Biostat 203B Homework 3

Due Feb 21 @ 11:59PM

Sakshi Oza, 606542442

Display machine information for reproducibility:

```
sessionInfo()
```

```
R version 4.4.1 (2024-06-14)
Platform: aarch64-apple-darwin20
Running under: macOS Sonoma 14.4

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRlapack.dylib;  l

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_4.4.1    fastmap_1.2.0     cli_3.6.3         tools_4.4.1
 [5] htmltools_0.5.8.1 rstudioapi_0.17.0 yaml_2.3.10       rmarkdown_2.29
 [9] knitr_1.48        jsonlite_1.8.9    xfun_0.49         digest_0.6.37
[13] 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 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2

-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x purrr::compose()      masks pryr::compose()
x lubridate::duration() masks arrow::duration()
x tidyr::extract()      masks R.utils::extract()
x dplyr::filter()       masks stats::filter()
```

```
x dplyr::lag()          masks stats::lag()
x purrr::partial()      masks pryr::partial()
x dplyr::where()        masks pryr::where(), gtsummary::where()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
library(lubridate)
```

Display your machine memory.

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

```
Totalram:    8.000 GiB
Freeram:   364.688 MiB
```

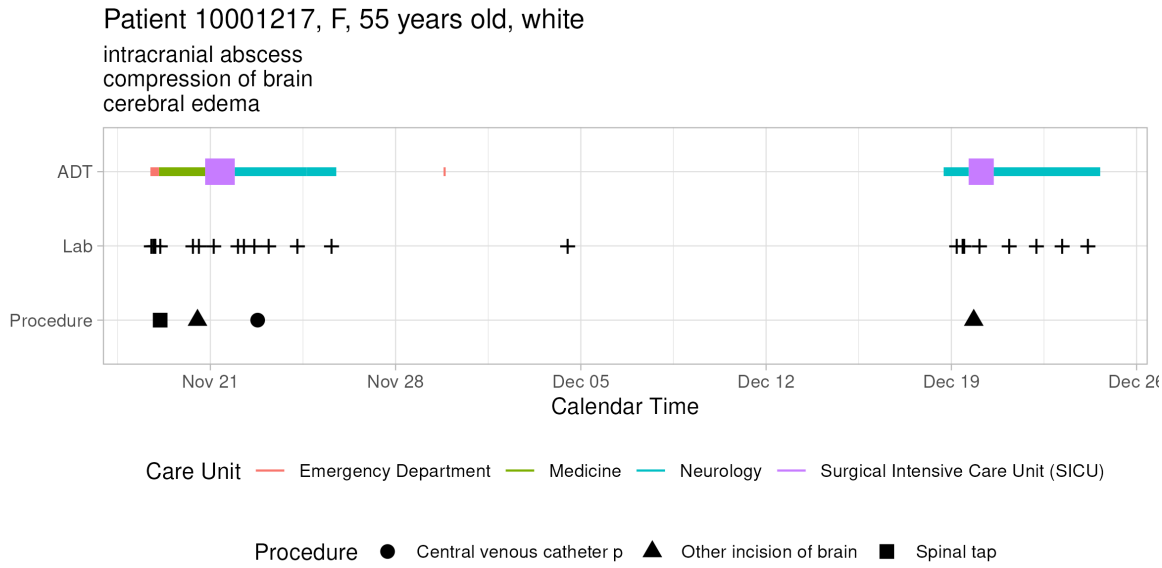
In this exercise, we use tidyverse (ggplot2, dplyr, etc) to explore the MIMIC-IV data introduced in homework 1 and to build a cohort of ICU stays.

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 1.1

```
system("gunzip -k ~/mimic/hosp/labevents.csv.gz")
```

```
# Write the CSV file to Parquet format
arrow::write_dataset(
  open_dataset("~/mimic/hosp/labevents.csv.gz", format = "csv"),
  path = "./labevents_pq.parque",
  format = "parquet"
)
```

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

```r
# File paths to the data files
patients_file <- "~/mimic/hosp/patients.csv.gz"
admissions_file <- "~/mimic/hosp/admissions.csv.gz"
transfers_file <- "~/mimic/hosp/transfers.csv.gz"
procedures_icd_file <- "~/mimic/hosp/procedures_icd.csv.gz"
diagnoses_icd_file <- "~/mimic/hosp/diagnoses_icd.csv.gz"
d_icd_procedures_file <- "~/mimic/hosp/d_icd_procedures.csv.gz"
d_icd_diagnoses_file <- "~/mimic/hosp/d_icd_diagnoses.csv.gz"
labevents_pq_dir <- "./labevents_pq.parque"
```

```r
library(arrow)

patients <- read.csv(gzfile(patients_file))
admissions <- read.csv(gzfile(admissions_file))
transfers <- read.csv(gzfile(transfers_file))
procedures_icd <- read.csv(gzfile(procedures_icd_file))
diagnoses_icd <- read.csv(gzfile(diagnoses_icd_file))
d_icd_procedures <- read.csv(gzfile(d_icd_procedures_file))
d_icd_diagnoses <- read.csv(gzfile(d_icd_diagnoses_file))
```

```r
# Filter patient data for subject_id = 10063848
subject_id <- 10063848
```

```r
lab_events_dataset <- open_dataset(labevents_pq_dir)
labevents_data <- lab_events_dataset %>%
  filter(subject_id == !!subject_id) %>%
  collect()
```

```r
head(labevents_data)
```

```
# A tibble: 6 x 16
  labevent_id subject_id  hadm_id specimen_id itemid order_provider_id
        <int>      <int>    <int>       <int>  <int> <chr>
1      950980   10063848 21345067    15946273  51133 ""
2      950981   10063848 21345067    15946273  51146 ""
3      950982   10063848 21345067    15946273  51200 ""
4      950983   10063848 21345067    15946273  51221 ""
5      950984   10063848 21345067    15946273  51222 ""
6      950985   10063848 21345067    15946273  51244 ""
# i 10 more variables: charttime <dttm>, storetime <dttm>, value <chr>,
#   valuenum <dbl>, valueuom <chr>, ref_range_lower <dbl>,
#   ref_range_upper <dbl>, flag <chr>, priority <chr>, comments <chr>
```

```r
patient_data <- patients %>% filter(subject_id == !!subject_id)
admissions_data <- admissions %>% filter(subject_id == !!subject_id)
transfers_data <- transfers %>% filter(subject_id == !!subject_id)
procedures_data <- procedures_icd %>% filter(subject_id == !!subject_id)
diagnoses_data <- diagnoses_icd %>% filter(subject_id == !!subject_id)
```

```r
# Extract necessary information
patient_info <- paste("Patient", subject_id, ",", patient_data$gender[1], ",", patient_data$a
diagnoses <- diagnoses_data %>%
  left_join(d_icd_diagnoses, by = "icd_code") %>%
  top_n(3, wt = "seq_num") %>%
  pull(long_title)
```

