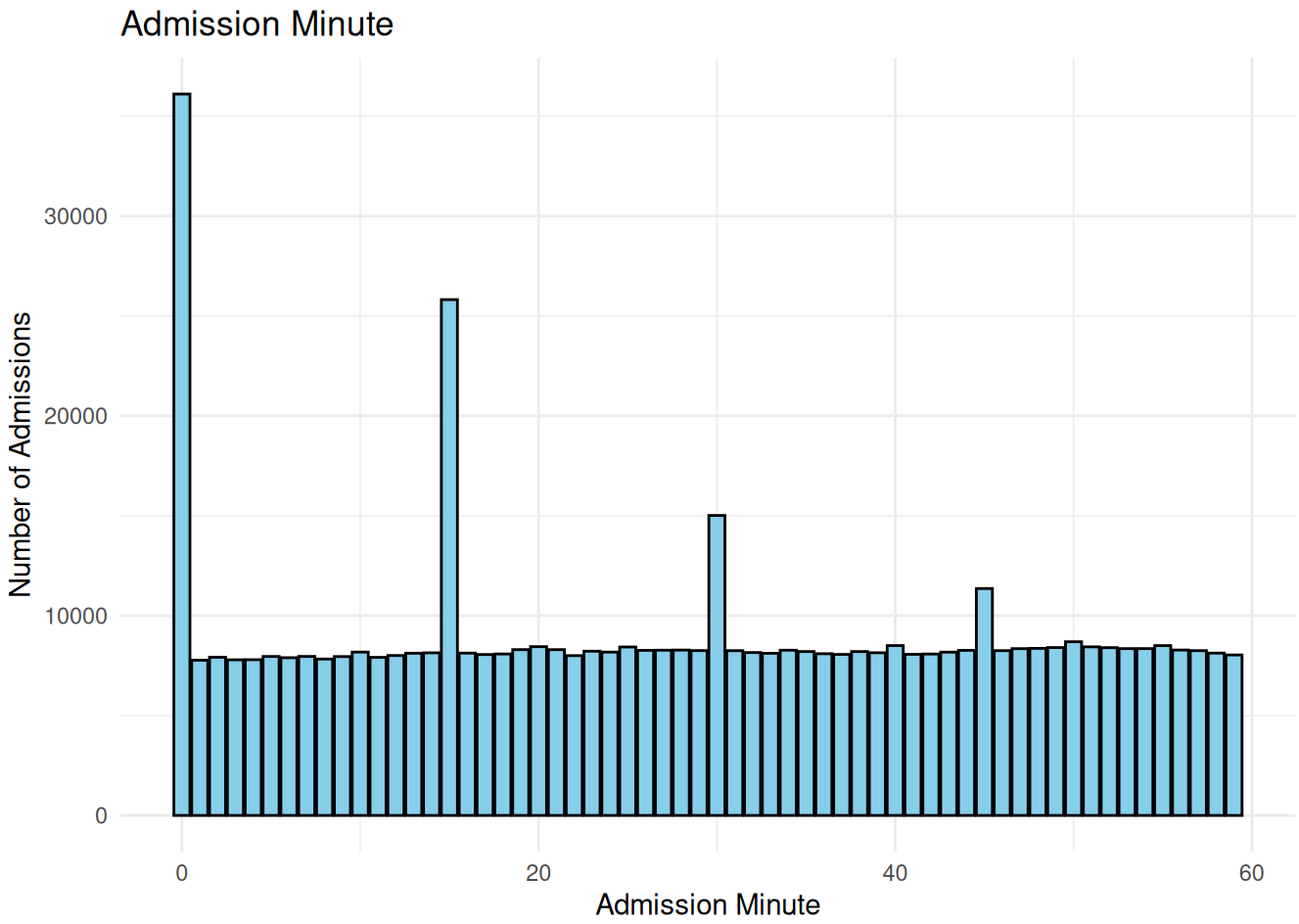
```
# 2. Admission Hour Analysis
admissions_tble_admithour <- admissions_tble %>%
  mutate(admittime = with_tz(admittime, "UTC"))

# Plot admission hour distribution
admissions_tble_admithour %>%
  ggplot(aes(x = hour(admittime))) +
  geom_histogram(binwidth = 1, fill = "skyblue", color = "black") +
  labs(title = "Admission Hour",
       x = "Admission Hour",
```

```
       y = "Frequency") +
  theme_minimal()
```



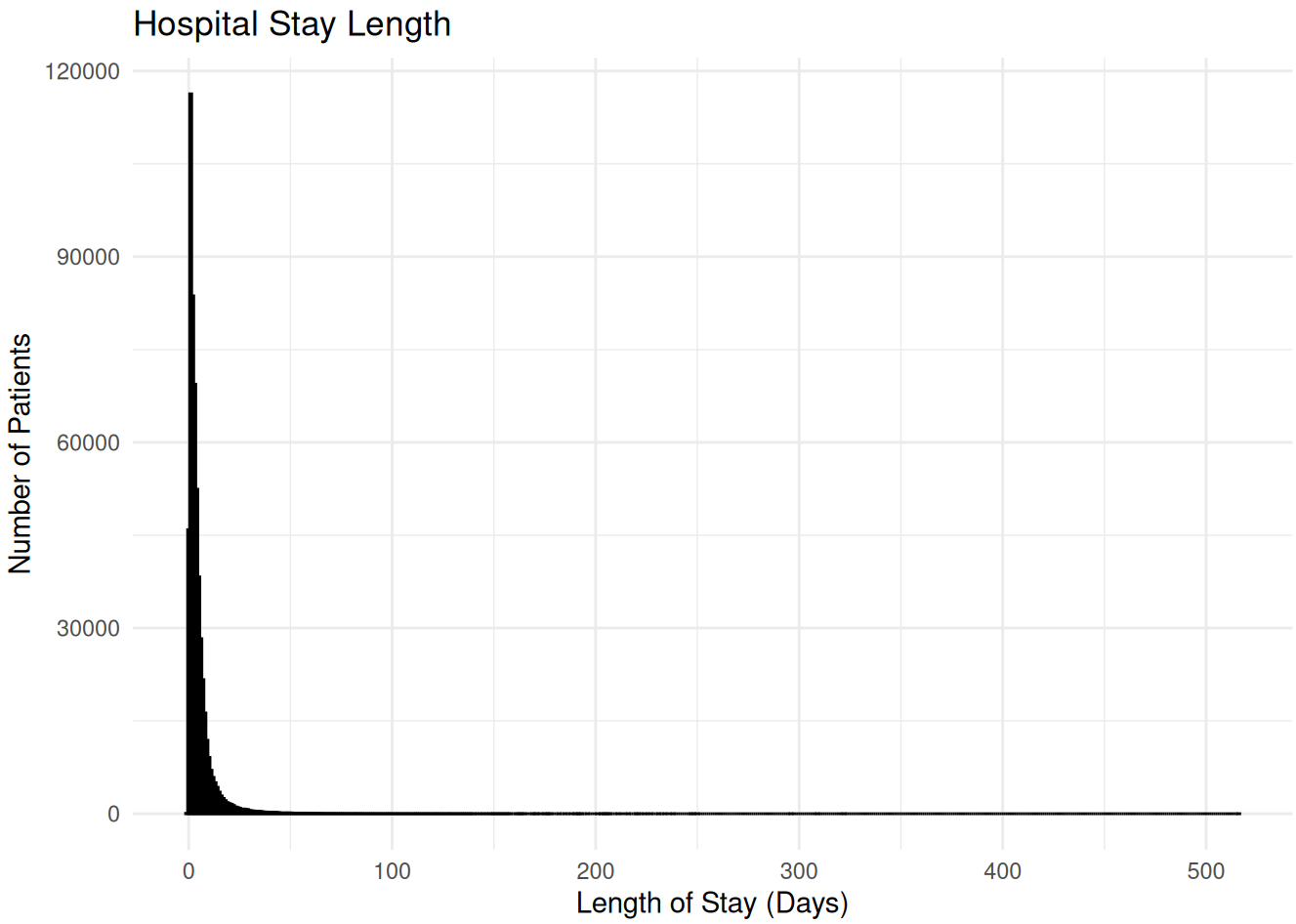
Based on the graph, we can see that the the number of admissions is peaked at 0 and then drops to around 12000 till a sudden stick-out to almost 25000 at 7, then drops again till a rise appears at 10 again till 18 then remain relatively stable after that. Some unusual things can be observed and explained. The peak number of admission at 0 could be explained by the patients suffer sudden burst of acute diseases when they thought they were alright and relax. That could be the reason why we hear the sound of ambulance relatively frequently at midnight. The sitck-out at 7 could be explained by the reason that people are trying to get themselves treated as early as possible once the hospital opens so that they could avoid the crowds.

```
# 3. Admission Minute Analysis
admissions_tble <- admissions_tble %>%
  mutate(admission_minute = minute(admittime))

# Plot admission minute distribution
ggplot(admissions_tble, aes(x = admission_minute)) +
  geom_bar(fill = "skyblue", color = "black") +
  labs(title = "Admission Minute",
       x = "Admission Minute",
       y = "Number of Admissions") +
  theme_minimal()
```



Based on the graph, we can see that while the distribution of the admission minute is mostly normal, there are some unusual stick-outs in frequencies at 0, 16, 30, and 45 minutes. The situation could be explained by some admission minutes have been rounded up or down and they happen to be cumulated at those listed certain minutes.

```
# 4. Length of Hospital Stay
admissions_tble <- admissions_tble %>%
  mutate(length_of_stay_days = as.numeric(difftime(dischtime, admittime, units="days")))

# Plot the distribution of hospital stay length
ggplot(admissions_tble, aes(x = length_of_stay_days)) +
  geom_histogram(binwidth = 1, fill = "skyblue", color = "black") +
  labs(title = "Hospital Stay Length",
       x = "Length of Stay (Days)",
       y = "Number of Patients") +
  theme_minimal()
```



Based on the graph, the distribution of length of stay is highly right skewed, with most of the patients stay in the hospital for less than 10 days wiht only very few exceptions, which is normal. This could be explain by the fact that most diseases or harm can be cured within a few days and there is no reason for the hospital to keep their patients in while they do not need to be, but some patients might suffer more serious harm that they need longer time of caring at the hospital.

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,
```

Q4.1 Ingestion

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

Solution:

```
patients_tble <- open_dataset(sources = "~/mimic/hosp/patients.csv.gz",
  format = "csv") %>% dplyr::collect()
```

Q4.2 Summary and visualization

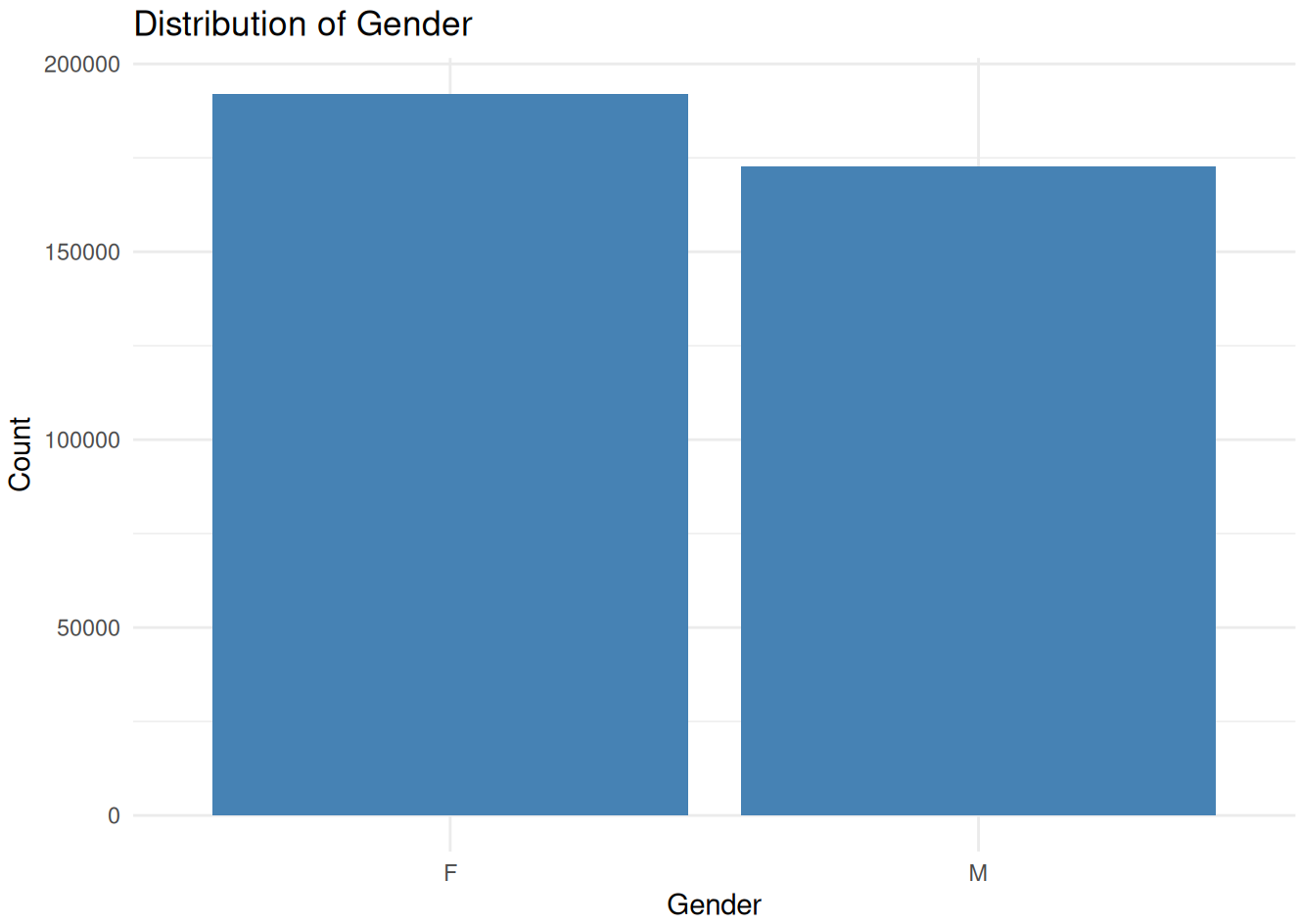
Summarize variables `gender` and `anchor_age` by graphics, and explain any patterns you see. Step1: create the patients_tble

```
# Import the CSV file as a tibble
patients_tble <- 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.
```

Step 2: create bar charts to summarize gender info

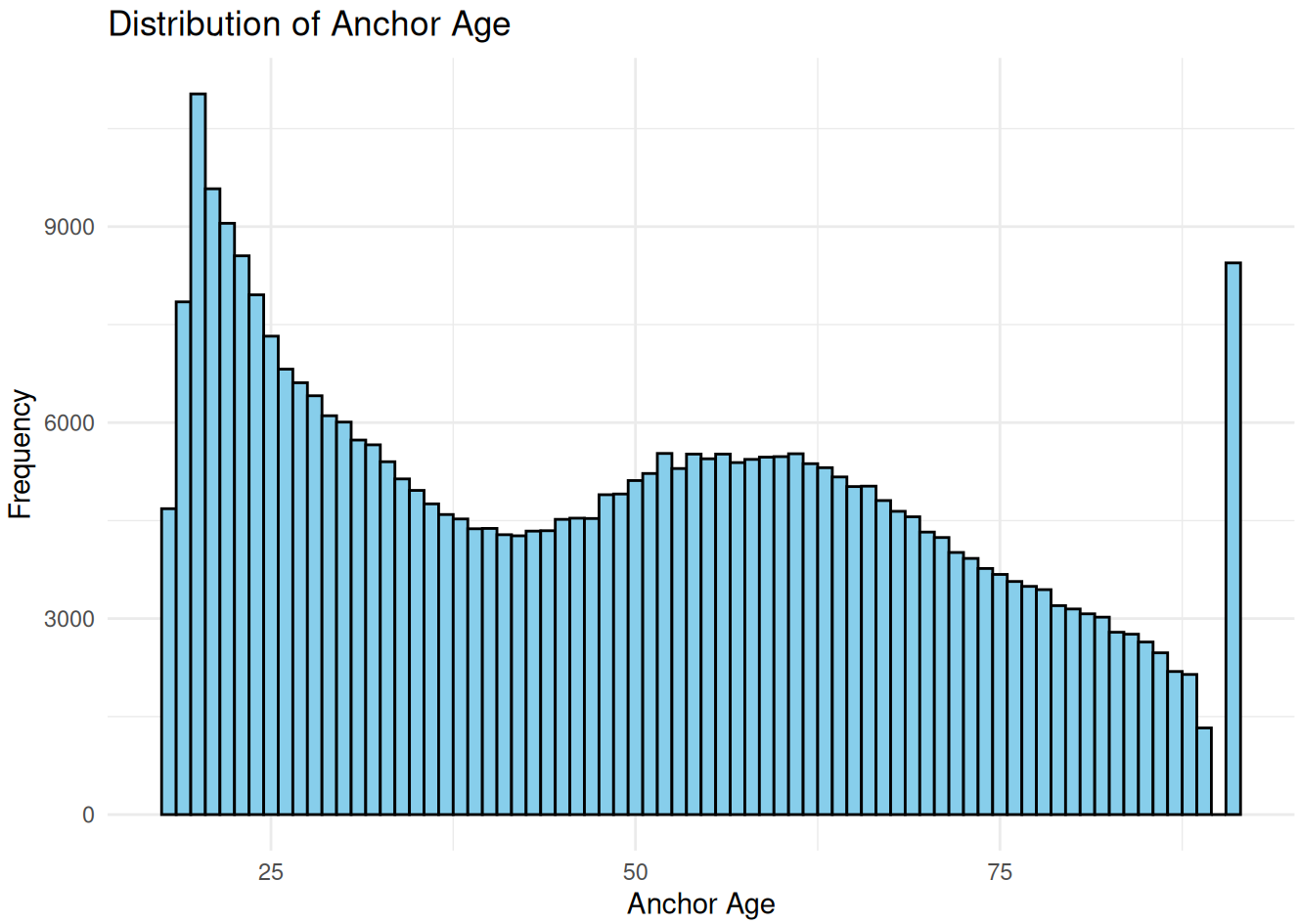
```
# Summary of gender as a bar chart
ggplot(patients_tble, aes(x = gender)) +
  geom_bar(fill = "steelblue") +
  labs(title = "Distribution of Gender", x = "Gender", y = "Count") +
  theme_minimal()
```



According to the bar chart of gender distribution, we can see a relatively balanced number of male and female subjects with slightly more females than males, suggesting that the data is representative across genders without a significant skew.

Step 3: create histogram and boxplot to summarize anchor_age

```
# Summary of anchor_age as a histogram
patients_tble %>% ggplot(aes(x = anchor_age)) +
geom_histogram(binwidth = 1, fill = "skyblue", color = "black") +
labs(title = "Distribution of Anchor Age",
     x = "Anchor Age",
     y = "Frequency") +
     theme_minimal()
```



According to the histogram, the frequency of anchor_age fluctuates across the ages, first peaks at around 20, then decreases and rises again at around 45 till drops again after 62 years old. A sudden peak appears at around 90. The distribution of data suggests that there different age groups are covered in the dataset. The reason why such distribution pattern occur could be that people at certain age ranges might require more hospital care. For instance, younger populations (around 18) might more likely to suffer from disease because of under developed immune systems and slowly become stronger so that less hospital cares are needed in later time. Once pass 50 years old, the older age populations will need more health carings since the body functions are reducing until they are not willing to go to hospital because of the trouble. A sudden peak in frequency at around 90 years old people might due to the reason that their lives are a criticle point that hospital cares are necessary.

```
# Summary of anchor_age group by gender as a boxplot
ggplot(patients_tble, aes(x = gender, y = anchor_age, fill = gender)) +
  geom_boxplot() +
  labs(title = "Age Distribution by Gender", x = "Gender", y = "Anchor Age") +
  theme_minimal()
```

According to the boxplot of age distribution by gender, both genders shows similar median anchor ages, with males have slightly older anchor age than females. The IQR range of male is slightly smaller than that of female. This suggests that some differences exist in the age that males or females need hospitabl care, showing that males and females are different in somatic functions and the needs for hosptital cares.

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,val
uenum,valueuom,ref_range_lower,ref_range_upper,flag,priority,comments
1,10000032,,2704548,50931,P69FQC,2180-03-23 11:51:00,2180-03-23
15:56:00,___,95,mg/dL,70,100,,ROUTINE,"IF FASTING, 70-100 NORMAL, >125 PROVISIONAL DIABETES."
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,"BENZODIAZEPINE IMMUNOASSAY SCREEN DOES NOT DETECT SOME
DRUGS,;INCLUDING LORAZEPAM, CLONAZEPAM, AND FLUNITRAZEPAM."
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,PRESUMPTIVELY POSITIVE.
8,10000032,,36092842,51090,P69FQC,2180-03-23 11:51:00,2180-03-23
16:00:00,NEG,,,,,,ROUTINE,METHADONE ASSAY DETECTS ONLY METHADONE (NOT OTHER OPIATES/OPIOIDS).
9,10000032,,36092842,51092,P69FQC,2180-03-23 11:51:00,2180-03-23
16:00:00,NEG,,,,,,ROUTINE,"OPIATE IMMUNOASSAY SCREEN DOES NOT DETECT SYNTHETIC OPIOIDS;SUCH AS
METHADONE, OXYCODONE, FENTANYL, BUPRENORPHINE, TRAMADOL,;NALOXONE, MEPERIDINE.  SEE ONLINE LAB
MANUAL FOR DETAILS."
```

`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.

```
> labevents_tble
# A tibble: 88,086 × 10
   subject_id  stay_id bicarbonate chloride creatinine glucose potassium sodium hematocrit   wbc
        <dbl>    <dbl>       <dbl>    <dbl>      <dbl>   <dbl>     <dbl>  <dbl>      <dbl> <dbl>
 1   10000032 39553978          25       95        0.7     102       6.7    126       41.1   6.9
 2   10000690 37081114          26      100        1        85       4.8    137       36.1   7.1
 3   10000980 39765666          21      109        2.3      89       3.9    144       27.3   5.3
 4   10001217 34592300          30      104        0.5      87       4.1    142       37.4   5.4
 5   10001217 37067082          22      108        0.6     112       4.2    142       38.1  15.7
 6   10001725 31205490          NA       98       NA        NA       4.1    139       NA    NA
 7   10001843 39698942          28       97        1.3     131       3.9    138       31.4  10.4
 8   10001884 37510196          30       88        1.1     141       4.5    130       39.7  12.2
 9   10002013 39060235          24      102        0.9     288       3.5    137       34.9   7.2
10   10002114 34672098          18       NA        3.1      95       6.5    125       34.3  16.8
# ℹ 88,076 more rows
# ℹ Use `print(n = ...)` to see more rows
```

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.

Solution: Step 1: Save the icustays.csv.gz, d_labitems.csv.gz (including the lab measurement that we are interested in), and labevents.csv.gz files as tibbles

```
# define the target items
target_items <- c(50912, 50971, 50983, 50902, 50882, 51221,  51301, 50931)

icustays_tble <- arrow::open_dataset("~/mimic/icu/icustays.csv.gz",
                                     format = "csv") %>%
  collect() %>%
  print(width = Inf)
```

```
# A tibble: 94,458 × 8
   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 07:00:00
 2 Medical Intensive Care Unit (MICU)              2150-11-02 11:37:00
 3 Medical Intensive Care Unit (MICU)              2189-06-27 01:42:00
 4 Surgical Intensive Care Unit (SICU)             2157-11-20 11:18:02
 5 Surgical Intensive Care Unit (SICU)             2157-12-19 07:42:24
 6 Medical/Surgical Intensive Care Unit (MICU/SICU) 2110-04-11 08:52:22
 7 Medical/Surgical Intensive Care Unit (MICU/SICU) 2134-12-05 10:50:03
 8 Medical Intensive Care Unit (MICU)              2131-01-10 20:20:05
 9 Cardiac Vascular Intensive Care Unit (CVICU)    2160-05-18 03:00:53
10 Coronary Care Unit (CCU)                        2162-02-17 15:30:00
   outtime              los
   <dttm>             <dbl>
 1 2180-07-23 16:50:47 0.410
 2 2150-11-06 09:03:17 3.89
 3 2189-06-27 13:38:27 0.498
 4 2157-11-21 14:08:00 1.12
 5 2157-12-20 06:27:41 0.948
 6 2110-04-12 16:59:56 1.34
 7 2134-12-06 06:38:26 0.825
 8 2131-01-20 00:27:30 9.17
 9 2160-05-19 10:33:33 1.31
10 2162-02-20 13:16:27 2.91
# ℹ 94,448 more rows
```

```
dlabitems_tble <- read_csv("~/mimic/hosp/d_labitems.csv.gz",
                           show_col_types = FALSE) %>%
                              filter(itemid %in% target_items) %>%
  mutate (itemid = as.integer(itemid)) %>% print(width = Inf)
```

```
# A tibble: 8 × 4
  itemid label             fluid category
   <int> <chr>             <chr> <chr>
1  50882 Bicarbonate       Blood Chemistry
2  50902 Chloride          Blood Chemistry
3  50912 Creatinine        Blood Chemistry
4  50931 Glucose           Blood Chemistry
5  50971 Potassium         Blood Chemistry
6  50983 Sodium            Blood Chemistry
7  51221 Hematocrit        Blood Hematology
8  51301 White Blood Cells Blood Hematology
```

Step 2: Save the labevents.csv.gz file as a tibble and filter the rows with the the items that we are interested in and the patients in icustays_tble

```
labevents_tble <- arrow::open_dataset('labevents_pq',  format = "parquet") %>%
  select(subject_id, itemid, storetime, valuenum) %>%
  filter(itemid %in% target_items) %>%
  filter(subject_id %in% icustays_tble$subject_id) %>%
  collect() %>%
  print(width = Inf)
```

```
# A tibble: 18,136,009 × 4
   subject_id itemid storetime           valuenum
        <int>  <int> <dttm>                 <dbl>
 1   10000032  50931 2180-03-23 08:56:00     95
 2   10000032  50882 2180-03-23 09:40:00     27
 3   10000032  50902 2180-03-23 09:40:00    101
 4   10000032  50912 2180-03-23 09:40:00      0.4
 5   10000032  50971 2180-03-23 09:40:00      3.7
 6   10000032  50983 2180-03-23 09:40:00    136
 7   10000032  51221 2180-03-23 08:19:00     45.4
 8   10000032  51301 2180-03-23 08:19:00      3
 9   10000032  51221 2180-05-06 15:42:00     42.6
10   10000032  51301 2180-05-06 15:42:00      5
# ℹ 18,135,999 more rows
```

Step 3: Finsh up the rest of the problem

```
# Define the file name for caching
cache_file <- "labevents_tble.rds"

# Check if cached file exists
if (file.exists(cache_file)) {
  labevents_tble <- read_rds(cache_file) %>% print(width = Inf)
```

```
} else {

  # Load ICU Stay Data
  icustays_tble <- arrow::open_dataset("~/mimic/icu/icustays.csv.gz",
                                       format = "csv") %>%
    select(subject_id, stay_id, intime) %>% collect() %>%
    mutate(intime = as.POSIXct(intime, format="%Y-%m-%d %H:%M:%S"))

  labevents_data <- arrow::open_dataset("labevents_pq", format = "parquet") %>%
    filter(itemid %in% target_items) %>%
    select(subject_id, itemid, storetime, valuenum)

  # Pre-filter to only ICU patients using Arrow before collecting
  icustay_subjects <- unique(icustays_tble$subject_id)
  labevents_filtered <- labevents_data %>%
    filter(subject_id %in% icustay_subjects) %>% collect() %>%
    mutate(storetime = as.POSIXct(storetime, format="%Y-%m-%d %H:%M:%S"))

  # Join ICU Stays FIRST to Prevent Many-to-Many
  labevents_joined <- labevents_filtered %>%
    left_join(icustays_tble, by = "subject_id") %>%
    filter(storetime <= intime) %>%
    group_by(subject_id, stay_id, itemid) %>%
    slice_max(order_by = storetime, n = 1) %>% ungroup() %>%
    select(-storetime, -intime)

  # Convert to Wide Format
  labevents_tble <- labevents_joined %>%
    pivot_wider(names_from = itemid, values_from = valuenum) %>%

    # Rename columns to meaningful names
    rename(creatinine = `50912`,
           potassium = `50971`,
           sodium = `50983`,
           chloride = `50902`,
           bicarbonate = `50882`,
           hematocrit = `51221`,
           white_blood_cell_count = `51301`,
           glucose = `50931`)

  write_rds(labevents_tble, cache_file)
  print(labevents_tble, width = Inf)
}
```

```
# A tibble: 88,087 × 10
# Groups:   subject_id, stay_id [88,087]
   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   10001217 37067082          22       108        0.6      112       4.2     142
 6   10001725 31205490          NA        98         NA       NA       4.1     139
 7   10001843 39698942          28        97        1.3      131       3.9     138
 8   10001884 37510196          30        88        1.1      141       4.5     130
 9   10002013 39060235          24       102        0.9      288       3.5     137
10   10002114 34672098          18        NA        3.1       95       6.5     125
   hematocrit white_blood_cell_count
        <dbl>                  <dbl>
 1       41.1                    6.9
 2       36.1                    7.1
 3       27.3                    5.3
 4       37.4                    5.4
 5       38.1                   15.7
 6       NA                     NA
 7       31.4                   10.4
 8       39.7                   12.2
 9       34.9                    7.2
10       34.3                   16.8
# ℹ 88,077 more rows
```

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,warnin
g
10000032,29079034,39553978,18704,2180-07-23 12:36:00,2180-07-23 14:45:00,226512,39.4,39.4,kg,0
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
Rhythm),,,0
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,0
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,highnormalvalue
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,90,140
220051,Arterial Blood Pressure diastolic,ABPd,chartevents,Routine Vital Signs,mmHg,Numeric,60,90
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.

```
> chartevents_tble
# A tibble: 94,424 × 7
   subject_id  stay_id heart_rate non_invasive_blood_pressure_systolic non_invasive_blood_pressure_diastolic respiratory_rate temperature_fahrenheit
        <int>    <dbl>      <dbl>                                <dbl>                                 <dbl>           <dbl>                 <dbl>
 1   10000032 39553978         91                                   84                                    48              24                  98.7
 2   10000690 37081114         79                                  107                                    63              23                  97.7
 3   10000980 39765666         77                                  150                                    77              23                  98
 4   10001217 34592300         96                                  167                                    95              11                  97.6
 5   10001217 37067082         86                                  151                                    90              18                  98.5
 6   10001725 31205490         55                                   73                                    56              19                  97.7
 7   10001843 39698942        118                                  112                                    71              17                  97.9
 8   10001884 37510196         38                                  180                                    12              10                  98.1
 9   10002013 39060235         80                                  104                                    70              14                  97.2
10   10002114 34672098        105                                  104                                    81              22                  97.9
# ℹ 94,414 more rows
# ℹ Use `print(n = ...)` to see more rows
```

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.

Solution: Step 1: Create the chartevents parquet file

```
chartevents <- arrow::open_dataset("~/mimic/icu/chartevents.csv",
                                   format = "csv")
arrow::write_dataset(chartevents, path = "chartevents_pq", format = "parquet")
dataset_parquet <- arrow::open_dataset ("chartevents_pq", format = "parquet")
```

Step 2: Create the d_item tibble

```
d_items_tble <- arrow::open_dataset("~/mimic/icu/d_items.csv.gz",
                                    format = "csv") %>%
  filter(itemid %in% c(220045, 220179, 220180, 223761, 220210)) %>%
  collect() %>%
  print(width = Inf)
```

```
# A tibble: 5 × 9
  itemid label                                abbreviation  linksto
   <int> <chr>                                <chr>         <chr>
1 220045 Heart Rate                           HR            chartevents
2 220179 Non Invasive Blood Pressure systolic  NBPs          chartevents
```

```
3 220180 Non Invasive Blood Pressure diastolic NBPd          chartevents
4 220210 Respiratory Rate                      RR            chartevents
5 223761 Temperature Fahrenheit                Temperature F chartevents
  category            unitname param_type lownormalvalue highnormalvalue
  <chr>               <chr>    <chr>              <int>           <dbl>
1 Routine Vital Signs bpm      Numeric               NA              NA
2 Routine Vital Signs mmHg     Numeric               NA              NA
3 Routine Vital Signs mmHg     Numeric               NA              NA
4 Respiratory         insp/min Numeric               NA              NA
5 Routine Vital Signs °F       Numeric               NA              NA
```

Step 3: Define and rename vital items

```
vital_item_ids <- c(220045, 220179, 220180, 223761, 220210)
vital_items <- setNames(
  c("heart_rate",
    "systolic_non_invasive_blood_pressure",
    "diastolic_non_invasive_blood_pressure",
    "temperature_fahrenheit",
    "respiratory_rate"),
  vital_item_ids
)
```

Step 4: Load and process data, use storetime instead of charttime as required, and apply the average vital measurements

```
chartevents_data <- open_dataset("chartevents_pq", format = "parquet") %>%
  to_duckdb() %>%
  select(subject_id, itemid, storetime, valuenum) %>%
  filter(itemid %in% vital_item_ids) %>%
  left_join(
    select(icustays_tble, subject_id, stay_id, intime, outtime),
    by = "subject_id", copy = TRUE) %>%
  filter(storetime >= intime & storetime <= outtime) %>%
  group_by(subject_id, stay_id, itemid, storetime) %>%

  # Apply average valuenum
  mutate(valuenum = mean(valuenum, na.rm = TRUE)) %>%
  ungroup() %>%
  group_by(subject_id, stay_id, itemid) %>%
  slice_min(order_by = storetime, n = 1) %>%
  select(-storetime, -intime) %>%
  ungroup() %>% pivot_wider(names_from = itemid, values_from = valuenum) %>%
  rename_with(~ recode(., !!!vital_items)) %>%
  collect()
```

Step 5: Create and print the chartevents_tble

```
chartevents_tble <- chartevents_data %>%
  select(
```

```
    subject_id,
    stay_id,
    heart_rate,
    diastolic_non_invasive_blood_pressure,
    systolic_non_invasive_blood_pressure,
    respiratory_rate,
    temperature_fahrenheit
  ) %>%
  arrange(subject_id, stay_id) %>%
  as_tibble()

print(chartevents_tble)
```

```
# A tibble: 94,364 × 7
   subject_id  stay_id heart_rate diastolic_non_invasiv…¹ systolic_non_invasiv…²
        <dbl>    <int>      <dbl>                   <dbl>                   <dbl>
 1   10000032 39553978       91                     48                      84
 2   10000690 37081114       78                     56.5                   106
 3   10000980 39765666       76                    102                     154
 4   10001217 34592300       79.3                   93.3                   156
 5   10001217 37067082       86                     90                     151
 6   10001725 31205490       86                     56                      73
 7   10001843 39698942      124.                    78                     110
 8   10001884 37510196       49                     30.5                   174.
 9   10002013 39060235       80                     62                      98.5
10   10002114 34672098      110.                    80                     112
# ℹ 94,354 more rows
# ℹ abbreviated names: ¹diastolic_non_invasive_blood_pressure,
#   ²systolic_non_invasive_blood_pressure
# ℹ 2 more variables: respiratory_rate <dbl>, temperature_fahrenheit <dbl>
```

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.

```
> mimic_icu_cohort
# A tibble: 94,458 × 41
   subject_id  hadm_id  stay_id first_careunit        last_careunit intime              outtime              los admittime           dischtime           deathtime
        <dbl>    <dbl>    <dbl> <chr>                 <chr>         <dttm>              <dttm>             <dbl> <dttm>              <dttm>              <dttm>
 1   10000032 29079034 39553978 Medical Intensive Car… Medical Inte… 2180-07-23 14:00:00 2180-07-23 23:50:47 0.410 2180-07-23 12:35:00 2180-07-25 17:55:00 NA
 2   10000690 25860671 37081114 Medical Intensive Car… Medical Inte… 2150-11-02 19:37:00 2150-11-06 17:03:17 3.89  2150-11-02 18:02:00 2150-11-12 13:45:00 NA
 3   10000980 26913865 39765666 Medical Intensive Car… Medical Inte… 2189-06-27 08:42:00 2189-06-27 20:38:27 0.498 2189-06-27 07:38:00 2189-07-03 03:00:00 NA
 4   10001217 24597018 37067082 Surgical Intensive Ca… Surgical Int… 2157-11-20 19:18:02 2157-11-21 22:08:00 1.12  2157-11-18 22:56:00 2157-11-25 18:00:00 NA
 5   10001217 27703517 34592300 Surgical Intensive Ca… Surgical Int… 2157-12-19 15:42:24 2157-12-20 14:27:41 0.948 2157-12-18 16:58:00 2157-12-24 14:55:00 NA
 6   10001725 25563031 31205490 Medical/Surgical Inte… Medical/Surg… 2110-04-11 15:52:22 2110-04-12 23:59:56 1.34  2110-04-11 15:08:00 2110-04-14 15:00:00 NA
 7   10001843 26133978 39698942 Medical/Surgical Inte… Medical/Surg… 2134-12-05 18:50:03 2134-12-06 14:38:26 0.825 2134-12-05 00:10:00 2134-12-06 12:54:00 2134-12-06 12:54:00
 8   10001884 26184834 37510196 Medical Intensive Car… Medical Inte… 2131-01-11 04:20:05 2131-01-20 08:27:30 9.17  2131-01-07 20:39:00 2131-01-20 05:15:00 2131-01-20 05:15:00
 9   10002013 23581541 39060235 Cardiac Vascular Inte… Cardiac Vasc… 2160-05-18 10:00:53 2160-05-19 17:33:33 1.31  2160-05-18 07:45:00 2160-05-23 13:30:00 NA
10   10002114 27793700 34672098 Coronary Care Unit (C… Coronary Car… 2162-02-17 23:30:00 2162-02-20 21:16:27 2.91  2162-02-17 22:32:00 2162-03-04 15:16:00 NA
# ℹ 94,448 more rows
# ℹ 30 more variables: admission_type <chr>, admit_provider_id <chr>, admission_location <chr>, discharge_location <chr>, insurance <chr>, language <chr>,
#   marital_status <chr>, race <chr>, edregtime <dttm>, edouttime <dttm>, hospital_expire_flag <dbl>, gender <chr>, anchor_age <dbl>, anchor_year <dbl>,
#   anchor_year_group <chr>, dod <date>, bicarbonate <dbl>, chloride <dbl>, creatinine <dbl>, glucose <dbl>, potassium <dbl>, sodium <dbl>, hematocrit <dbl>, wbc <dbl>,
#   heart_rate <dbl>, non_invasive_blood_pressure_systolic <dbl>, non_invasive_blood_pressure_diastolic <dbl>, respiratory_rate <dbl>, temperature_fahrenheit <dbl>,
#   age_intime <dbl>
# ℹ Use `print(n = ...)` to see more rows
```