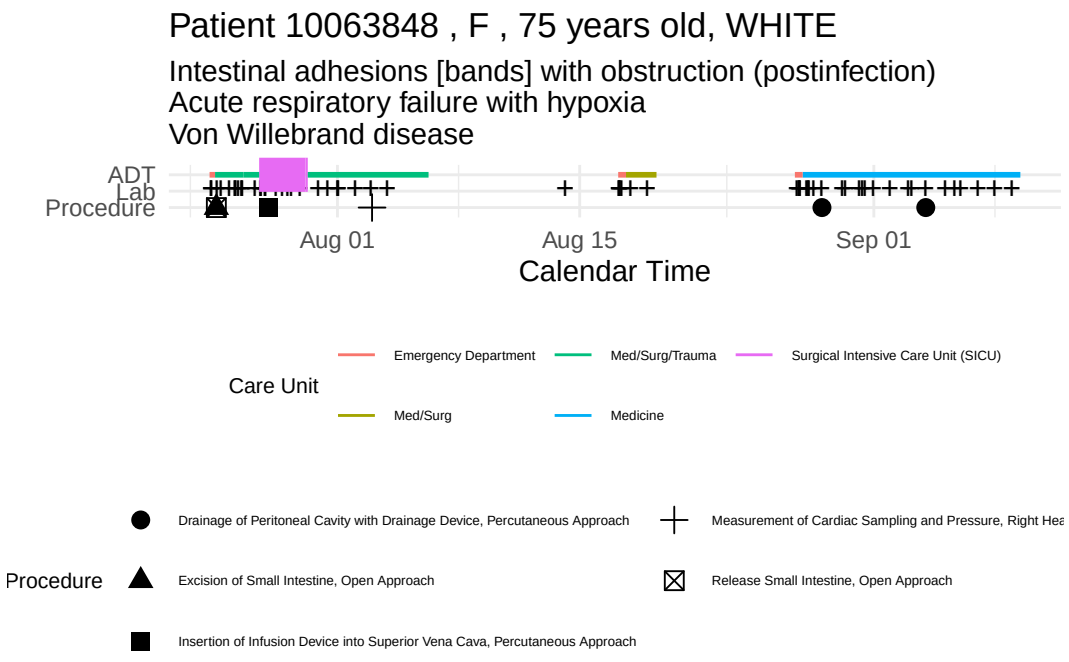
```
Warning in left_join(., d_icd_diagnoses, by = "icd_code"): Detected an unexpected many-to-ma
i Row 17 of `x` matches multiple rows in `y`.
i Row 15793 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship =
  "many-to-many"` to silence this warning.
```

```r
diagnoses_text <- paste(head(diagnoses, 3), collapse = "\n")

# Convert intime and outtime to POSIXct format
transfers_data <- transfers_data %>%
  mutate(intime = as.POSIXct(intime, format = "%Y-%m-%d %H:%M:%S", tz = "UTC"),
         outtime = as.POSIXct(outtime, format = "%Y-%m-%d %H:%M:%S", tz = "UTC"))

# Prepare data for ADT events (Admission, Discharge, Transfer)
adt_data <- transfers_data %>%
  filter(careunit != "UNKNOWN") %>%  # Drop rows where careunit is "unknown"
  mutate(care_unit_duration = difftime(outtime, intime, units = "hours"),
         line_width = ifelse(grepl("icu", careunit, ignore.case = TRUE),
                             3, 2.5))


# Prepare lab event data
lab_data <- labevents_data %>%
  group_by(charttime) %>%
  summarise(n = n()) %>%
  mutate(event = "Lab")

# Prepare procedures data
```

```r
procedures_data_1 <- procedures_data %>%
  left_join(d_icd_procedures, by = "icd_code") %>%
  mutate(event =
  case_when(icd_code == "Central venous catheter" ~ "Central venous catheter",
  icd_code == "Other incision of brain" ~ "Other incision of brain",
                           TRUE ~ "Spinal tap"))
procedures_data_1 <- procedures_data_1 %>%
  mutate(chartdate = as.POSIXct(chartdate, format = "%Y-%m-%d"))
```

```r
ggplot() +
  # Procedure events
  geom_point(data = procedures_data_1,
             aes(x = chartdate,
                 y = factor("Procedure",
                            levels = c("ADT", "Lab", "Procedure")),
                 shape = long_title),
             size = 3, position = position_nudge(y = 0)) +

  # Lab events
  geom_point(data = lab_data,
             aes(x = charttime,
                 y = factor("Lab",
                            levels = c("ADT", "Lab", "Procedure"))),
             shape = '+', size = 4, position = position_nudge(y = 0)) +

  # ADT events
  geom_segment(data = adt_data,
               aes(x = intime, xend = outtime,
                   y = factor("ADT", levels = c("ADT", "Lab", "Procedure")),
                   yend = factor("ADT", levels = c("ADT", "Lab", "Procedure")),
                   color = careunit, linewidth = line_width),
               show.legend = c(linewidth = FALSE)) +


  #ggtitle(paste(patient_info, "\n", diagnoses_text)) +
  ggtitle(patient_info, subtitle = diagnoses_text) +

  # Axis labels
  xlab("Calendar Time") +
  ylab("") +
```

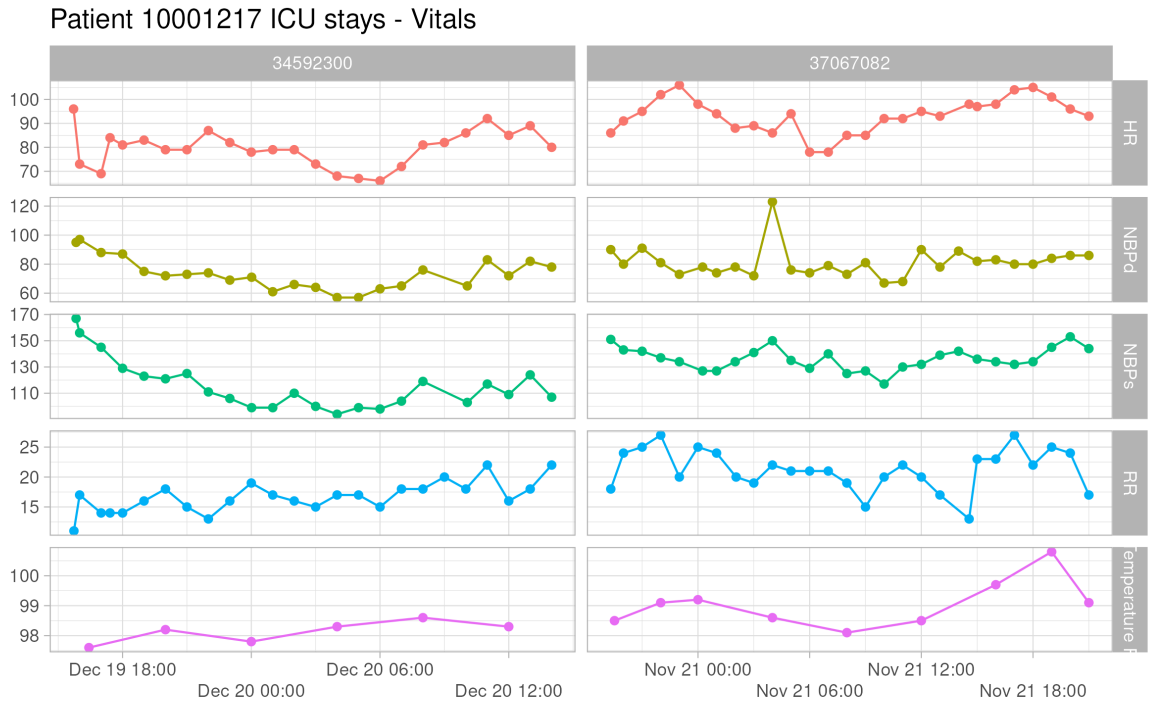
```
  labs(shape = "Procedure") +
  labs(color = "Care Unit") +

  guides(color = guide_legend(order = 1), shape = guide_legend(order = 2)) +
guides(
  shape = guide_legend(nrow = 3),
  color = guide_legend(nrow = 2)
) +
  theme_minimal() +

  coord_cartesian(clip = "off") +
theme(
  plot.margin = margin(1, 1, 1, 1),
  legend.position = "bottom",
  legend.box = "vertical",
  legend.text = element_text(size = 5),
  legend.title  = element_text(size = 8)
)
```



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 1.2

```
arrow::write_dataset(
  open_dataset("~/mimic/icu/procedureevents.csv.gz", format = "csv"),
  path = "./procedureevents_pq.parque",
  format = "parquet"
)
```

```
chartevents_pq_dir<- "./chartevents_pq.parque"
```

```
subject_id <- 10063848
```