Solution: Step 1: List all the tbles that needed for the question. Since labevents_tble and chartevents_tble are being processed by specific requirements in previous questions, they are not listed here again.

```
icustays_tble <- arrow::open_dataset("~/mimic/icu/icustays.csv.gz",
format = "csv") %>% collect() %>% print(width = Inf)
```

```
# A tibble: 94,458 × 8
   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 07:00:00
 2 Medical Intensive Care Unit (MICU)            2150-11-02 11:37:00
 3 Medical Intensive Care Unit (MICU)            2189-06-27 01:42:00
 4 Surgical Intensive Care Unit (SICU)           2157-11-20 11:18:02
 5 Surgical Intensive Care Unit (SICU)           2157-12-19 07:42:24
 6 Medical/Surgical Intensive Care Unit (MICU/SICU) 2110-04-11 08:52:22
 7 Medical/Surgical Intensive Care Unit (MICU/SICU) 2134-12-05 10:50:03
 8 Medical Intensive Care Unit (MICU)            2131-01-10 20:20:05
 9 Cardiac Vascular Intensive Care Unit (CVICU)  2160-05-18 03:00:53
10 Coronary Care Unit (CCU)                      2162-02-17 15:30:00
   outtime               los
   <dttm>              <dbl>
 1 2180-07-23 16:50:47 0.410
 2 2150-11-06 09:03:17 3.89
 3 2189-06-27 13:38:27 0.498
 4 2157-11-21 14:08:00 1.12
 5 2157-12-20 06:27:41 0.948
 6 2110-04-12 16:59:56 1.34
```

```
 7 2134-12-06 06:38:26 0.825
 8 2131-01-20 00:27:30 9.17
 9 2160-05-19 10:33:33 1.31
10 2162-02-20 13:16:27 2.91
# ℹ 94,448 more rows
```

```
admissions_tble <- arrow::open_dataset("~/mimic/hosp/admissions.csv.gz",
format = "csv") %>% collect() %>% print(width = Inf)
```

```
# A tibble: 546,028 × 16
   subject_id  hadm_id admittime           dischtime           deathtime
        <int>    <int> <dttm>              <dttm>              <dttm>
 1   10000032 22595853 2180-05-06 15:23:00 2180-05-07 10:15:00 NA
 2   10000032 22841357 2180-06-26 11:27:00 2180-06-27 11:49:00 NA
 3   10000032 25742920 2180-08-05 16:44:00 2180-08-07 10:50:00 NA
 4   10000032 29079034 2180-07-23 05:35:00 2180-07-25 10:55:00 NA
 5   10000068 25022803 2160-03-03 15:16:00 2160-03-03 22:26:00 NA
 6   10000084 23052089 2160-11-20 17:56:00 2160-11-25 06:52:00 NA
 7   10000084 29888819 2160-12-27 21:11:00 2160-12-28 08:07:00 NA
 8   10000108 27250926 2163-09-27 16:17:00 2163-09-28 02:04:00 NA
 9   10000117 22927623 2181-11-14 18:05:00 2181-11-15 06:52:00 NA
10   10000117 27988844 2183-09-18 11:10:00 2183-09-21 09:30:00 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         ""
 6 EW EMER.          P42H7G            WALK-IN/SELF REFERRAL  "HOME HEALTH CARE"
 7 EU OBSERVATION    P35NE4            PHYSICIAN REFERRAL     ""
 8 EU OBSERVATION    P40JML            EMERGENCY ROOM         ""
 9 EU OBSERVATION    P47EY8            EMERGENCY ROOM         ""
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 12:17:00
 2 "Medicaid" English  WIDOWED        WHITE 2180-06-26 08:54:00
 3 "Medicaid" English  WIDOWED        WHITE 2180-08-05 13:58:00
 4 "Medicaid" English  WIDOWED        WHITE 2180-07-22 22:54:00
 5 ""         English  SINGLE         WHITE 2160-03-03 13:55:00
 6 "Medicare" English  MARRIED        WHITE 2160-11-20 12:36:00
 7 "Medicare" English  MARRIED        WHITE 2160-12-27 10:32:00
 8 ""         English  SINGLE         WHITE 2163-09-27 09:18:00
 9 "Medicaid" English  DIVORCED       WHITE 2181-11-14 13:51:00
10 "Medicaid" English  DIVORCED       WHITE 2183-09-18 01:41:00
   edouttime           hospital_expire_flag
   <dttm>                             <int>
 1 2180-05-06 16:30:00                    0
 2 2180-06-26 14:31:00                    0
```

```
 3 2180-08-05 18:44:00                   0
 4 2180-07-23 07:00:00                   0
 5 2160-03-03 22:26:00                   0
 6 2160-11-20 19:20:00                   0
 7 2160-12-28 08:07:00                   0
 8 2163-09-28 02:04:00                   0
 9 2181-11-15 01:57:00                   0
10 2183-09-18 13:20:00                   0
# ℹ 546,018 more rows
```

```
patients_tble <- arrow::open_dataset("~/mimic/hosp/patients.csv.gz",
format = "csv") %>% collect() %>% print(width = Inf)
```

```
# A tibble: 364,627 × 6
   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 364,617 more rows
```

Step 2: Create the required mimic_icu_cohort tibble

```
mimic_icu_cohort <- icustays_tble %>%
  left_join(admissions_tble, by = c("subject_id", "hadm_id")) %>%
  left_join(patients_tble, by = "subject_id") %>%
  left_join(labevents_tble, by = c("subject_id", "stay_id")) %>%
  left_join(chartevents_tble, by = c("subject_id", "stay_id")) %>%
  mutate(age_intime = year(intime) - anchor_year + anchor_age) %>%
  filter(age_intime >= 18)

print(mimic_icu_cohort)
```

```
# A tibble: 94,458 × 41
   subject_id  hadm_id  stay_id first_careunit last_careunit intime
        <dbl>    <int>    <int> <chr>          <chr>         <dttm>
 1   10000032 29079034 39553978 Medical Inten… Medical Inte… 2180-07-23 07:00:00
 2   10000690 25860671 37081114 Medical Inten… Medical Inte… 2150-11-02 11:37:00
 3   10000980 26913865 39765666 Medical Inten… Medical Inte… 2189-06-27 01:42:00
 4   10001217 24597018 37067082 Surgical Inte… Surgical Int… 2157-11-20 11:18:02
 5   10001217 27703517 34592300 Surgical Inte… Surgical Int… 2157-12-19 07:42:24
 6   10001725 25563031 31205490 Medical/Surgi… Medical/Surg… 2110-04-11 08:52:22
 7   10001843 26133978 39698942 Medical/Surgi… Medical/Surg… 2134-12-05 10:50:03
```

```
 8   10001884 26184834 37510196 Medical Inten… Medical Inte… 2131-01-10 20:20:05
 9   10002013 23581541 39060235 Cardiac Vascu… Cardiac Vasc… 2160-05-18 03:00:53
10   10002114 27793700 34672098 Coronary Care… Coronary Car… 2162-02-17 15:30:00
# ℹ 94,448 more rows
# ℹ 35 more variables: outtime <dttm>, los <dbl>, admittime <dttm>,
#   dischtime <dttm>, deathtime <dttm>, admission_type <chr>,
#   admit_provider_id <chr>, admission_location <chr>,
#   discharge_location <chr>, insurance <chr>, language <chr>,
#   marital_status <chr>, race <chr>, edregtime <dttm>, edouttime <dttm>,
#   hospital_expire_flag <int>, gender <chr>, anchor_age <int>, …
```

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

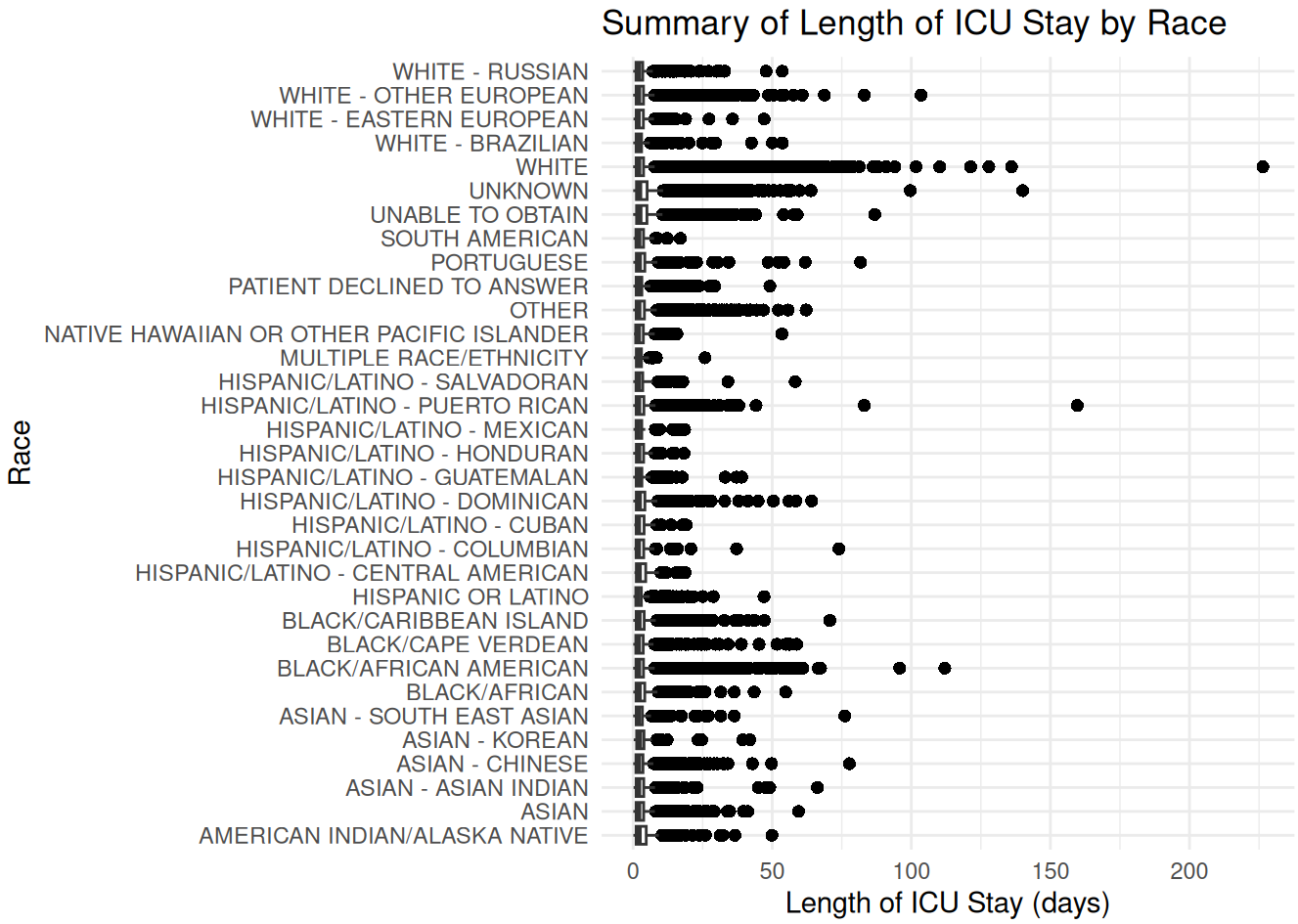
Solution: (8.1) Length of ICU stay `los` vs demographic variables (race, insurance, marital_status, gender, age at intime): Step 1: Length of ICU stay los vs race

```
# show summary statistics
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los)) %>%
  group_by(race) %>%
  summarise(
    count = n(),
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    min_los = min(los, na.rm = TRUE),
    max_los = max(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
    iqr_los = IQR(los, na.rm = TRUE)
  )
```

```
# A tibble: 33 × 8
   race                count mean_los median_los min_los max_los sd_los iqr_los
   <chr>               <int>    <dbl>      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
 1 AMERICAN INDIAN/ALA…   198     4.31       2.08 0.0879     49.9   6.48    3.61
 2 ASIAN                 1095     3.56       1.92 0.0450     59.5   4.95    2.70
 3 ASIAN - ASIAN INDIAN   248     4.08       1.90 0.160      66.3   7.28    2.73
 4 ASIAN - CHINESE       1061     3.59       1.89 0.0206     77.7   5.47    2.52
 5 ASIAN - KOREAN          73     4.44       2.25 0.211      41.9   7.42    2.64
```

```
 6 ASIAN - SOUTH EAST …   408      3.45       1.86 0.00939    76.1   5.72    2.22
 7 BLACK/AFRICAN          431      4.01       2.08 0.0345     54.8   5.80    3.12
 8 BLACK/AFRICAN AMERI…  8677      3.54       1.90 0.00346   112.    5.42    2.69
 9 BLACK/CAPE VERDEAN     656      3.67       1.83 0.0296     58.8   6.32    2.67
10 BLACK/CARIBBEAN ISL…   621      4.34       2.04 0.0445     70.7   6.97    2.87
# ℹ 23 more rows
```

```r
# plot the plot
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(race)) %>%
  ggplot(aes(x = los, y = race)) +
  geom_boxplot(outlier.colour = "black", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Summary of Length of ICU Stay by Race",
       x = "Length of ICU Stay (days)", y = "Race") +
  theme_minimal()
```



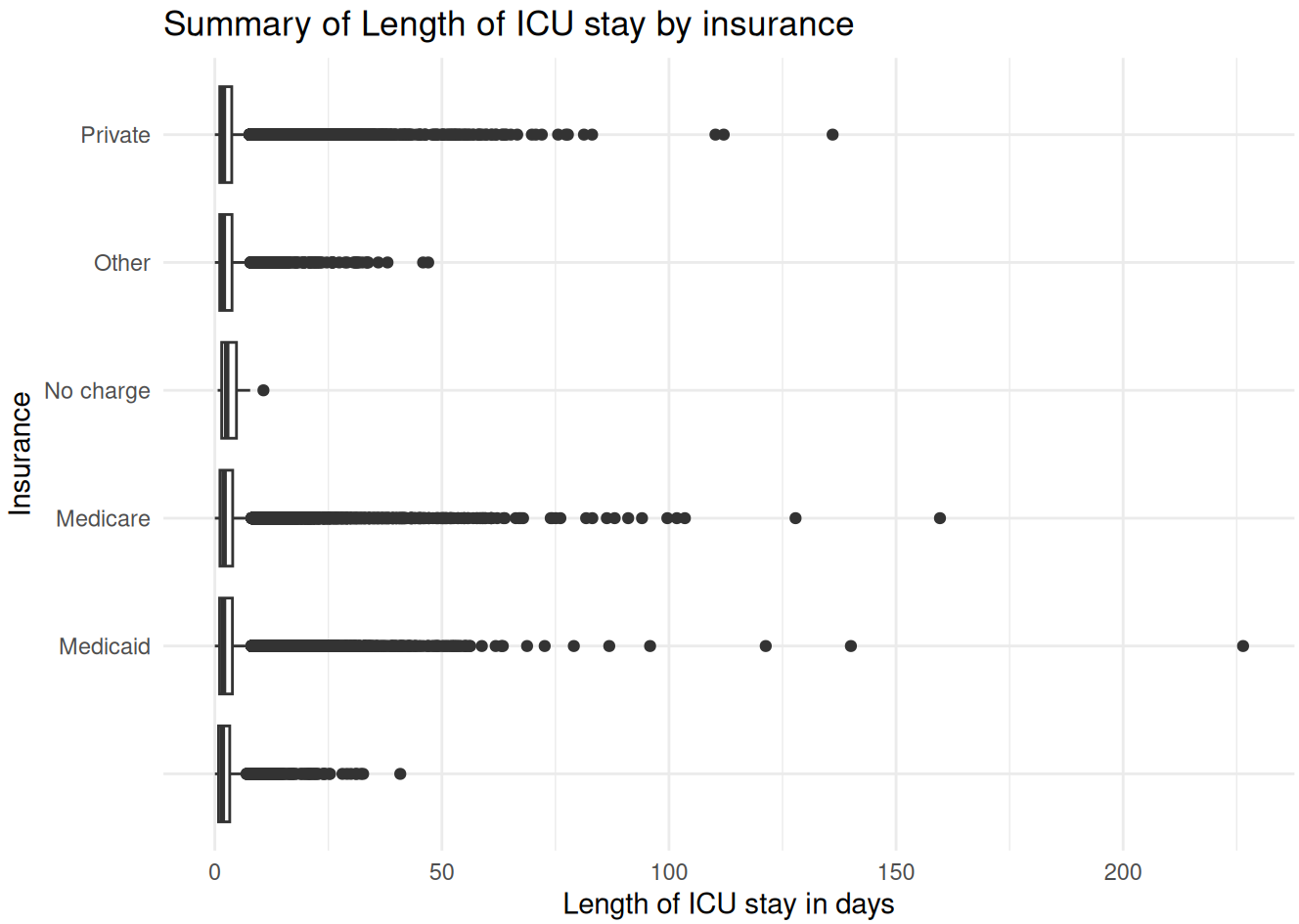
Step 2: Length of ICU stay los vs insurance

```r
# show summary statistics
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los)) %>%
  group_by(insurance) %>%
  summarise(
    count = n(),
```

```
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    min_los = min(los, na.rm = TRUE),
    max_los = max(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
    iqr_los = IQR(los, na.rm = TRUE)
  )
```

```
# A tibble: 6 × 8
  insurance   count mean_los median_los min_los max_los sd_los iqr_los
  <chr>       <int>    <dbl>      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
1 ""           1523     3.21       1.65 0.0230     40.8   4.52    2.47
2 "Medicaid"  14237     3.79       1.90 0.00242   226.    6.21    2.82
3 "Medicare"  51815     3.60       2.03 0.00145   160.    5.10    2.81
4 "No charge"     8     3.87       2.60 0.626      10.7   3.55    3.26
5 "Other"      2328     3.39       1.86 0.00745    47.0   4.56    2.71
6 "Private"   24533     3.64       1.88 0.00125   136.    5.64    2.64
```

```
# get the plot
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los)) %>%
  ggplot(aes(x = los, y = insurance)) +
  geom_boxplot() +
  labs(title = "Summary of Length of ICU stay by insurance",
       x = "Length of ICU stay in days", y = "Insurance") +
  theme_minimal()
```