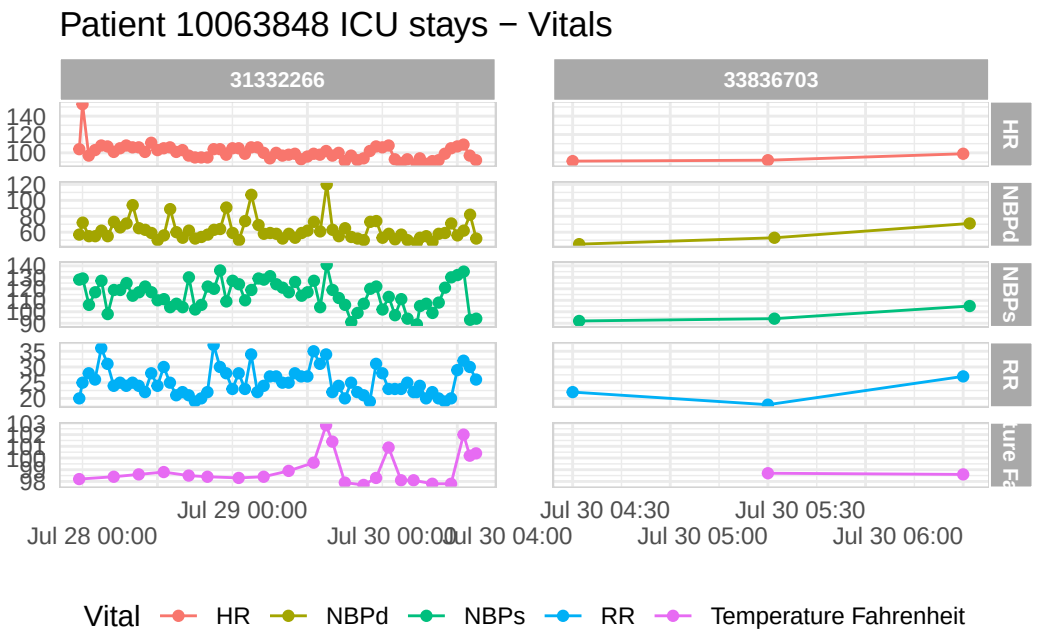
```
chartevents_dataset <- open_dataset(chartevents_pq_dir)
chartevents_data <- chartevents_dataset %>%
```

```
  filter(subject_id == !!subject_id) %>%
  collect()
```

```
# Filter data to include the relevant vitals
vital_data <- chartevents_data %>%
  filter(itemid %in% c(220045, 220180, 220179, 223761, 220210)) %>%
  mutate(

    # Create a new column to label the vital type based on itemid
    vital_type = case_when(
      itemid == 220045 ~ "HR",
      itemid == 220180 ~ "NBPd",    # Non-invasive Blood Pressure (Diastolic)
      itemid == 220179 ~ "NBPs",     # Non-invasive Blood Pressure (Systolic)
      itemid == 220210 ~ "RR",        # Respiratory Rate
      itemid == 223761 ~ "Temperature Fahrenheit", # Temperature
      TRUE ~ NA_character_
    )
  )



# Create a line plot for the vitals
ggplot(vital_data, aes(x = charttime, y = valuenum, color = vital_type)) +
  geom_line() +
  geom_point()+


  # Facet by vital type and stay_id
  facet_grid(vital_type ~ stay_id, scales = "free") +
  scale_x_datetime(
    labels = function(x) {
      labels <- format(x, "%b %d %H:%M")
      labels_alternate <- ifelse(seq_along(labels) %% 2 == 0,
                                 paste0("\n", labels),
                                 labels)
      return(labels_alternate)
    }
  ) +
  labs(
    title = paste("Patient",subject_id,"ICU stays - Vitals"),
    x= NULL,
    y= NULL,
```

```
    color ="Vital"
  )+
  # Set a theme for clarity
  theme_minimal() +

  # Adjust legend position
theme(

    strip.text.x = element_text(size = 8, face = "bold", color = "white"),
    strip.text.y = element_text(size = 8, face = "bold", color = "white"),
    strip.background = element_rect(fill = "darkgray", color = "white"),

  axis.text.x = element_text(angle = 0 , hjust = 1),
  panel.border = element_rect(color = "lightgray", fill = NA, size = 0.5),
     panel.spacing.x = unit(0.75, "cm"),
    legend.position = "bottom"  ,

  )
```

```
Warning: The `size` argument of `element_rect()` is deprecated as of ggplot2 3.4.0.
i Please use the `linewidth` argument instead.
```



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
```

Q2.1 Ingestion

Solution 2.1

```
# Load necessary libraries
library(tibble)
library(readr)

# Import icustays.csv.gz as a tibble
icustays_tble <- read_csv("~/mimic/icu/icustays.csv.gz") %>%
  as_tibble()
```

```
Rows: 94458 Columns: 8
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (2): first_careunit, last_careunit
dbl  (4): subject_id, hadm_id, stay_id, los
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.
```

```
# Check the first few rows of the tibble
print(head(icustays_tble),10)
```

```
# A
#   tibble:
#   6 x 8
# i 8
#   more
#   variables:
#   subject_id <dbl>,
```

```
#   hadm_id <dbl>,
#   stay_id <dbl>,
#   first_careunit <chr>, ...
```

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.

Solution 2.2

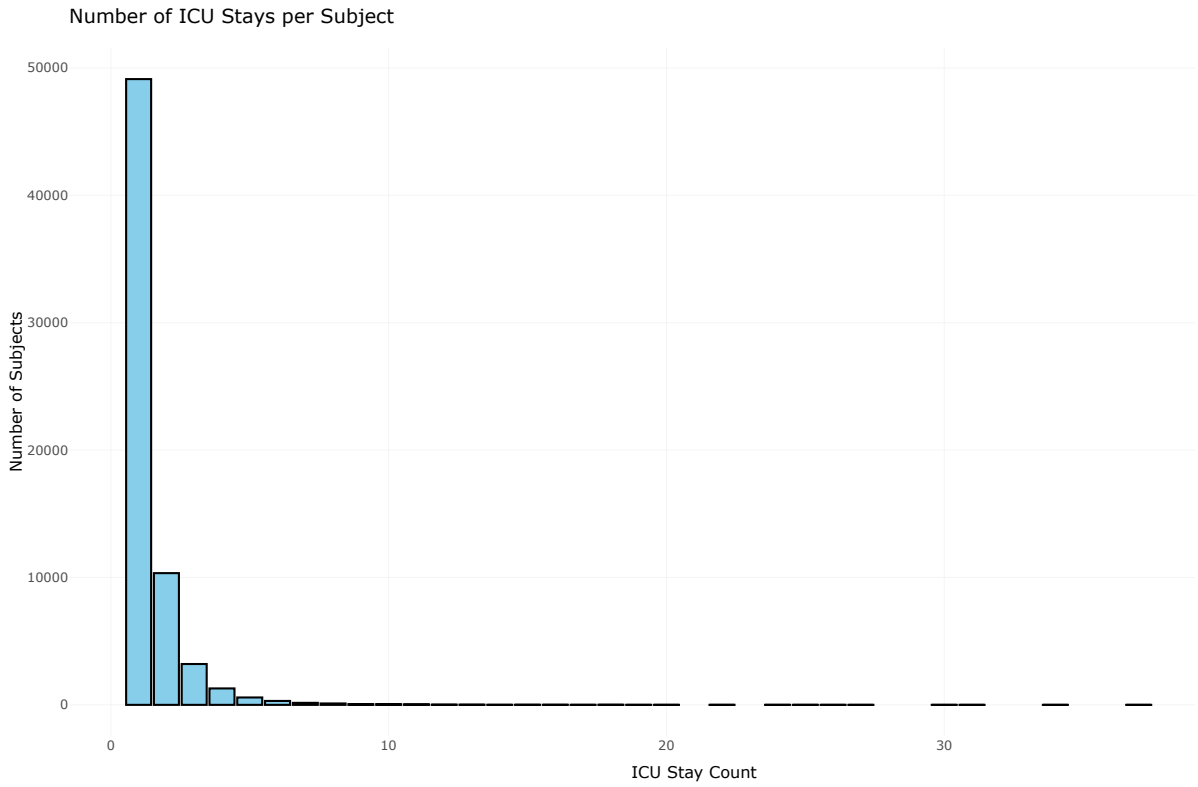
```
library(plotly)
```

```
Attaching package: 'plotly'

The following object is masked from 'package:ggplot2':

    last_plot

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

    schema

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

    filter

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

    layout
```

```
# Count the number of ICU stays per subject
icu_stays_summary <- icustays_tble %>%
  group_by(subject_id) %>%
  summarise(icu_stay_count = n()) %>%
  arrange(desc(icu_stay_count))
print(icu_stays_summary)
```

```
# A tibble: 65,366 x 2
   subject_id icu_stay_count
        <dbl>          <int>
 1   12468016             41
 2   18358138             37
 3   17585185             34
 4   17295976             31
 5   13269859             30
 6   18676703             27
 7   12517625             26
 8   11281568             25
 9   15229355             25
10   15455517             25
# i 65,356 more rows
```

```
p <- ggplot(icu_stays_summary, aes(x = icu_stay_count)) +
  geom_bar(fill = "skyblue", color = "black") +
  labs(
    title = "Number of ICU Stays per Subject",
    x = "ICU Stay Count",
    y = "Number of Subjects"
  ) +
  theme_minimal()

# Convert to an interactive plot using plotly
interactive_plot <- ggplotly(p)

# Display the plot
interactive_plot
```



- Most subjects have a single ICU stay: The vast majority of subjects (around 50,000) had only one ICU stay. This is the tallest bar on the far left, indicating that most patients only visit the ICU once.
- Gradual decrease for multiple stays: There is a sharp decline in the number of subjects as the ICU stay count increases. Fewer subjects had multiple ICU stays.
- Very few subjects have more than 10 ICU stays: Beyond 10 ICU stays, the number of subjects drops to almost zero, and the graph flattens out. There are only a few outliers with an ICU stay count greater than 20, indicating that it's extremely rare for patients to be admitted to the ICU more than 10-15 times.

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
```

Q3.1 Ingestion

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

Solution 3.1

```
# Import icustays.csv.gz as a tibble
admissions_tble <- read_csv("~/mimic/hosp/admissions.csv.gz") %>%
  as_tibble()
```

```
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.
```

```
# Check the first few rows of the tibble
print(admissions_tble)
```

```
# A tibble: 546,028 x 16
   subject_id  hadm_id admittime           dischtime
        <dbl>    <dbl> <dttm>              <dttm>
 1   10000032 22595853 2180-05-06 22:23:00 2180-05-07 17:15:00
 2   10000032 22841357 2180-06-26 18:27:00 2180-06-27 18:49:00
 3   10000032 25742920 2180-08-05 23:44:00 2180-08-07 17:50:00
 4   10000032 29079034 2180-07-23 12:35:00 2180-07-25 17:55:00
 5   10000068 25022803 2160-03-03 23:16:00 2160-03-04 06:26:00
 6   10000084 23052089 2160-11-21 01:56:00 2160-11-25 14:52:00
 7   10000084 29888819 2160-12-28 05:11:00 2160-12-28 16:07:00
 8   10000108 27250926 2163-09-27 23:17:00 2163-09-28 09:04:00
 9   10000117 22927623 2181-11-15 02:05:00 2181-11-15 14:52:00
10   10000117 27988844 2183-09-18 18:10:00 2183-09-21 16:30:00
# i 546,018 more rows
# i 12 more variables: 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 <dbl>
```

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.

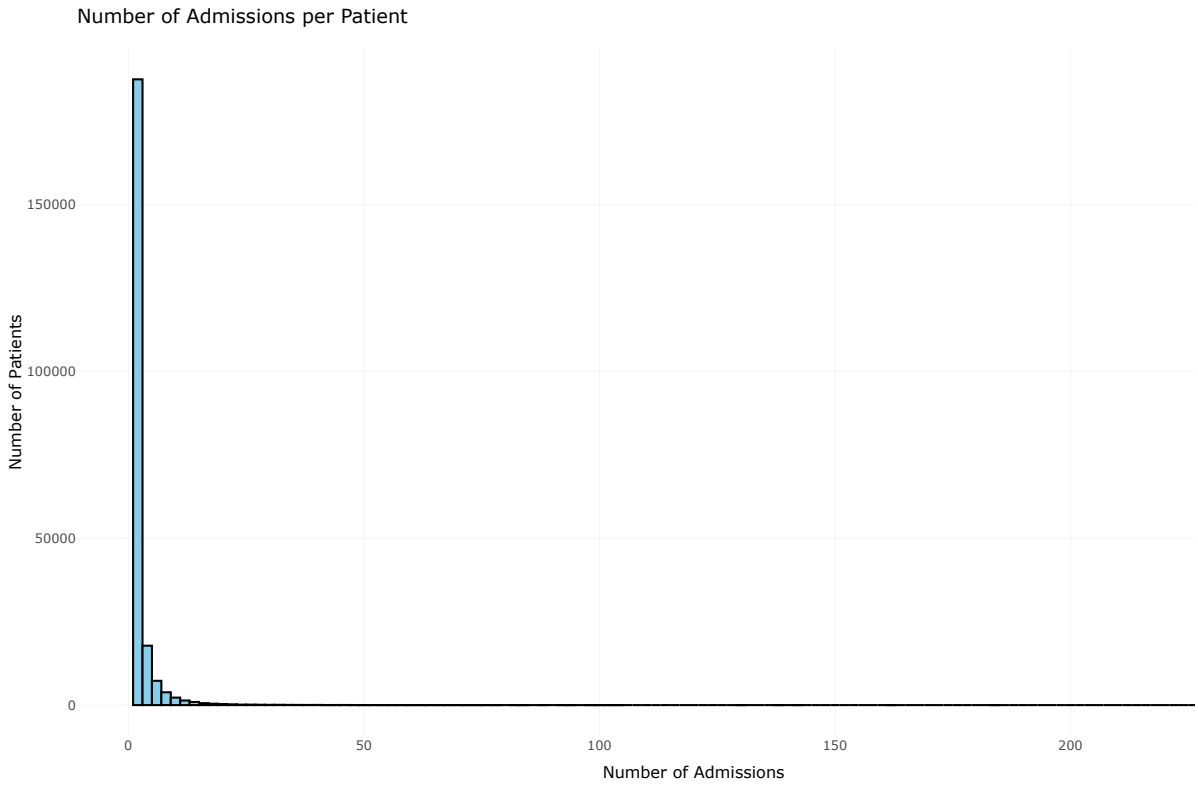
Admissions per patient

Solution 3.2

Number of admissions per patient

```r
library(plotly)

# Number of admissions per patient
admissions_per_patient <- admissions_tble %>%
  group_by(subject_id) %>%
  summarise(admission_count = n())

# Static ggplot2 plot
p1 <- ggplot(admissions_per_patient, aes(x = admission_count)) +
  geom_histogram(binwidth = 2, fill = "skyblue", color = "black") +
  labs(
    title = "Number of Admissions per Patient",
    x = "Number of Admissions",
    y = "Number of Patients"
```

```
  ) +
  theme_minimal()

# Convert to interactive plot
ggplotly(p1)
```



The "Number of Admissions per Patient" histogram demonstrates a highly skewed distribution, with a significant majority of patients experiencing only a single admission. Visually, the first bar on the histogram is dramatically taller than all subsequent bars, indicating a disproportionately high count of single admissions. This indicates that most hospital visits are likely for acute conditions or short-term procedures. The rapid decline in patient counts with increasing admissions points to a smaller cohort requiring multiple hospitalizations.

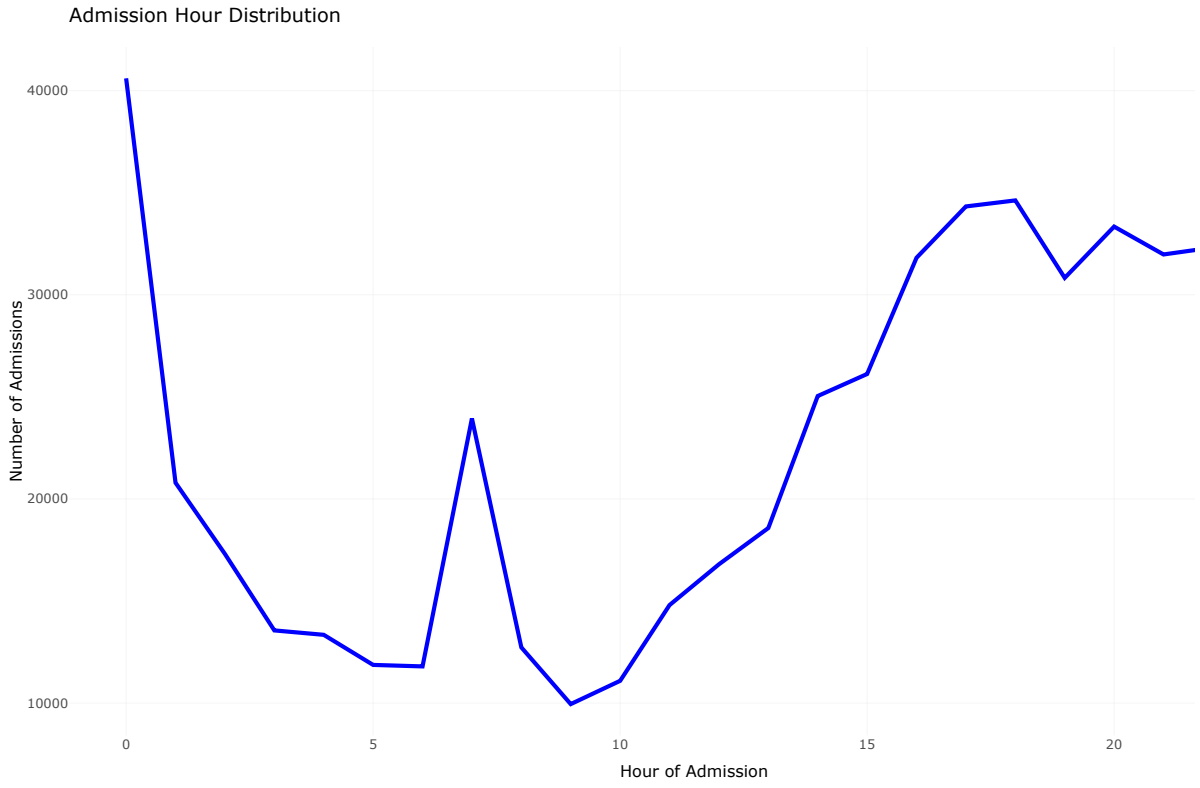
number of Admission hour

```
# Extract admission hour
admissions_hour <- admissions_tble %>%
  mutate(admission_hour = hour(admittime)) %>%
  group_by(admission_hour) %>%
```

```
  summarise(count = n())

# Static ggplot2 plot
p2 <- ggplot(admissions_hour, aes(x = admission_hour, y = count)) +
  geom_line(color = "blue", size = 1) +
  labs(
    title = "Admission Hour Distribution",
    x = "Hour of Admission",
    y = "Number of Admissions"
  ) +
  theme_minimal()
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```