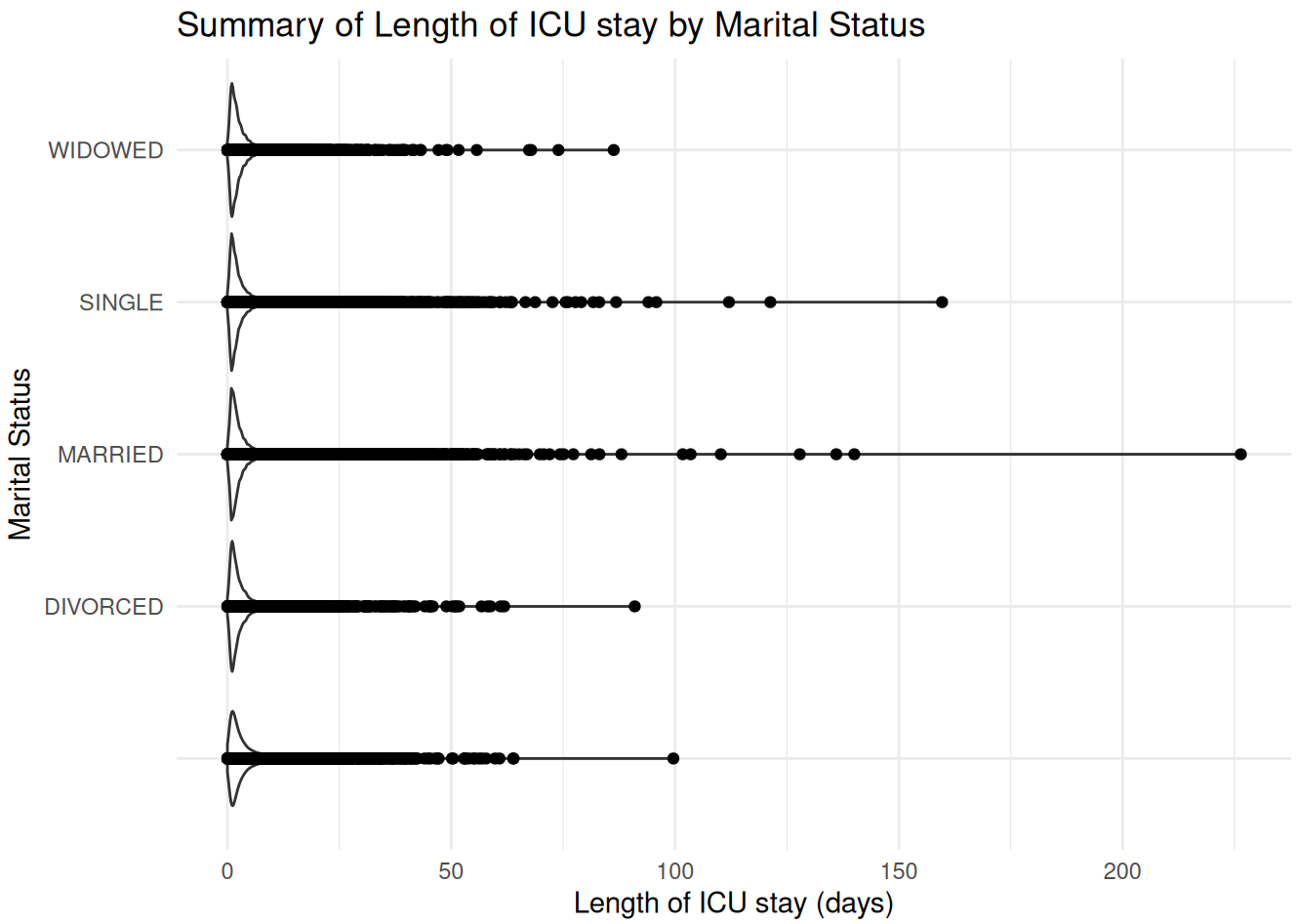


Step 3: Length of ICU stay los vs marital_status

```
# show summary statistics:
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(marital_status)) %>%
  group_by(marital_status) %>%
  summarise(
    count = n(),
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    min_los = min(los, na.rm = TRUE),
    max_los = max(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
    iqr_los = IQR(los, na.rm = TRUE)
  )
```

```
# A tibble: 5 × 8
  marital_status count mean_los median_los min_los max_los sd_los iqr_los
  <chr>          <int>    <dbl>      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
1 ""              7756     4.64       2.33 0.0144     99.6   6.40    3.98
2 "DIVORCED"      6932     3.58       1.95 0.0075     91.0   5.16    2.78
3 "MARRIED"      41901     3.59       1.97 0.00125   226.    5.44    2.68
4 "SINGLE"       26784     3.59       1.91 0.00227   160.    5.49    2.75
5 "WIDOWED"      11071     3.18       1.93 0.00733    86.3   4.22    2.50
```

```
# group by marital_status and plot the plot
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(marital_status)) %>%
  group_by(marital_status) %>%
  ggplot(aes(x = los, y = marital_status)) + geom_violin() + geom_point() +
  labs(title = "Summary of Length of ICU stay by Marital Status",
       x = "Length of ICU stay (days)", y = "Marital Status") +
  theme_minimal()
```



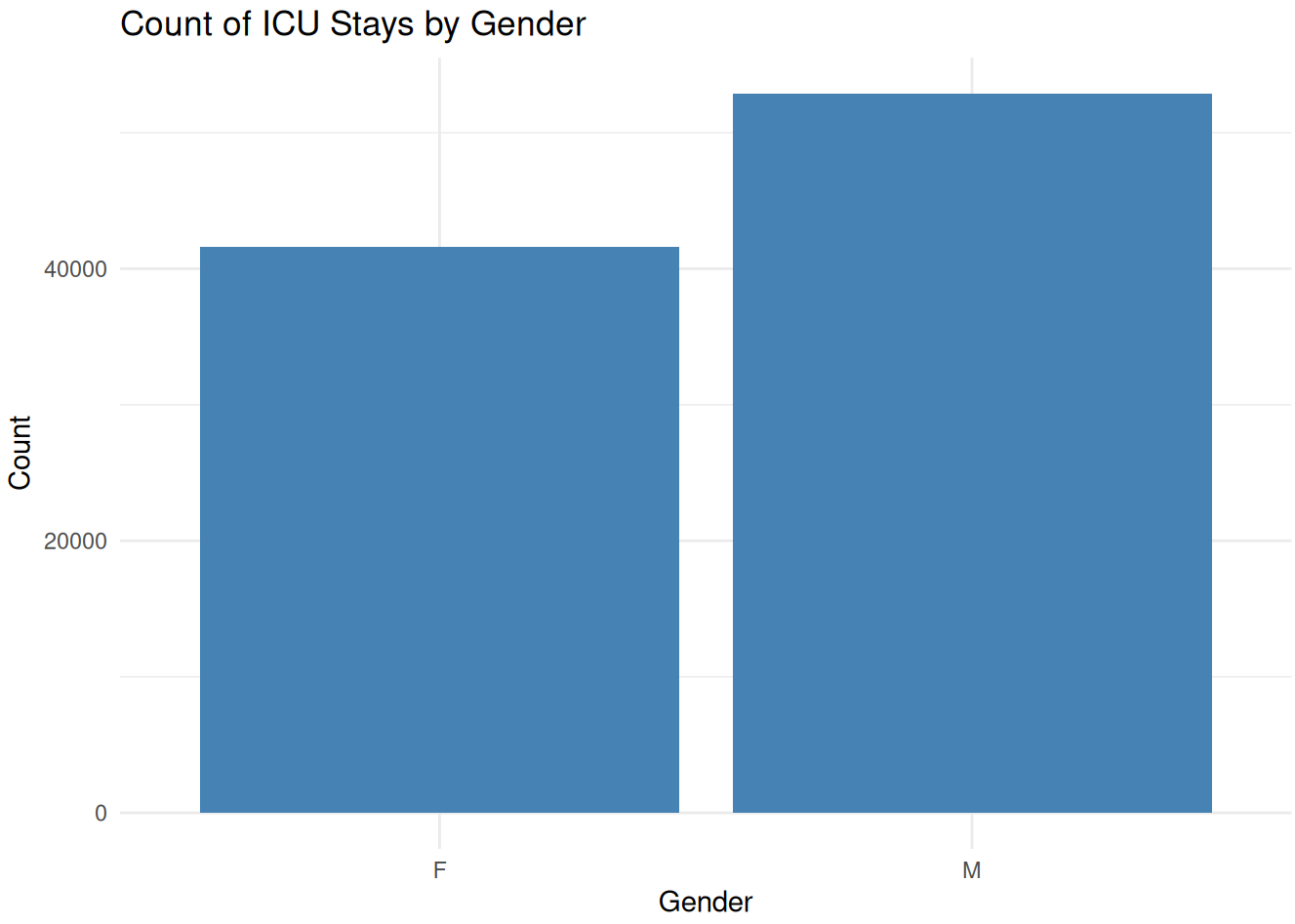
Step 4: Length of ICU stay los vs gender

```
# Remove missing values and compute summary statistics
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(gender)) %>%
  group_by(gender) %>%
  summarise(
    count = n(),
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    min_los = min(los, na.rm = TRUE),
    max_los = max(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
```

```
    iqr_los = IQR(los, na.rm = TRUE)
  )
```

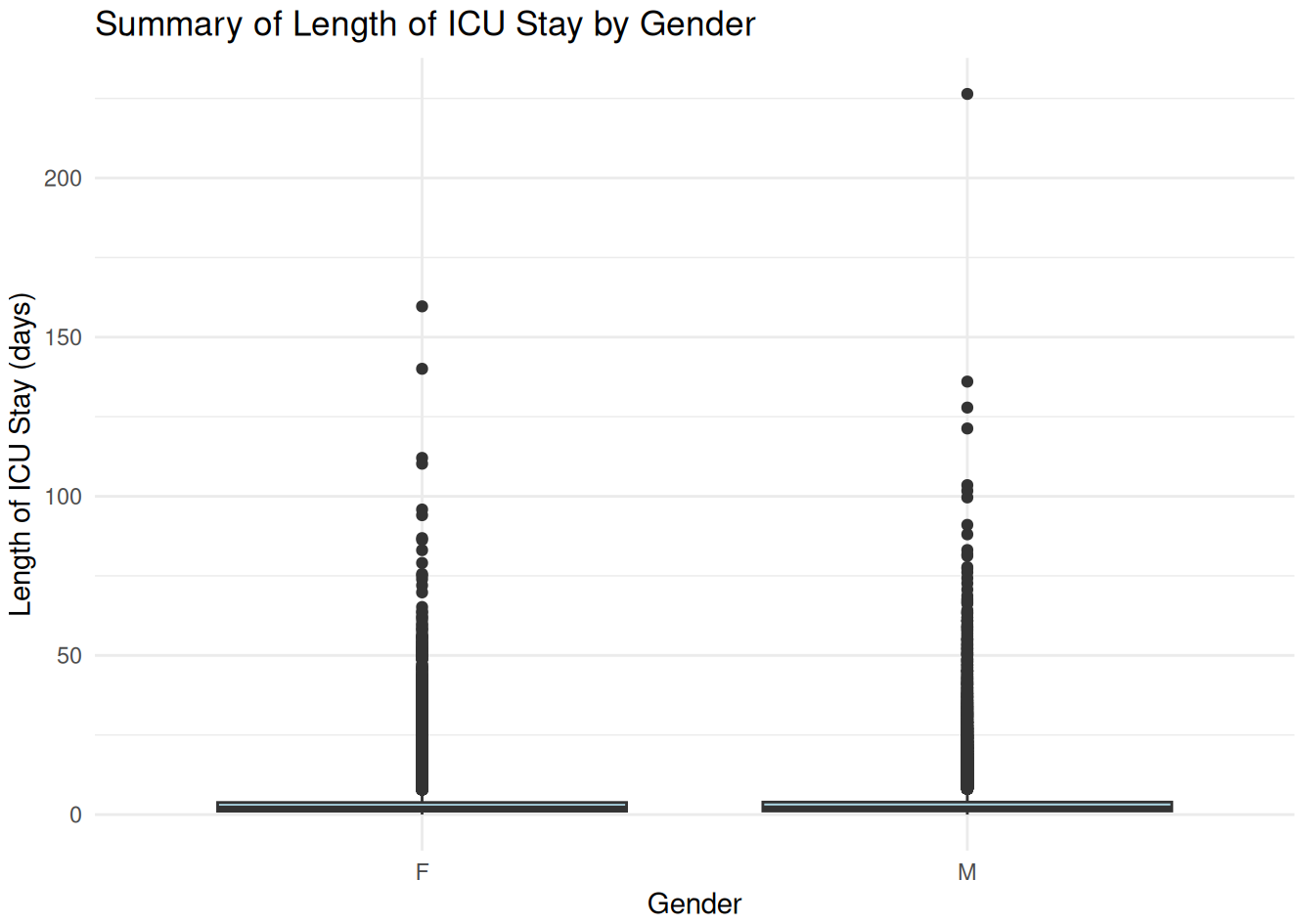
```
# A tibble: 2 × 8
  gender count mean_los median_los min_los max_los sd_los iqr_los
  <chr>  <int>    <dbl>      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
1 F      41577     3.51       1.94 0.00145    160.   5.17    2.73
2 M      52867     3.72       1.98 0.00125    226.   5.58    2.80
```

```
# Plot count of ICU stays by gender
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(gender)) %>%
  ggplot(aes(x = gender)) +
  geom_bar(fill = "steelblue") +
  labs(title = "Count of ICU Stays by Gender",
       x = "Gender",
       y = "Count") +
  theme_minimal()
```




```
# Plot boxplot of LOS by gender
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(gender)) %>%
  ggplot(aes(x = gender, y = los)) +
```

```
  geom_boxplot(fill = "lightblue") +
  labs(title = "Summary of Length of ICU Stay by Gender",
       x = "Gender",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```



Step 5: Length of ICU stay los vs age at intime

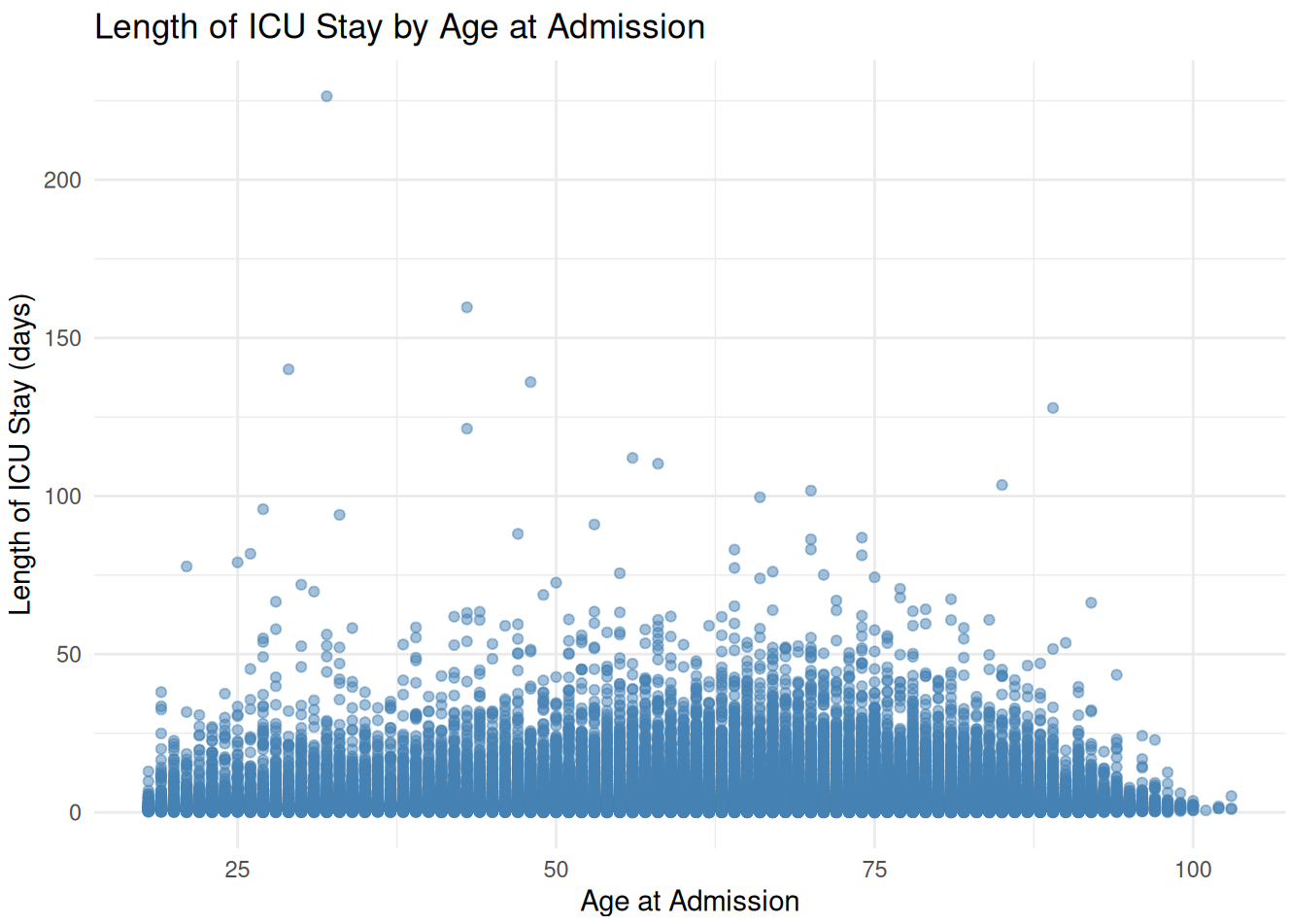
```
# Show summary statistics for LOS grouped by Age at Admission
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(age_intime)) %>%
  group_by(age_intime) %>%
  summarise(
    count = n(),
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    min_los = min(los, na.rm = TRUE),
    max_los = max(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
    iqr_los = IQR(los, na.rm = TRUE)
  )
```