```
# Convert to interactive plot
ggplotly(p2)
```

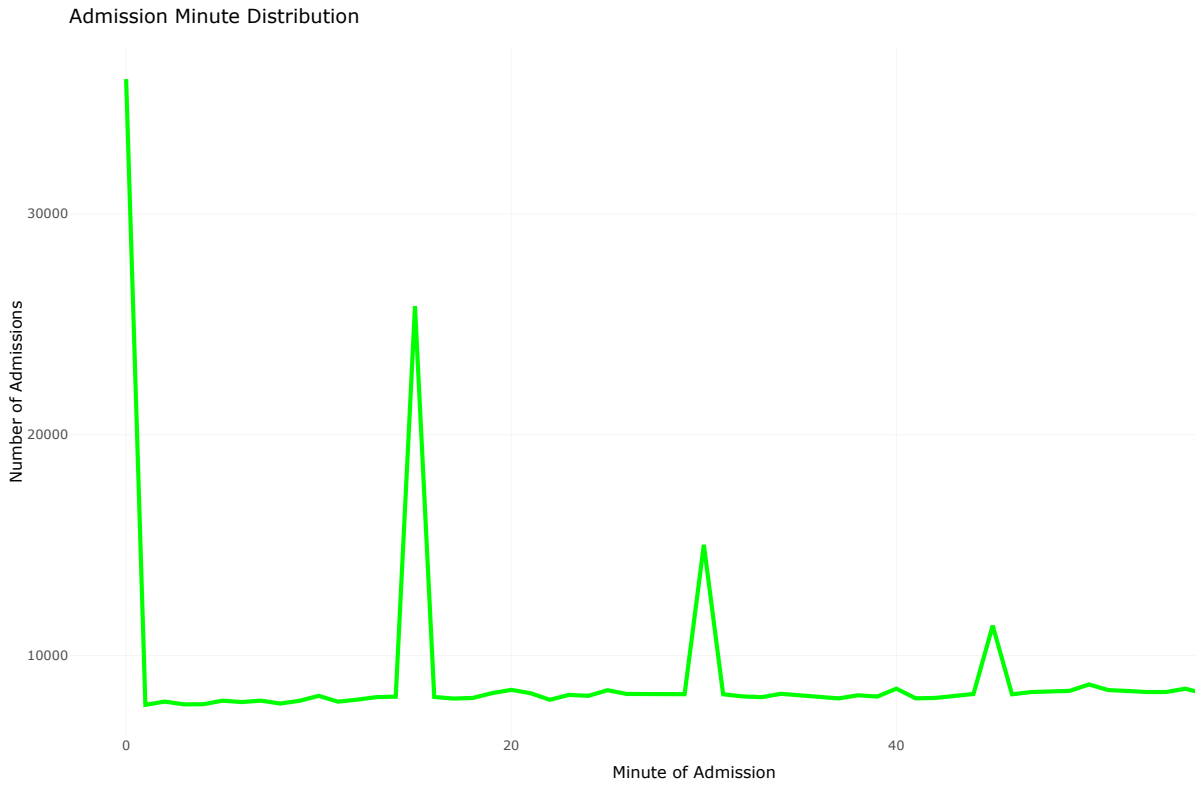


The "Admission Hour Distribution" graph reveals a notable peak in admissions at midnight, suggesting administrative or system-related influences. The midnight peak is the highest point

on the graph, significantly exceeding the values observed in other hours.The spike at hour 0 is unusually high and might not represent actual patient admissions but more of a data recording anomaly. It's common for large spikes at midnight due to default values in databases or rounding practices in electronic health records (EHRs). A decrease in admissions is observed during early morning hours, followed by increases during mid-morning and afternoon, reflecting scheduled procedures and emergency department activity. A plateau in the evening indicates sustained hospital activity, with a subsequent decline late at night.

Admission Minute

```
# Extract admission minute
admissions_minute <- admissions_tble %>%
  mutate(admission_minute = minute(admittime)) %>%
  group_by(admission_minute) %>%
  summarise(count = n())

# Static ggplot2 plot
p3 <- ggplot(admissions_minute, aes(x = admission_minute, y = count)) +
  geom_line(color = "green", size = 1) +
  labs(
    title = "Admission Minute Distribution",
    x = "Minute of Admission",
    y = "Number of Admissions"
  ) +
  theme_minimal()

# Convert to interactive plot
ggplotly(p3)
```



The "Admission Minute Distribution" graph displays peculiar peaks at specific minutes (approximately 0, 15, 30, and 45). These peaks are significantly higher than the baseline values, indicating a concentration of recorded admissions at these specific minutes. These peaks are likely artifacts of data recording practices, such as rounding or default values in the electronic health record system, rather than actual variations in admission frequency.

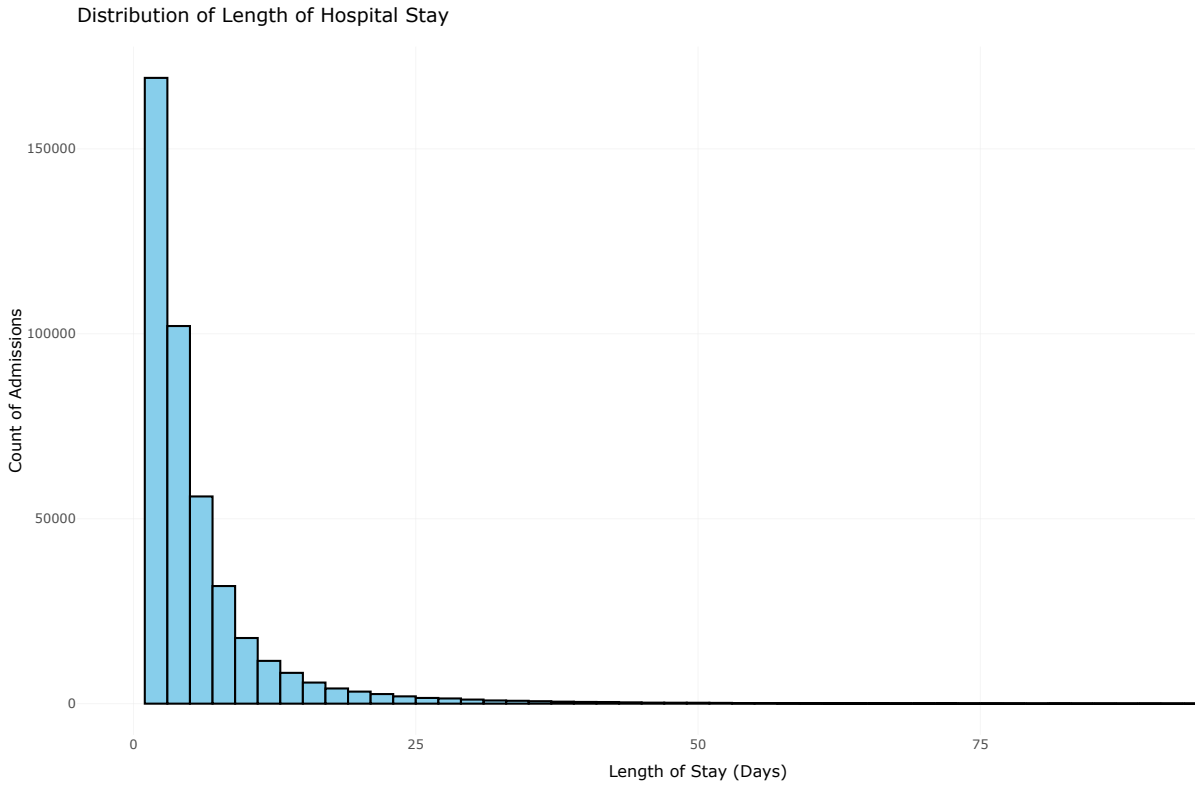
Length of hospital stay

```
# Required libraries


# Calculate length of stay in days
admissions_tble <- admissions_tble %>%
  mutate(
    length_of_stay = as.numeric(difftime(dischtime, admittime, units = "days"))
  )

# Plot a histogram for length of stay with x-axis limited to 0-100 days
plot <- ggplot(admissions_tble, aes(x = length_of_stay)) +
  geom_histogram(binwidth = 2, fill = "skyblue", color = "black") +
```

```
  xlim(0, 100) +  # Limit x-axis to 0-100 days
  labs(
    title = "Distribution of Length of Hospital Stay ",
    x = "Length of Stay (Days)",
    y = "Count of Admissions"
  ) +
  theme_minimal()

# Convert to interactive plot with plotly
interactive_plot <- ggplotly(plot)
```

```
Warning: Removed 417 rows containing non-finite outside the scale range
(`stat_bin()`).
```

```
# Show the plot
interactive_plot
```



The "Distribution of Length of Hospital Stay" histogram shows a strongly skewed distribution, with the highest frequency of admissions occurring at very short lengths of stay. The histogram

indicates that the majority of admissions result in brief hospitalizations, while a long tail on the right side of the graph signifies a smaller proportion of patients requiring extended stays exceeding 50 days, likely due to complex medical needs or surgical procedures.

- The high count of single admissions in the "Number of Admissions per Patient" graph directly corresponds to the high frequency of short stays in the "Distribution of Length of Hospital Stay" graph. This suggests that a large portion of single admissions are associated with brief hospitalizations.
- The significant midnight peak in the "Admission Hour Distribution" graph might contribute to the high number of single admissions and short stays, as these midnight admissions could be related to administrative procedures or short-term observations.
- The data recording anomalies in the "Admission Minute Distribution" graph raise concerns about the accuracy of time-related analyses, which could potentially affect the interpretation of length of stay and admission frequency data.

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
```

Q4.1 Ingestion

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

Solution 4.1

```
# Import patients.csv.gz as a tibble
patients_tble <- read_csv("~/mimic/hosp/patients.csv.gz") %>%
  as_tibble()
```

```
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.
```

```r
# Check the first few rows of the tibble
print(patients_tble)
```

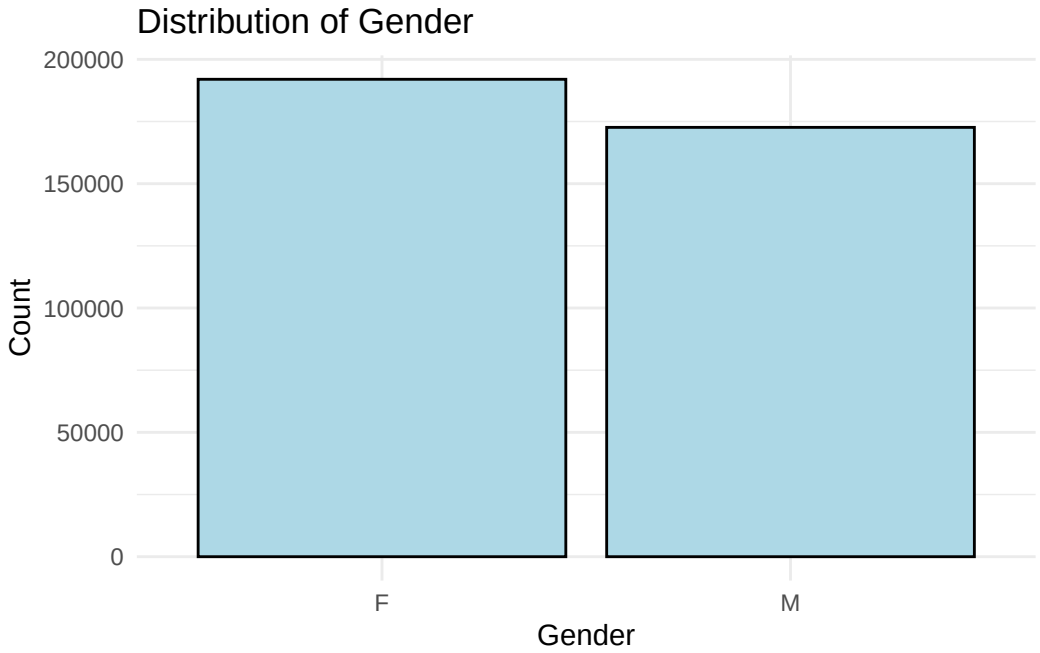
```
# A tibble: 364,627 x 6
   subject_id gender anchor_age anchor_year anchor_year_group dod
        <dbl> <chr>       <dbl>       <dbl> <chr>              <date>
 1   10000032 F              52        2180 2014 - 2016       2180-09-09
 2   10000048 F              23        2126 2008 - 2010       NA
 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
```

Q4.2 Summary and visualization

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

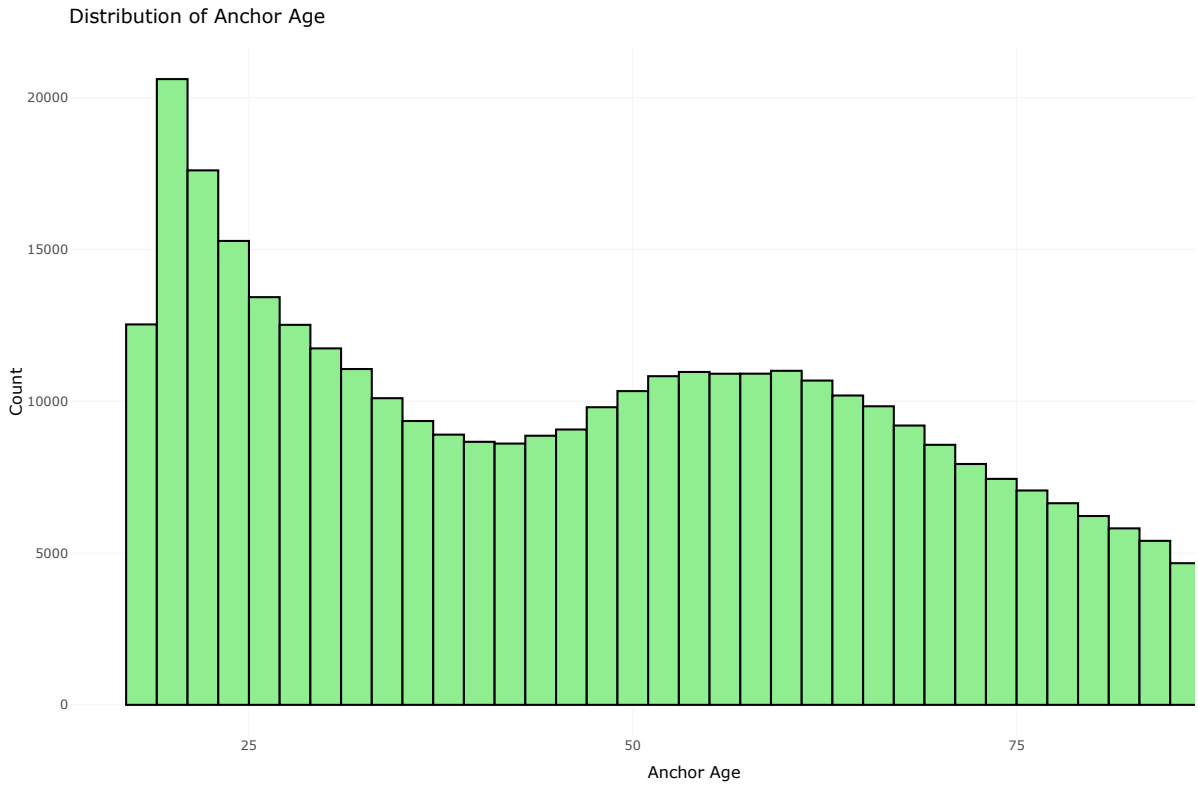
Solution 4.2

```r
# Load necessary libraries
library(ggplot2)

# Gender distribution
ggplot(patients_tble, aes(x = gender)) +
  geom_bar(fill = "lightblue", color = "black") +
  labs(title = "Distribution of Gender", x = "Gender", y = "Count") +
  theme_minimal()
```