```
# A tibble: 86 × 8
   age_intime count mean_los median_los min_los max_los sd_los iqr_los
```

	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	18	107	1.96	1.35	0.202	12.9	1.91	1.80
2	19	159	3.60	1.81	0.184	38.0	5.71	2.52
3	20	285	2.70	1.39	0.0820	22.6	3.67	1.89
4	21	297	2.85	1.58	0.0464	77.7	5.51	1.83
5	22	305	2.74	1.46	0.0784	30.8	3.99	1.90
6	23	335	3.36	1.60	0.0608	27.0	4.47	2.55
7	24	304	3.17	1.70	0.122	37.5	4.80	2.17
8	25	357	3.39	1.62	0.145	79.0	6.08	2.12
9	26	388	3.16	1.67	0.0345	81.8	6.01	2.08
10	27	338	4.67	1.98	0.0575	95.8	8.92	2.98

ℹ 76 more rows

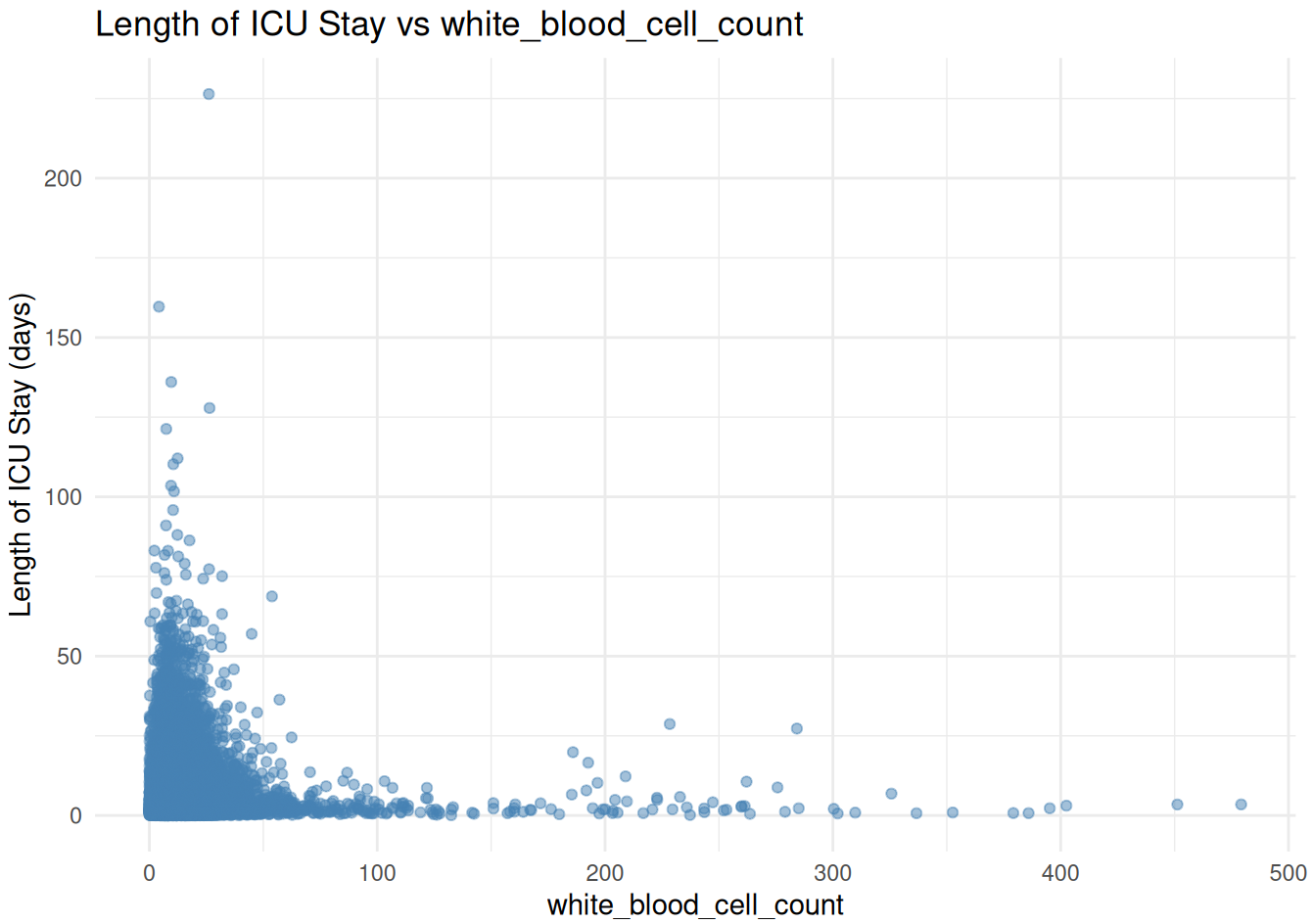
```
# Plot the plot
mimic_icu_cohort %>%
  filter(!is.na(los) & is.finite(los) & !is.na(age_intime)) %>%
  ggplot(aes(x = age_intime, y = los)) +
  geom_point(alpha = 0.5, color = "steelblue") +
  labs(title = "Length of ICU Stay by Age at Admission",
       x = "Age at Admission", y = "Length of ICU Stay (days)") +
  theme_minimal()
```



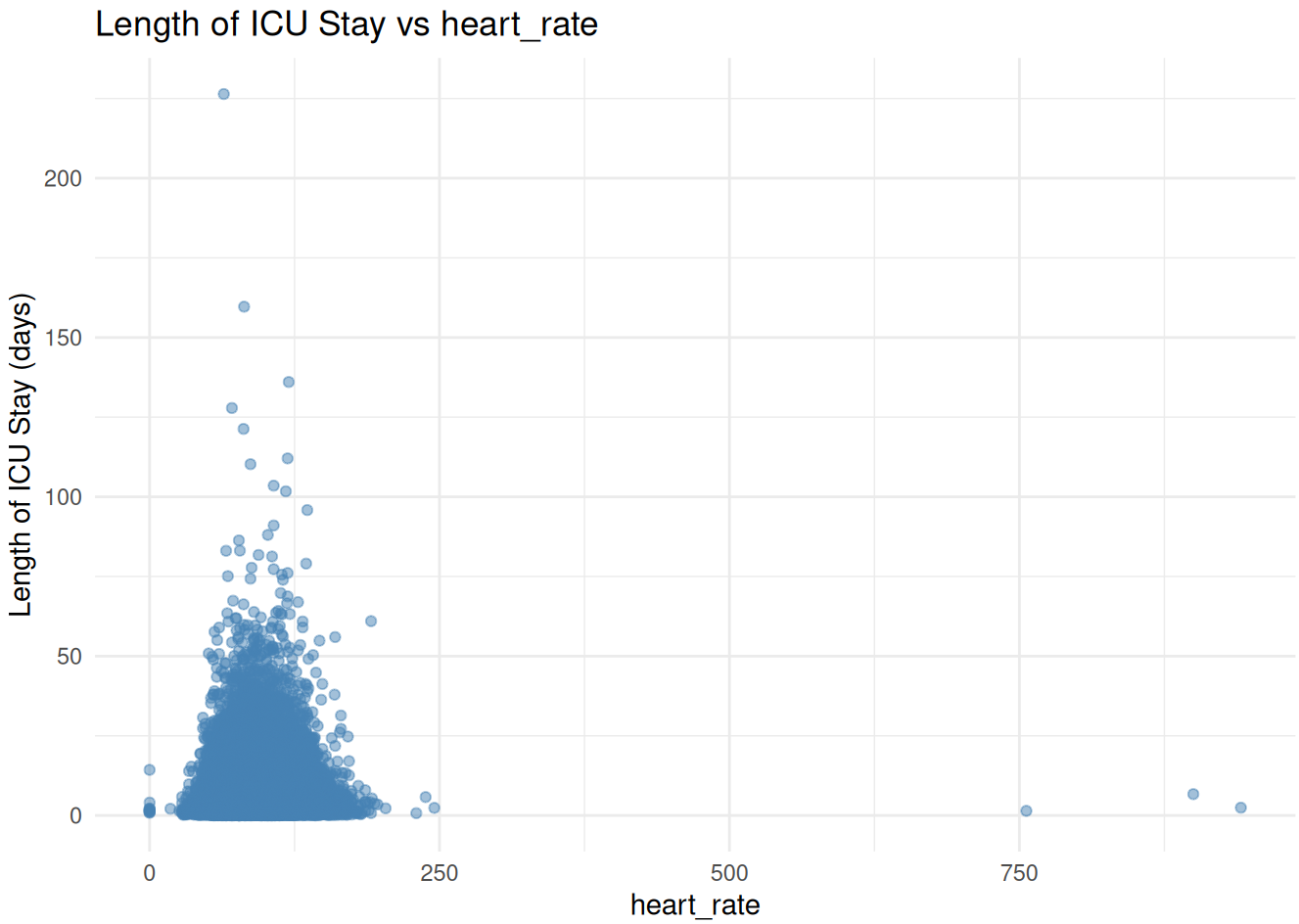
(8.2) Length of ICU stay `los` vs the last available lab measurements before ICU stay

```r
# get the variables which will be used to plot
variables <- c("white_blood_cell_count", "heart_rate",
    "systolic_non_invasive_blood_pressure",
    "diastolic_non_invasive_blood_pressure",
    "temperature_fahrenheit",
    "respiratory_rate", "potassium", "sodium",
               "chloride", "bicarbonate", "hematocrit",
               "white_blood_cell_count", "glucose" )


# get the last lab measurements before ICU stay
mimic_icu_cohort_variables <- mimic_icu_cohort %>%
  select(all_of(variables), los) %>%
  filter(if_all(c(all_of(variables), los), ~ !is.na(.))) %>%
  mutate(across(c(all_of(variables), los),
                ~ trim(., trim_proportion = 1, na.rm = TRUE)))

plots <- list()

# Drop missing values before plotting
mimic_icu_cohort_filtered <- mimic_icu_cohort %>%
  filter(if_all(all_of(variables), ~ !is.na(.)) & !is.na(los))

# Generate scatter plots for each variable
for (variable in variables) {
  plots[[variable]] <- mimic_icu_cohort_filtered %>%
    ggplot(aes(x = !!sym(variable), y = los)) +
    geom_point(alpha = 0.5, color = "steelblue") +
    labs(title = paste("Length of ICU Stay vs", variable),
         x = variable,
         y = "Length of ICU Stay (days)") +
    theme_minimal()
}

print(plots)
```

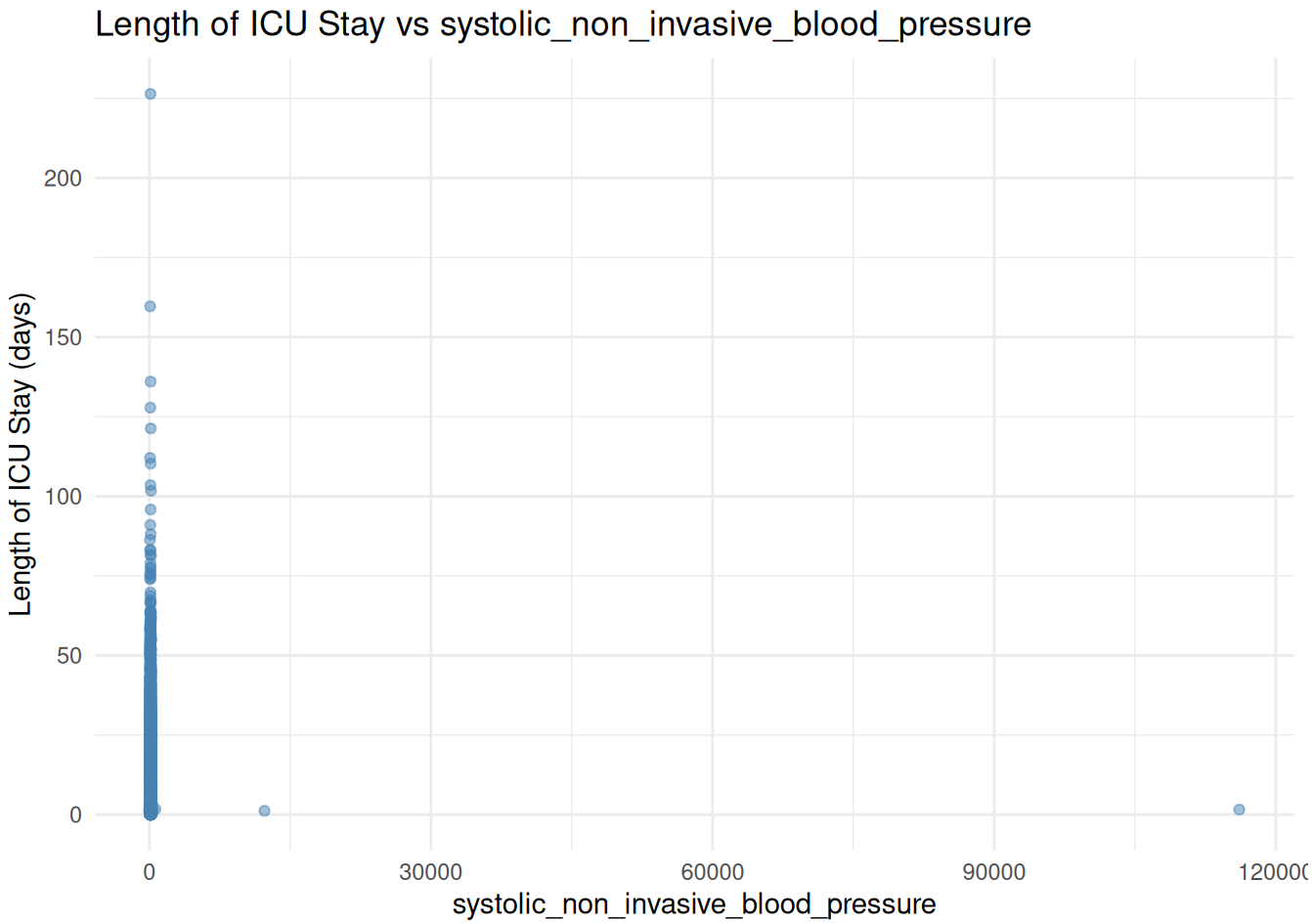
```
$white_blood_cell_count
```


Length of ICU Stay vs white_blood_cell_count
Length of ICU Stay (days)
0
50
100
150
200
0
100
200
300
400
500
white_blood_cell_count

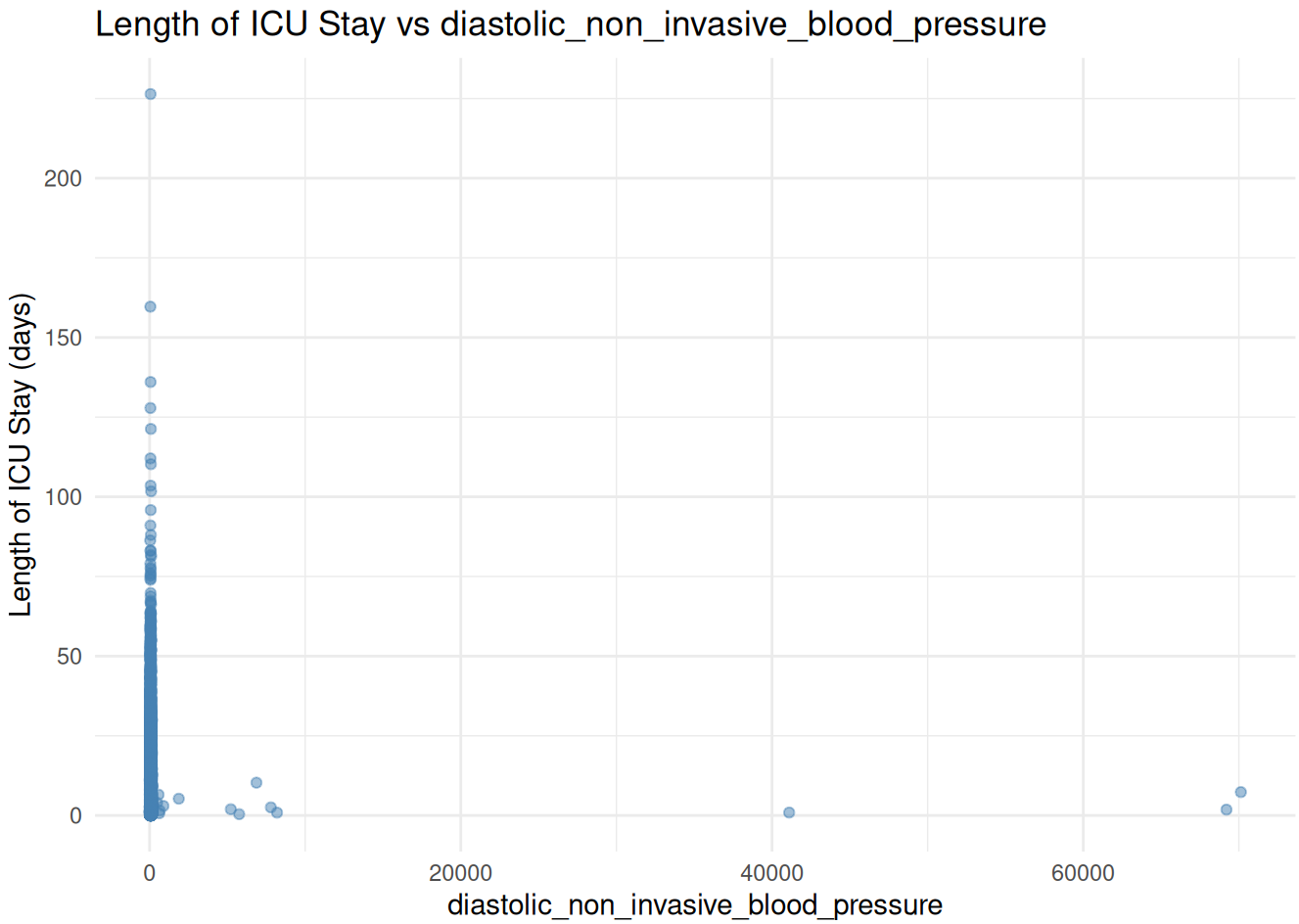
```
$heart_rate
```



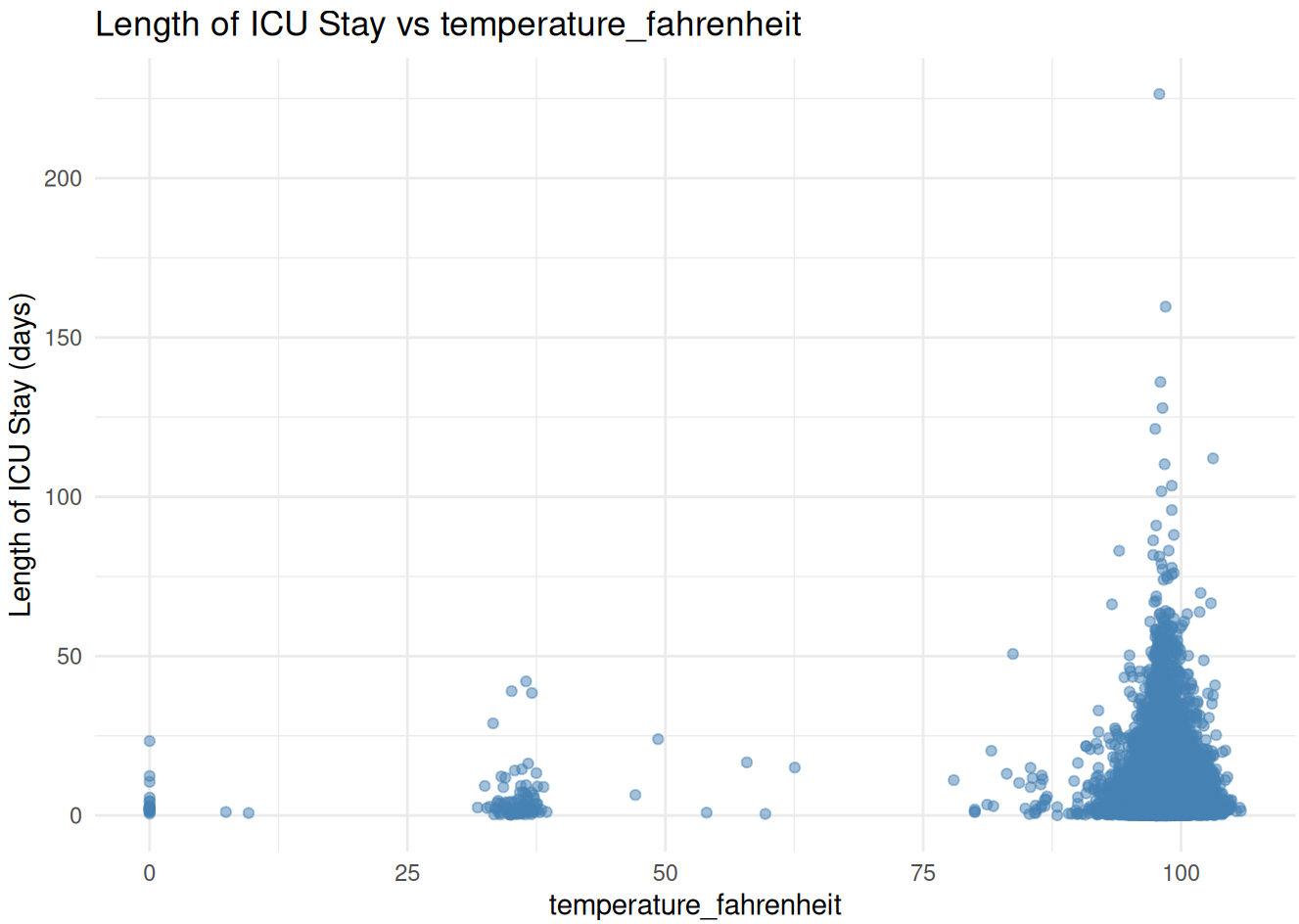