The bar chart shows the distribution of gender. Here, the counts of females (F) and males (M) are displayed. While both genders are well-represented, there are slightly more females than males.

```
library(plotly)

# Create the histogram plot
p <- ggplot(patients_tble, aes(x = anchor_age)) +
  geom_histogram(binwidth = 2, fill = "lightgreen", color = "black") +
  labs(title = "Distribution of Anchor Age", x = "Anchor Age", y = "Count") +
  theme_minimal()

# Convert the ggplot to an interactive plotly plot
ggplotly(p)
```



- The younger peak could correspond to more frequent hospitalization of adults for acute conditions, accidents, or surgeries.
- The older peak reflects elderly patients often requiring ICU care for chronic conditions or age-related ailments.
- The spike at the right indicates the dataset's anonymization strategy for older patients, aggregating all patients above 89 into a single category to ensure privacy. The spike at age 89-100 likely results from an anonymization process used in MIMIC. For privacy reasons, patients over a certain age (usually around 89) are assigned an anchor age of 90+. This is done to reduce the risk of identifying older individuals who could be more easily recognized due to their age. Hence, the spike on the right represents a group of patients who are over 89 years old, but whose exact ages have been masked.

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
```

`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
```

We are interested in the lab measurements of creatinine (50912), potassium (50971), sodium (50983), chloride (50902), bicarbonate (50882), hematocrit (51221), white blood cell count (51301), and glucose (50931). Retrieve a subset of `labevents.csv.gz` that only containing these items for the patients in `icustays_tble`. Further restrict to the last available measurement (by `storetime`) before the ICU stay. The final `labevents_tble` should have one row per ICU stay and columns for each lab measurement. Hint: Use the Parquet format you generated in Homework 2. For reproducibility, make `labevents_pq` folder available at the current working directory `hw3`, for example, by a symbolic link.

```
> 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
```

Solution 5

```
icustays_tble <- read_csv("~/mimic/icu/icustays.csv.gz",
                          show_col_types = FALSE) %>%
  as_tibble()
```

```
dlabitems_tble <- read_csv("~/mimic/hosp/d_labitems.csv.gz",
                           show_col_types = FALSE) |>
  filter(itemid %in% c(
    50912,
    50971,
    50983,
    50902,
    50882,
    51221,
```

```
    51301,
    50931
  )) |>
  mutate(itemid = as.integer(itemid))
```

```
library(arrow)
library(duckdb)
```

```
Loading required package: DBI
```

```
library(dplyr)
library(tidyr)
library(stringr)

# Open the parquet dataset
labevents_tble <- open_dataset(labevents_pq_dir, format = "parquet")
```

```
labs_data <- labevents_tble |>
  to_duckdb() |>


  # Select necessary variables
  select(subject_id, itemid, storetime, valuenum) |>

  # Filter for itemid of interest
  filter(itemid %in% dlabitems_tble$itemid) |>

  # Join with icustays table to get intime
  left_join(
    select(icustays_tble, subject_id, stay_id, intime),
    by = c("subject_id"),
    copy = TRUE
  ) |>

  # Filter to keep only records before ICU intime
  filter(storetime < intime) |>

  # Group by subject_id, stay_id, and itemid
  group_by(subject_id, stay_id, itemid) |>

  # Keep only the last storetime for each item before intime
```

```r
  slice_max(storetime, n = 1) |>

  # Remove storetime and intime columns, ungroup the data
  select(-storetime, -intime) |>
  ungroup() |>

  # Pivot wider to make itemid names as columns
  pivot_wider(names_from = itemid, values_from = valuenum) |>

  # Rename columns based on dlabitems_tble labels
  rename_at(
    vars(as.character(dlabitems_tble$itemid)),
    ~str_to_lower(dlabitems_tble$label)
  ) |>

  # Rename specific columns (e.g., white blood cells)
  rename(wbc = `white blood cells`) |>

  # Show the query for debugging purposes
  show_query() |>

  # Collect the results into an R dataframe
  collect() |>
  relocate(subject_id,stay_id,bicarbonate,chloride,creatinine,glucose,
  potassium, sodium,hematocrit, wbc)|>

  # Arrange by subject_id and stay_id
  arrange(subject_id, stay_id)  %>%


  # Print the dataframe with full width
  print(labs_data,width=Inf)
```

```
<SQL>
SELECT
  subject_id,
  stay_id,
  MAX(CASE WHEN (itemid = 50983.0) THEN valuenum END) AS sodium,
  MAX(CASE WHEN (itemid = 50902.0) THEN valuenum END) AS chloride,
  MAX(CASE WHEN (itemid = 50882.0) THEN valuenum END) AS bicarbonate,
  MAX(CASE WHEN (itemid = 50931.0) THEN valuenum END) AS glucose,
  MAX(CASE WHEN (itemid = 51221.0) THEN valuenum END) AS hematocrit,
```

```
  MAX(CASE WHEN (itemid = 50971.0) THEN valuenum END) AS potassium,
  MAX(CASE WHEN (itemid = 51301.0) THEN valuenum END) AS wbc,
  MAX(CASE WHEN (itemid = 50912.0) THEN valuenum END) AS creatinine
FROM (
  SELECT subject_id, itemid, valuenum, stay_id
  FROM (
    SELECT
      q01.*,
      RANK() OVER (PARTITION BY subject_id, stay_id, itemid ORDER BY storetime DESC) AS col01
    FROM (
      SELECT LHS.*, stay_id, intime
      FROM (
        SELECT subject_id, itemid, storetime, valuenum
        FROM arrow_001
        WHERE (itemid IN (50882, 50902, 50912, 50931, 50971, 50983, 51221, 51301))
      ) LHS
      LEFT JOIN dbplyr_ZIbBSGuDkA
        ON (LHS.subject_id = dbplyr_ZIbBSGuDkA.subject_id)
    ) q01
    WHERE (storetime < intime)
  ) q01
  WHERE (col01 <= 1)
) q01
GROUP BY subject_id, stay_id
# A tibble: 88,086 x 10
   subject_id  stay_id bicarbonate chloride creatinine glucose potassium sodium
        <dbl>    <dbl>       <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   wbc
        <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 88,076 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
```

`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
```

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 6

```
# Load necessary tables
chartevents_tble <- open_dataset("./chartevents_pq.parque", format = "parquet")
```

```
# List of vital item IDs
vitals_itemids <- c(
  220045,  # Heart rate
  220179,  # Systolic non-invasive blood pressure
  220180,  # Diastolic non-invasive blood pressure
  223761,  # Body temperature (Fahrenheit)
  220210   # Respiratory rate
)

# Filter `chartevents` for the relevant itemids and join with `icustays_tble`
vitals_data <- chartevents_tble |>
  to_duckdb() |>

  # Select required variables
  select(subject_id, stay_id, itemid, charttime, value,valuenum,storetime) |>

  # Filter for vitals itemids of interest
  filter(itemid %in% vitals_itemids) |>

  # Join with icustays table to get intime
  left_join(
    select(icustays_tble, subject_id, stay_id, intime,outtime),
    by = c("subject_id", "stay_id"),
    copy = TRUE
  ) |>

  filter(storetime >= intime & storetime <= outtime) |>
  filter(!is.na(valuenum)) |>

  # Group by subject, stay, and itemid
  group_by(subject_id, stay_id, itemid) |>
  slice_min(storetime, with_ties = TRUE) |>

  # Calculate the average value for each itemid during the ICU stay
  summarise(mean = mean(valuenum, na.rm = TRUE)) |>

  # Ungroup
```

```
  ungroup() |>

  # Pivot to wider format to make itemid names as columns
  pivot_wider(names_from = itemid, values_from = mean) |>

  # Rename columns to meaningful names
  rename(
    heart_rate = `220045`,
    non_invasive_bloodpressue_systolic = `220179`,
    non_invasive_bloodpressure_diastolic = `220180`,
    temperature_farhenheit = `223761`,
    respiratory_rate = `220210`
  ) |>
  mutate(
    heart_rate = round(heart_rate, 1),
    non_invasive_bloodpressue_systolic =
      round(non_invasive_bloodpressue_systolic, 1),
    non_invasive_bloodpressure_diastolic =
      round(non_invasive_bloodpressure_diastolic, 1),
    temperature_farhenheit = round(temperature_farhenheit, 1),
    respiratory_rate = round(respiratory_rate, 1)
  ) |>

  # Collect the results
  collect() |>
  relocate(subject_id,stay_id,heart_rate,non_invasive_bloodpressure_diastolic,
  non_invasive_bloodpressue_systolic, respiratory_rate,temperature_farhenheit)|>

  # Arrange by subject_id and stay_id
  arrange(subject_id, stay_id) |>

  # Print with full width
  print(width = Inf)
```

```
`summarise()` has grouped output by "subject_id" and "stay_id". You can
override using the `.groups` argument.
`summarise()` has grouped output by "subject_id" and "stay_id". You can
override using the `.groups` argument.