```
$systolic_non_invasive_blood_pressure
```



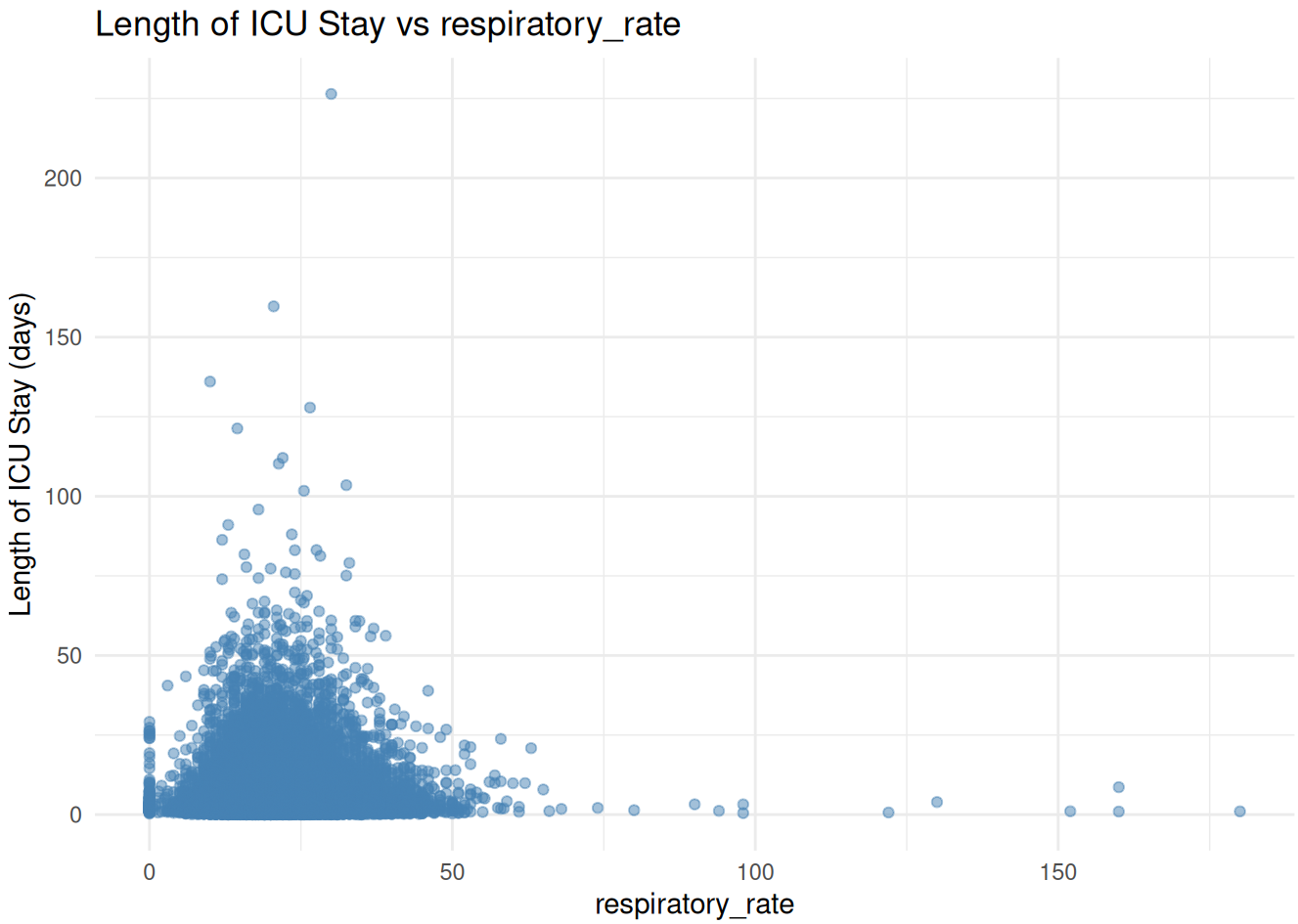

```
$diastolic_non_invasive_blood_pressure
```



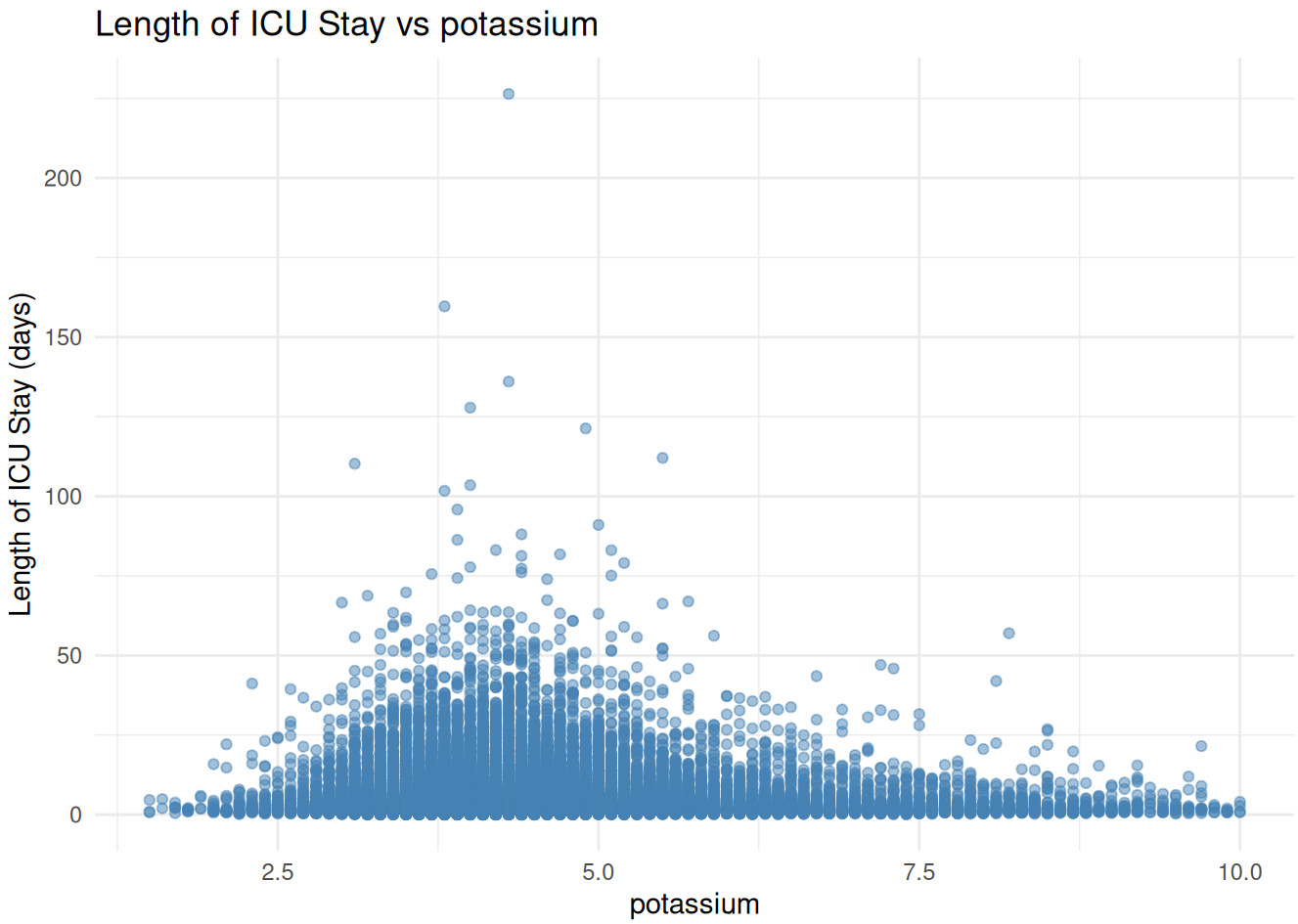

```
$temperature_fahrenheit
```



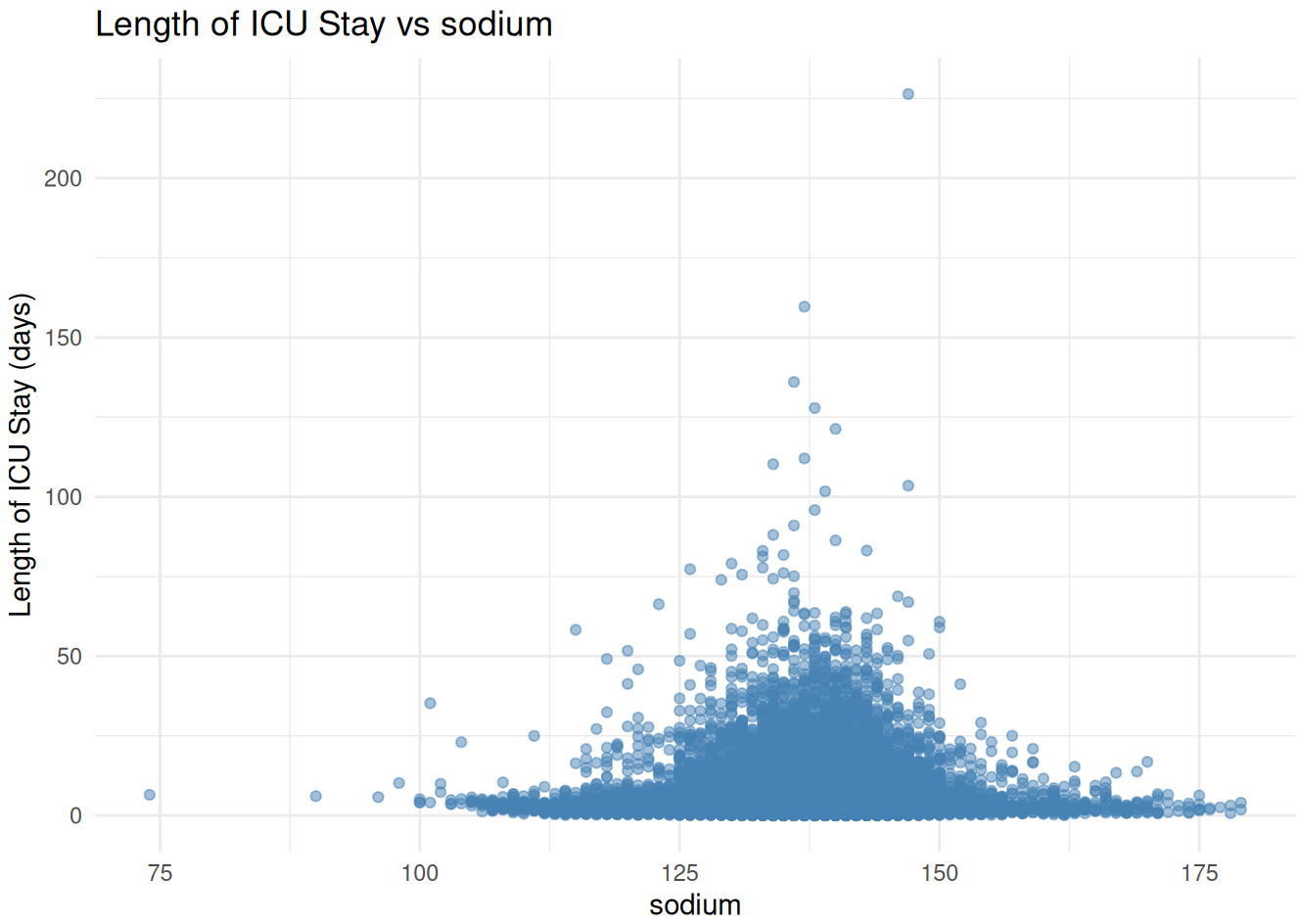

```
$respiratory_rate
```


Length of ICU Stay vs respiratory_rate
Length of ICU Stay (days)
200
150
100
50
0
0
50
100
150
respiratory_rate

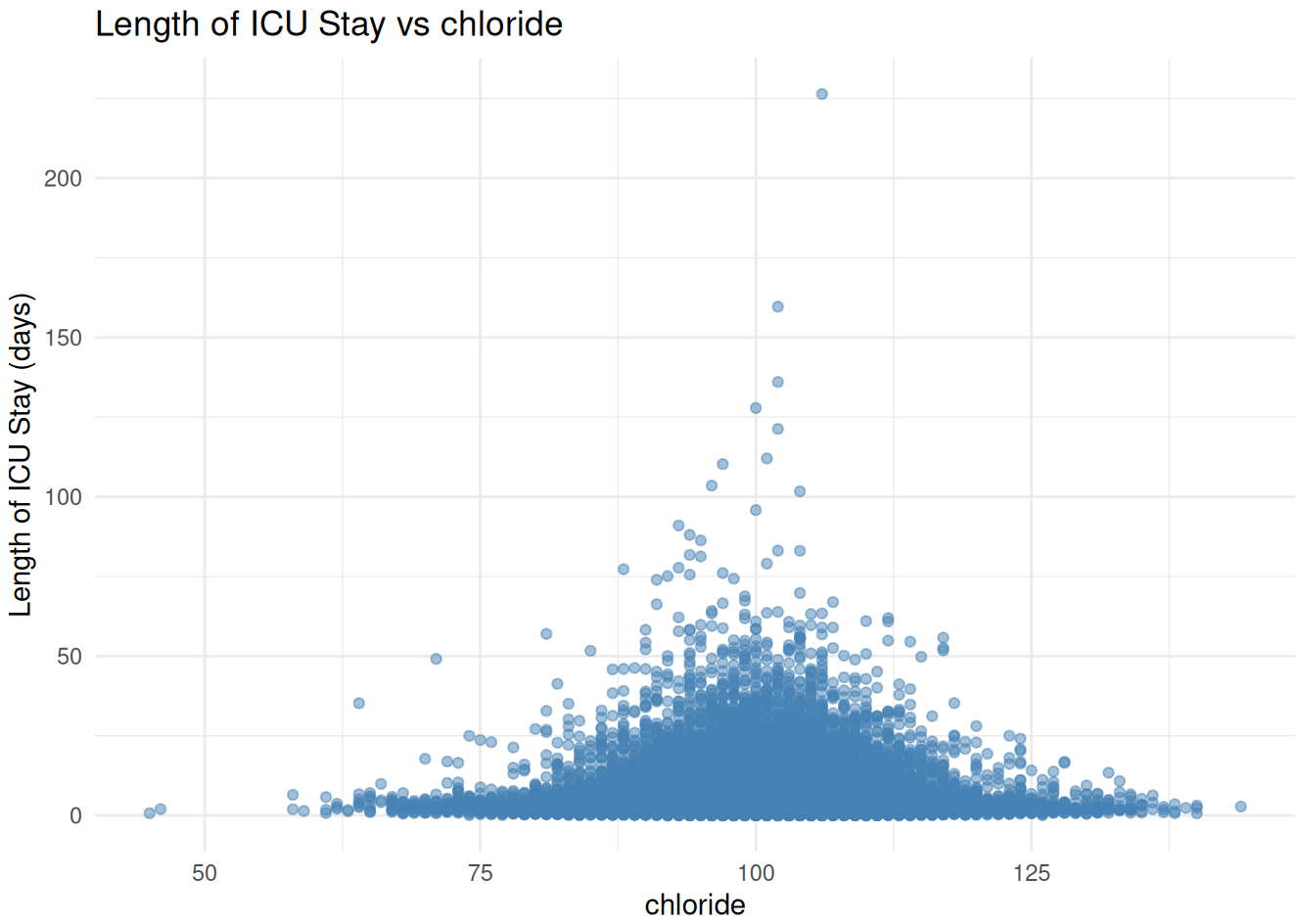
```
$potassium
```


Length of ICU Stay vs potassium
Length of ICU Stay (days)
200
150
100
50
0
2.5
5.0
7.5
10.0
potassium

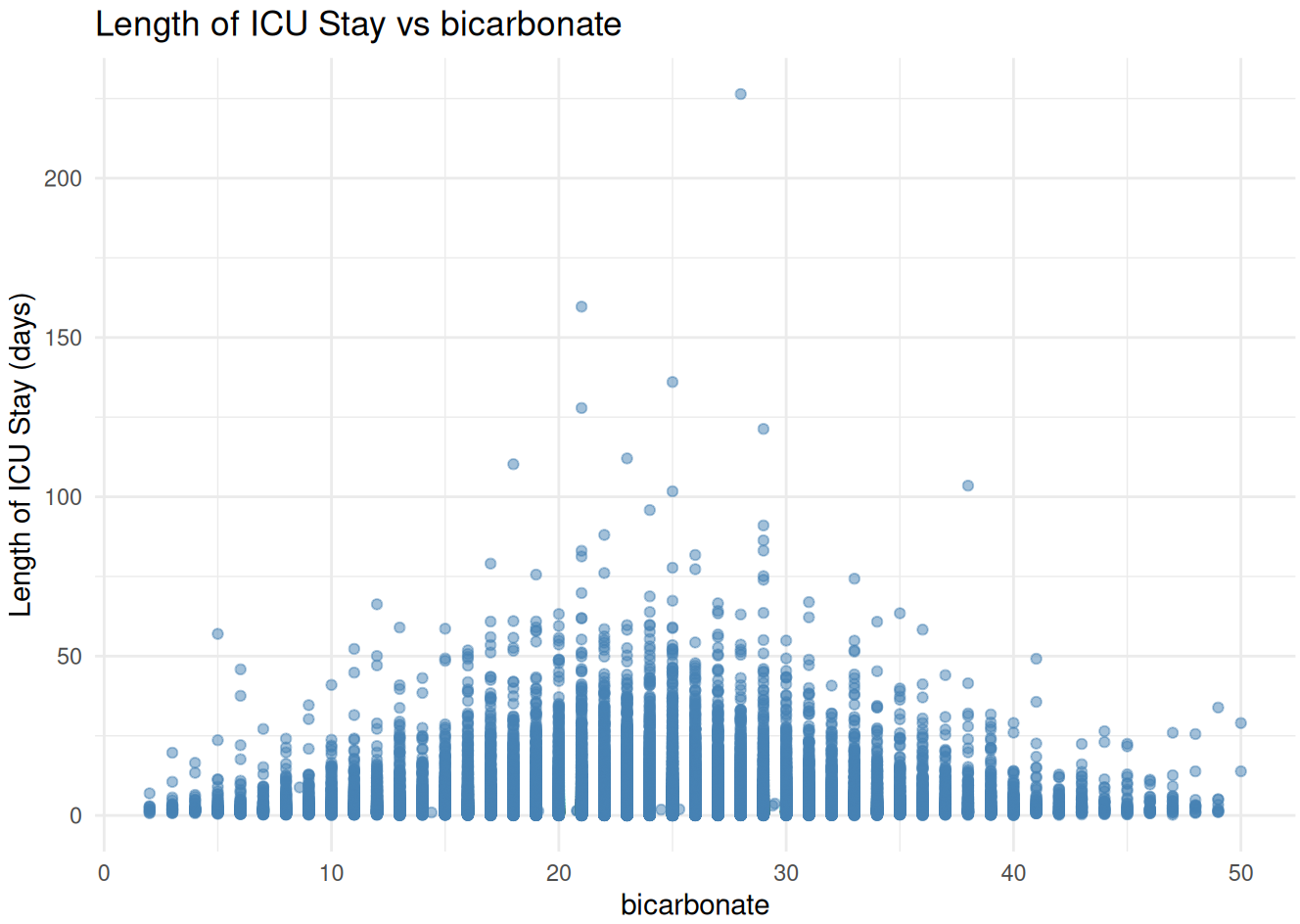
```
$sodium
```


Length of ICU Stay vs sodium
Length of ICU Stay (days)
200
150
100
50
0
75
100
125
150
175
sodium

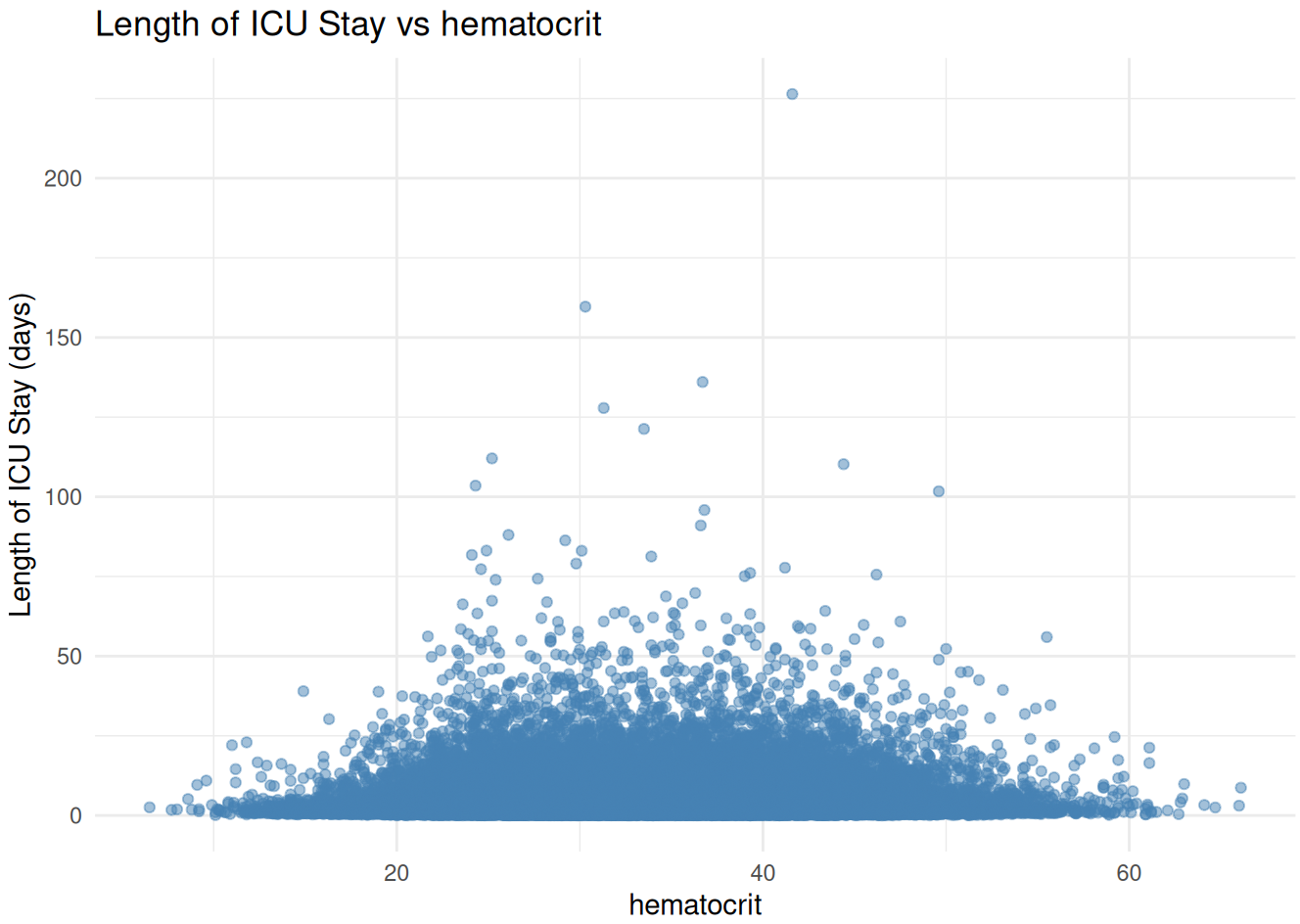
```
$chloride
```



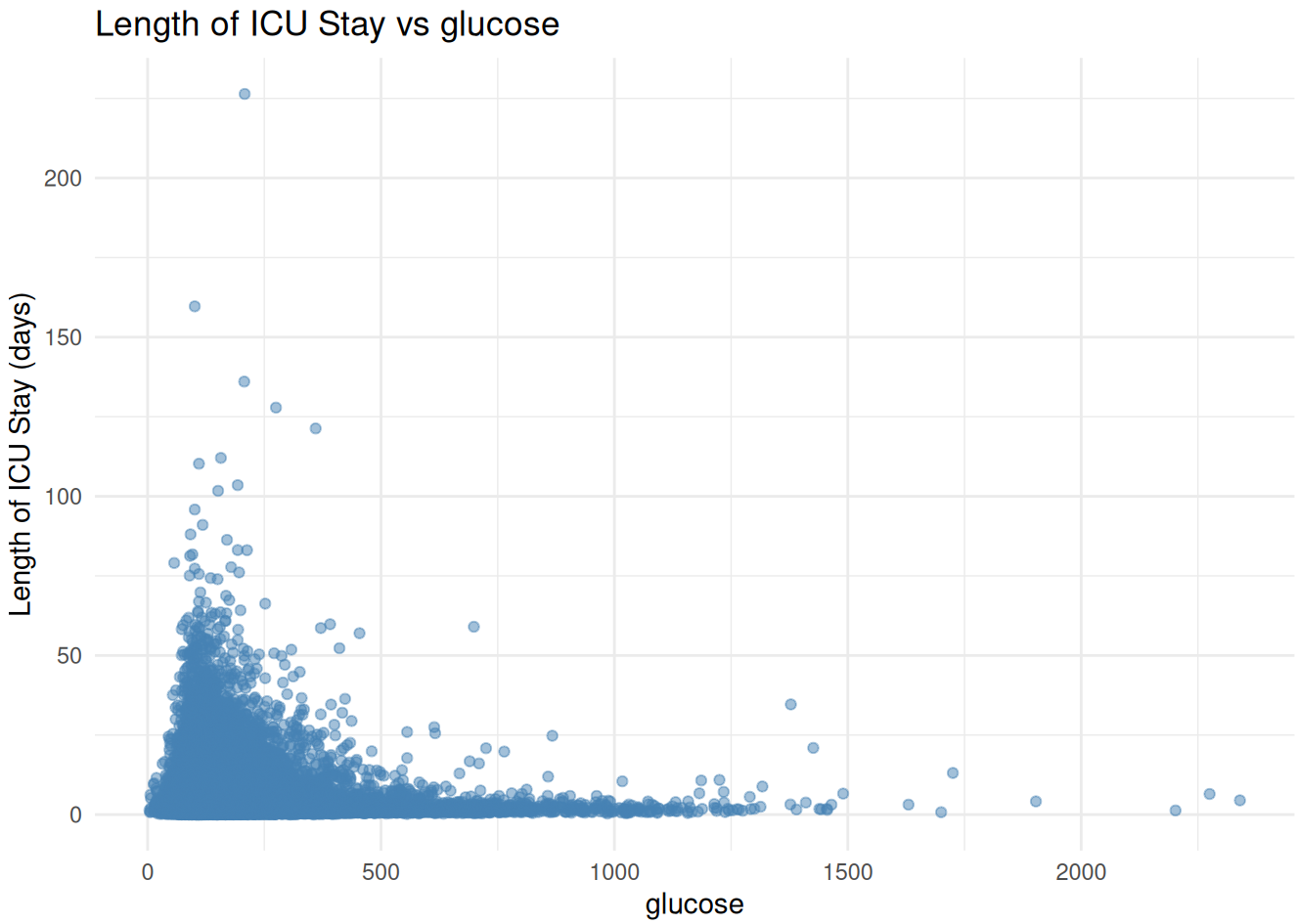

```
$bicarbonate
```


Length of ICU Stay vs bicarbonate
Length of ICU Stay (days)
0
50
100
150
200
0
10
20
30
40
50
bicarbonate

```
$hematocrit
```


Length of ICU Stay vs hematocrit
Length of ICU Stay (days)
200
150
100
50
0
20
40
60
hematocrit

```
$glucose
```

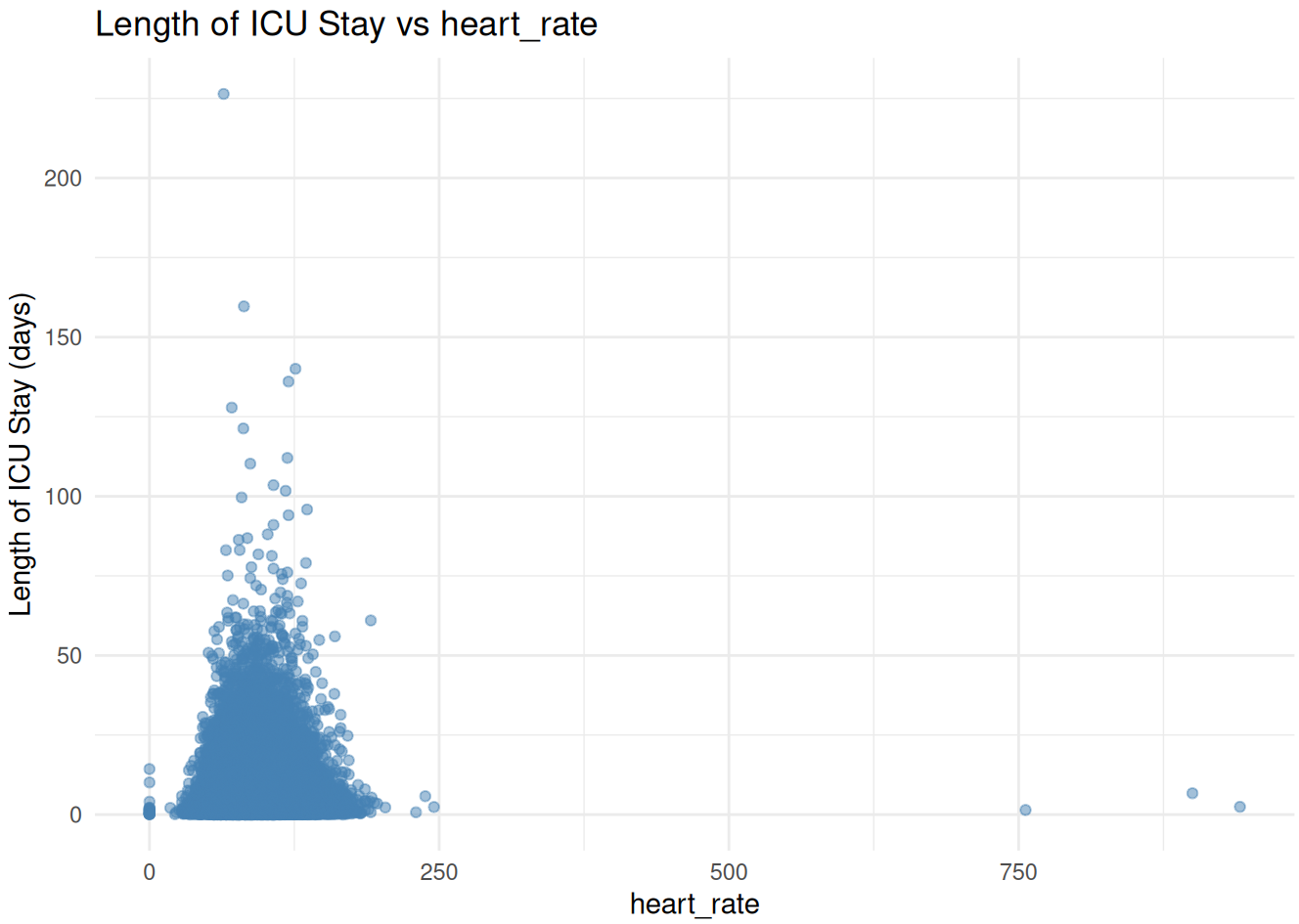


(8.3) Length of ICU stay los vs the first vital measurements within the ICU stay

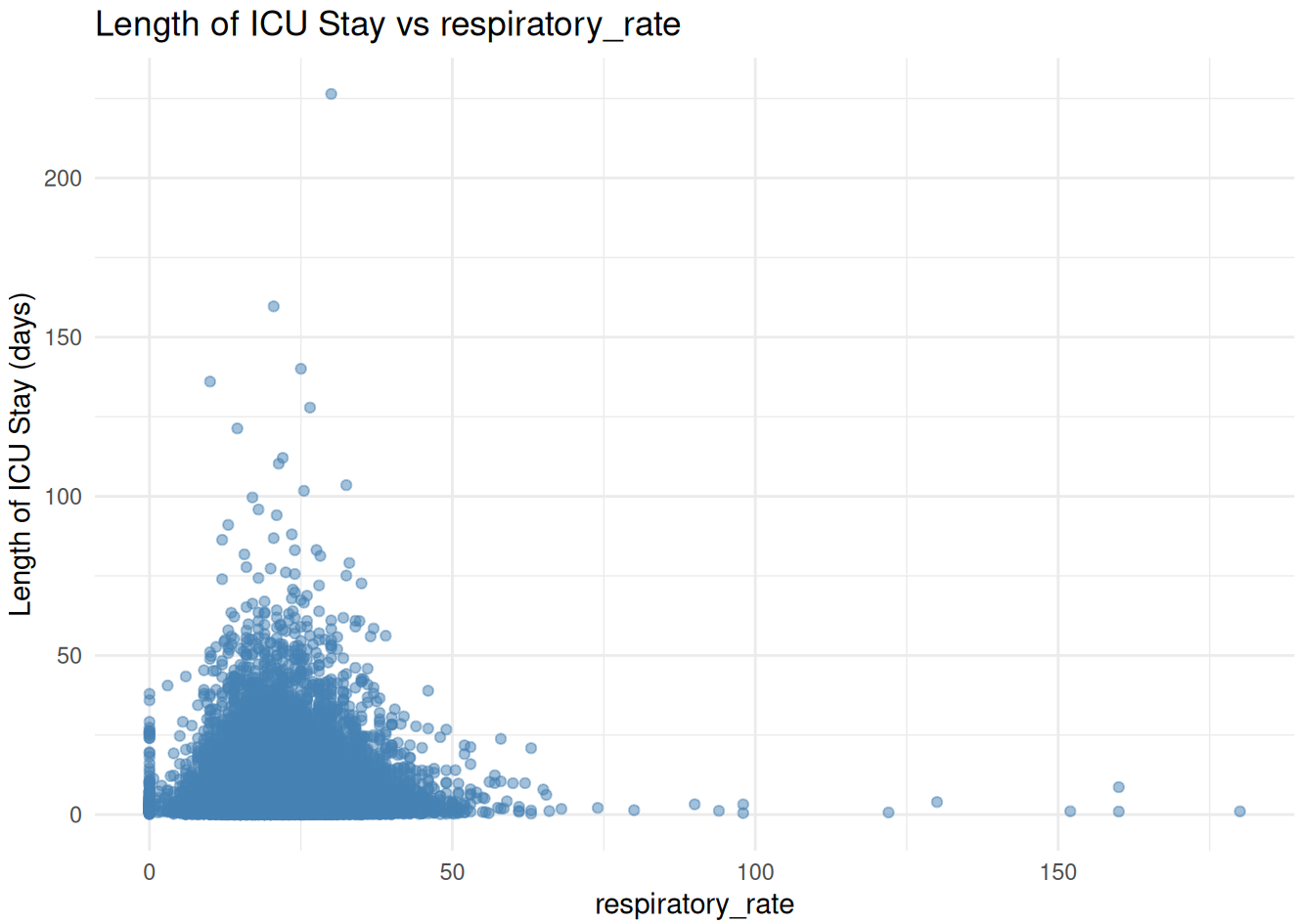
```
# get the variables which will be used to plot
variables <- c("heart_rate", "respiratory_rate", "temperature_fahrenheit",
               "systolic_non_invasive_blood_pressure",
               "diastolic_non_invasive_blood_pressure")

mimic_icu_cohort_variables <- mimic_icu_cohort %>%
  select(all_of(variables), los) %>%
  filter(if_all(all_of(variables), ~ !is.na(.)) & !is.na(los)) %>%
  mutate(across(c(all_of(variables), los),
                ~ trim(., trim_proportion = 1, na.rm = TRUE)))

# plot the required plots
plots <- list()

for (variable in variables) {
  plots[[variable]] <- mimic_icu_cohort %>%
    filter(if_all(all_of(c(variable, "los")), ~ !is.na(.))) %>%
    ggplot(aes(x = !!sym(variable), y = los)) +
    geom_point(alpha = 0.5, color = "steelblue") +
    labs(title = paste("Length of ICU Stay vs", variable),
         x = variable,
         y = "Length of ICU Stay (days)") +
```

```
    theme_minimal()
}

print(plots)
```

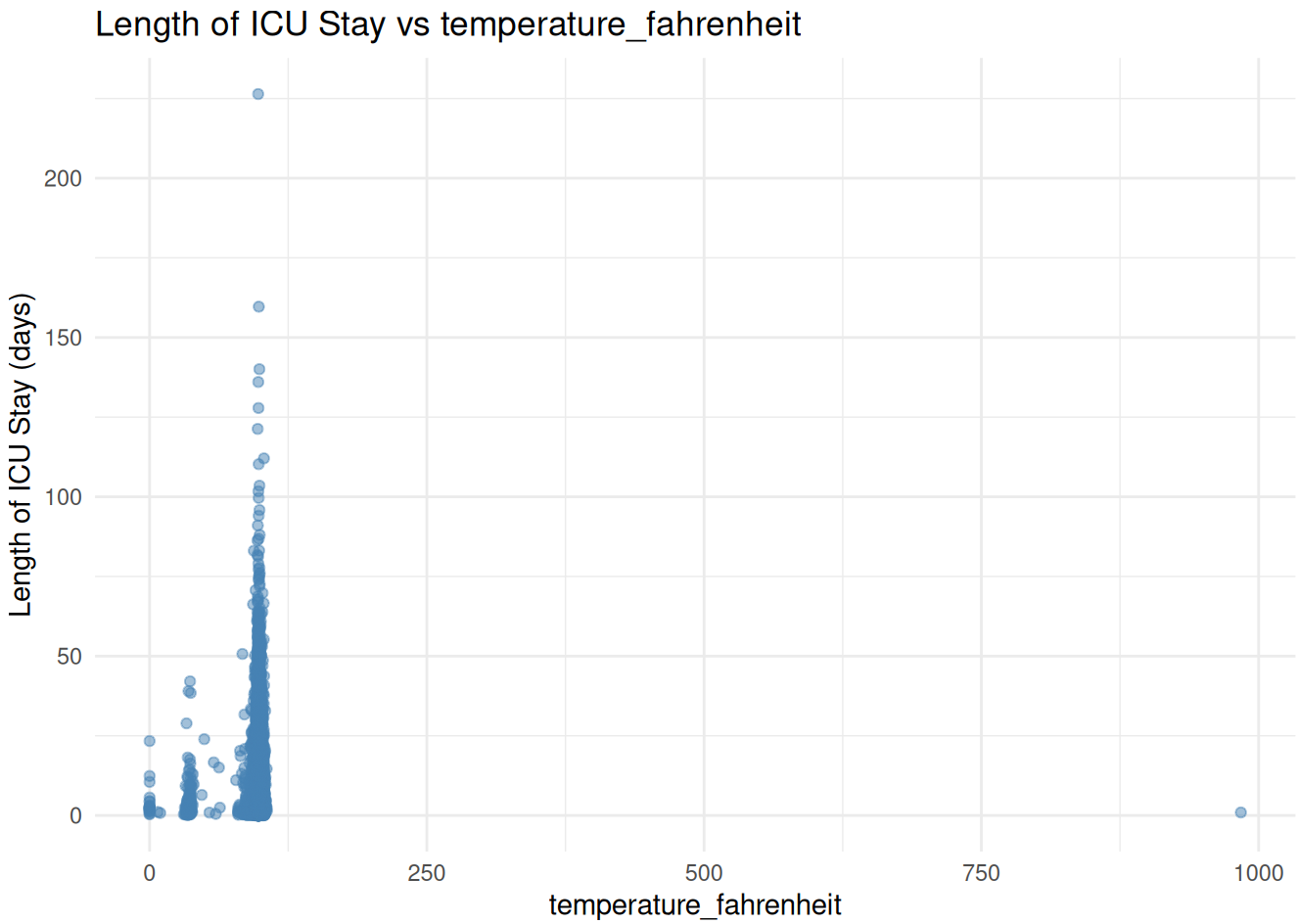
```
$heart_rate
```



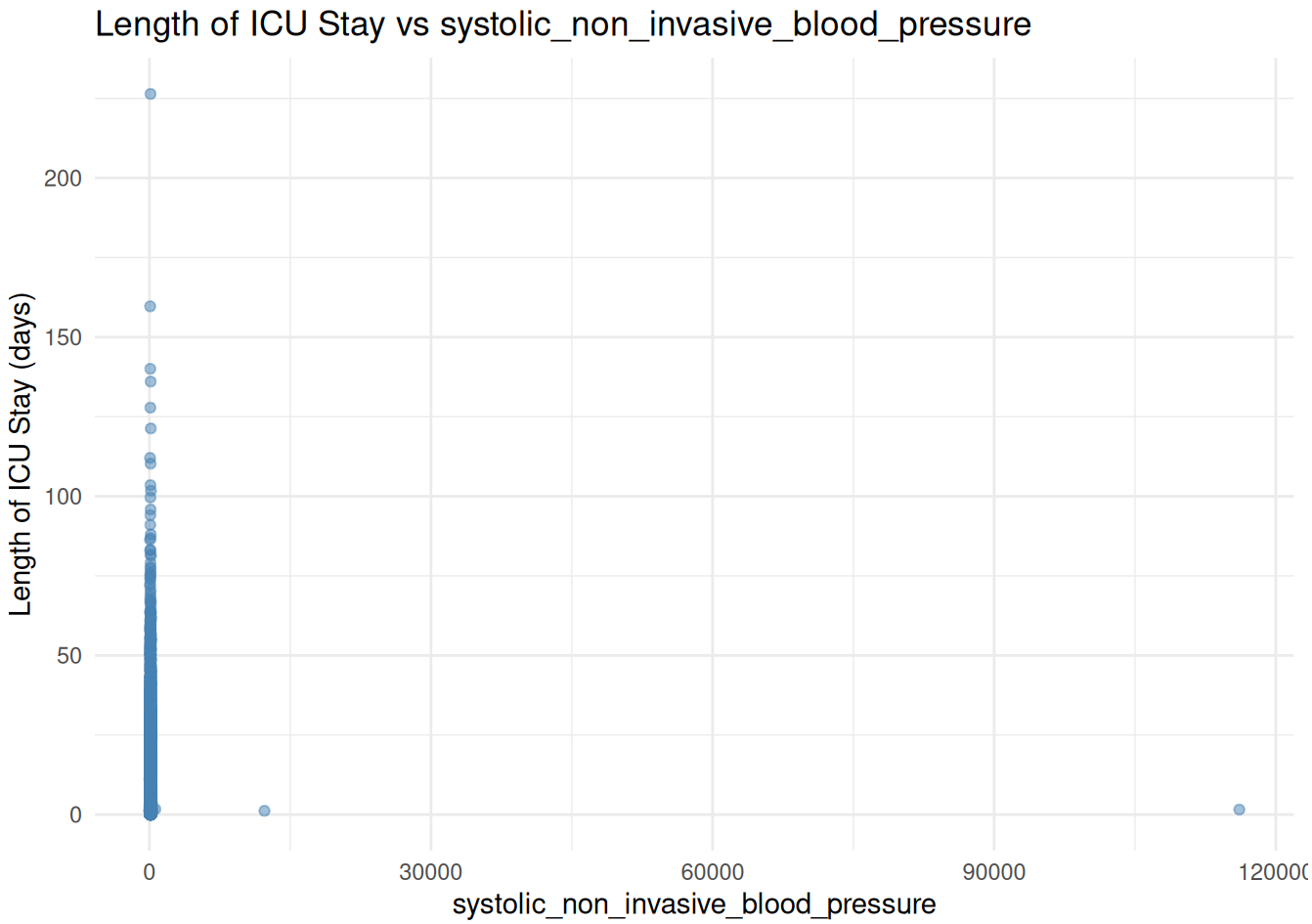

```
$respiratory_rate
```



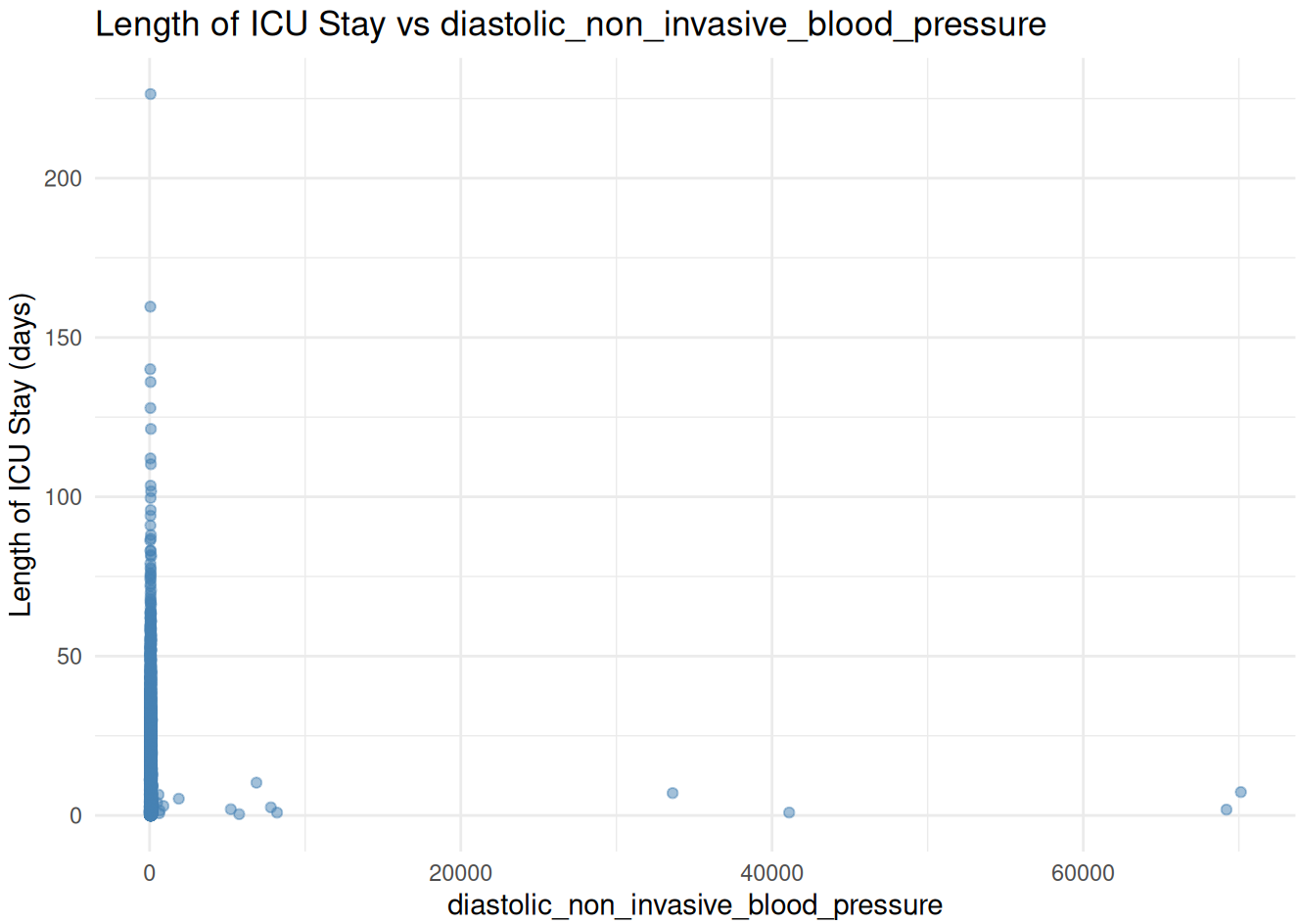

```
$temperature_fahrenheit
```




```
$systolic_non_invasive_blood_pressure
```




```
$diastolic_non_invasive_blood_pressure
```



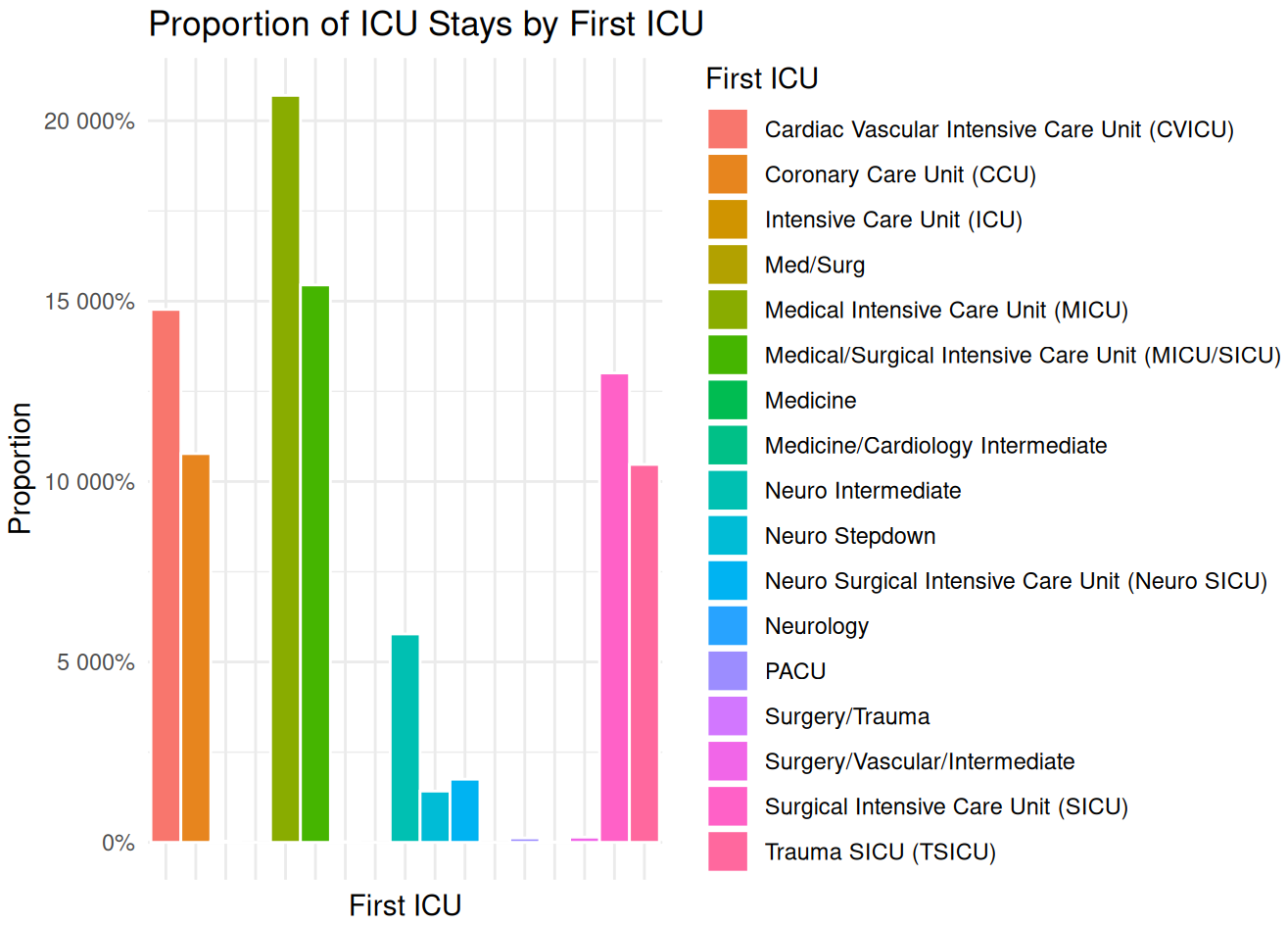
(8.4) Length of ICU stay `los` vs first ICU unit

```
# get summary statistics
mimic_icu_cohort %>%
  filter(!is.na(los) & !is.na(first_careunit)) %>%
  group_by(first_careunit) %>%
  summarise(
    count = n(),
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    min_los = min(los, na.rm = TRUE),
    max_los = max(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
    iqr_los = IQR(los, na.rm = TRUE)
  )
```

```
# A tibble: 17 × 8
   first_careunit       count mean_los median_los min_los max_los sd_los iqr_los
   <chr>                <int>    <dbl>      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
 1 Cardiac Vascular In… 14769     3.32       1.99 3.52e-3  103.     4.99    2.09
 2 Coronary Care Unit … 10771     3.09       2.01 1.25e-3   74.0    3.57    2.68
 3 Intensive Care Unit…    33     8.79       5.76 7.03e-1   41.7   10.3    10.2
 4 Med/Surg                 1     1.44       1.44 1.44e+0    1.44  NA       0
 5 Medical Intensive C… 20699     3.76       1.91 2.27e-3  226.     5.91    2.90
```

```
 6 Medical/Surgical In… 15447     3.09       1.79 1.45e-3  128.     4.59    2.17
 7 Medicine                16    15.8       13.8  1.88e+0   50.9   11.4     7.92
 8 Medicine/Cardiology…     1     2.58       2.58 2.58e+0    2.58  NA       0
 9 Neuro Intermediate    5776     5.02       3.00 3.46e-3   62.2    6.05    4.41
10 Neuro Stepdown        1421     4.07       2.20 5.70e-2   59.7    5.30    3.84
11 Neuro Surgical Inte…  1750     4.48       2.24 2.16e-2   94.1    6.46    3.59
12 Neurology                1    28.2       28.2  2.82e+1   28.2  NA       0
13 PACU                   122     4.02       2.00 1.38e-2   37.6    5.65    3.21
14 Surgery/Trauma          10    10.6       11.6  1.38e+0   20.7    6.63   11.6
15 Surgery/Vascular/In…   145    15.7       13.7  2.16e-1   63.9   12.2    15.1
16 Surgical Intensive …  13008     3.90       1.98 7.45e-3  136.     6.15    2.88
17 Trauma SICU (TSICU)   10474     3.64       1.88 4.95e-3  112.     5.42    2.77
```

```r
# graph bar plot
mimic_icu_cohort %>%
  count(first_careunit) %>%
  ggplot(aes(x = first_careunit, y = n, fill = first_careunit)) +
  geom_bar(width = 1, stat = "identity", color = "white") +

  labs(title = "Proportion of ICU Stays by First ICU",
       x = "First ICU",
       y = "Proportion",
       fill = "First ICU") +

  scale_y_continuous(labels = scales::percent_format(scale = 1)) +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank())
```


Proportion of ICU Stays by First ICU
20 000%
15 000%
10 000%
5 000%
0%
Proportion
First ICU
First ICU
Cardiac Vascular Intensive Care Unit (CVICU)
Coronary Care Unit (CCU)
Intensive Care Unit (ICU)
Med/Surg
Medical Intensive Care Unit (MICU)
Medical/Surgical Intensive Care Unit (MICU/SICU)
Medicine
Medicine/Cardiology Intermediate
Neuro Intermediate
Neuro Stepdown
Neuro Surgical Intensive Care Unit (Neuro SICU)
Neurology
PACU
Surgery/Trauma
Surgery/Vascular/Intermediate
Surgical Intensive Care Unit (SICU)
Trauma SICU (TSICU)