# A tibble: 94,363 x 7
   subject_id  stay_id heart_rate non_invasive_bloodpressure_diastolic
```

```
        <dbl>    <dbl>      <dbl>                                  <dbl>
 1   10000032 39553978       91                                     48
 2   10000690 37081114       78                                     56.5
 3   10000980 39765666       76                                    102
 4   10001217 34592300       79.3                                   93.3
 5   10001217 37067082       86                                     90
 6   10001725 31205490       86                                     56
 7   10001843 39698942      124.                                    78
 8   10001884 37510196       49                                     30.5
 9   10002013 39060235       80                                     62
10   10002114 34672098      110.                                    80
   non_invasive_bloodpressue_systolic respiratory_rate temperature_farhenheit
                                <dbl>            <dbl>                  <dbl>
 1                               84               24                     98.7
 2                              106               24.3                   97.7
 3                              154               23.5                   98
 4                              156               14                     97.6
 5                              151               18                     98.5
 6                               73               19                     97.7
 7                              110               16.5                   97.9
 8                              174.              13                     98.1
 9                               98.5             14                     97.2
10                              112               21                     97.9
# i 94,353 more rows
```

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 7

```r
mimic_icu_cohort <- icustays_tble %>%
  left_join(patients,by = "subject_id")  %>%
  left_join(admissions
            , by = c("hadm_id","subject_id")) %>%
  left_join(vitals_data,by = c("subject_id","stay_id")) %>%
  left_join(labs_data,by=c("subject_id","stay_id"))%>%
  mutate(
    intime_year = year(intime),  # Extract year from ICU admission time
    age_intime = anchor_age + (intime_year - anchor_year)
  ) %>%

  # Optionally remove the intermediate intime_year column
  select(-intime_year)

print(mimic_icu_cohort,width=Inf)
```

```
# A tibble: 94,458 x 41
   subject_id  hadm_id  stay_id first_careunit
        <dbl>    <dbl>    <dbl> <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 gender anchor_age anchor_year anchor_year_group
   <dttm>              <dbl> <chr>       <int>       <int> <chr>
 1 2180-07-23 23:50:47 0.410 F              52        2180 2014 - 2016
 2 2150-11-06 17:03:17 3.89  F              86        2150 2008 - 2010
 3 2189-06-27 20:38:27 0.498 F              73        2186 2008 - 2010
 4 2157-11-21 22:08:00 1.12  F              55        2157 2011 - 2013
 5 2157-12-20 14:27:41 0.948 F              55        2157 2011 - 2013
 6 2110-04-12 23:59:56 1.34  F              46        2110 2011 - 2013
 7 2134-12-06 14:38:26 0.825 M              73        2131 2017 - 2019
 8 2131-01-20 08:27:30 9.17  F              68        2122 2008 - 2010
 9 2160-05-19 17:33:33 1.31  F              53        2156 2008 - 2010
10 2162-02-20 21:16:27 2.91  M              56        2162 2020 - 2022
   dod          admittime           dischtime           deathtime
   <chr>        <chr>               <chr>               <chr>
 1 "2180-09-09" 2180-07-23 12:35:00 2180-07-25 17:55:00 ""
 2 "2152-01-30" 2150-11-02 18:02:00 2150-11-12 13:45:00 ""
 3 "2193-08-26" 2189-06-27 07:38:00 2189-07-03 03:00:00 ""
 4 ""           2157-11-18 22:56:00 2157-11-25 18:00:00 ""
 5 ""           2157-12-18 16:58:00 2157-12-24 14:55:00 ""
 6 ""           2110-04-11 15:08:00 2110-04-14 15:00:00 ""
 7 "2134-12-06" 2134-12-05 00:10:00 2134-12-06 12:54:00 "2134-12-06 12:54:00"
 8 "2131-01-20" 2131-01-07 20:39:00 2131-01-20 05:15:00 "2131-01-20 05:15:00"
 9 ""           2160-05-18 07:45:00 2160-05-23 13:30:00 ""
10 "2162-12-11" 2162-02-17 22:32:00 2162-03-04 15:16:00 ""
   admission_type              admit_provider_id admission_location
   <chr>                       <chr>             <chr>
 1 EW EMER.                    P06OTX            EMERGENCY ROOM
 2 EW EMER.                    P26QQ4            EMERGENCY ROOM
 3 EW EMER.                    P06OTX            EMERGENCY ROOM
 4 EW EMER.                    P3610N            EMERGENCY ROOM
```

```
 5 DIRECT EMER.                P276OU            PHYSICIAN REFERRAL
 6 EW EMER.                    P32W56            PACU
 7 URGENT                      P67ATB            TRANSFER FROM HOSPITAL
 8 OBSERVATION ADMIT           P49AFC            EMERGENCY ROOM
 9 SURGICAL SAME DAY ADMISSION P8286C            PHYSICIAN REFERRAL
10 OBSERVATION ADMIT           P46834            PHYSICIAN REFERRAL
   discharge_location insurance language marital_status race
   <chr>              <chr>     <chr>    <chr>          <chr>
 1 HOME               Medicaid  English  "WIDOWED"      WHITE
 2 REHAB              Medicare  English  "WIDOWED"      WHITE
 3 HOME HEALTH CARE   Medicare  English  "MARRIED"      BLACK/AFRICAN AMERICAN
 4 HOME HEALTH CARE   Private   Other    "MARRIED"      WHITE
 5 HOME HEALTH CARE   Private   Other    "MARRIED"      WHITE
 6 HOME               Private   English  "MARRIED"      WHITE
 7 DIED               Medicare  English  "SINGLE"       WHITE
 8 DIED               Medicare  English  "MARRIED"      BLACK/AFRICAN AMERICAN
 9 HOME HEALTH CARE   Medicare  English  "SINGLE"       OTHER
10 HOME HEALTH CARE   Medicaid  English  ""             UNKNOWN
   edregtime             edouttime             hospital_expire_flag heart_rate
   <chr>                 <chr>                                <int>      <dbl>
 1 "2180-07-23 05:54:00" "2180-07-23 14:00:00"                    0       91
 2 "2150-11-02 11:41:00" "2150-11-02 19:37:00"                    0       78
 3 "2189-06-27 06:25:00" "2189-06-27 08:42:00"                    0       76
 4 "2157-11-18 17:38:00" "2157-11-19 01:24:00"                    0       86
 5 ""                    ""                                       0       79.3
 6 ""                    ""                                       0       86
 7 ""                    ""                                       1      124.
 8 "2131-01-07 13:36:00" "2131-01-07 22:13:00"                    1       49
 9 ""                    ""                                       0       80
10 "2162-02-17 19:35:00" "2162-02-17 23:30:00"                    0      110.
   non_invasive_bloodpressure_diastolic non_invasive_bloodpressue_systolic
                                  <dbl>                               <dbl>
 1                                 48                                  84
 2                                 56.5                               106
 3                                102                                 154
 4                                 90                                 151
 5                                 93.3                               156
 6                                 56                                  73
 7                                 78                                 110
 8                                 30.5                               174.
 9                                 62                                  98.5
10                                 80                                 112
   respiratory_rate temperature_farhenheit bicarbonate chloride creatinine
```

```
             <dbl>                  <dbl>       <dbl>    <dbl>       <dbl>
 1            24                     98.7          25       95         0.7
 2            24.3                   97.7          26      100         1
 3            23.5                   98            21      109         2.3
 4            18                     98.5          22      108         0.6
 5            14                     97.6          30      104         0.5
 6            19                     97.7          NA       98        NA
 7            16.5                   97.9          28       97         1.3
 8            13                     98.1          30       88         1.1
 9            14                     97.2          24      102         0.9
10            21                     97.9          18       NA         3.1
   glucose potassium sodium hematocrit   wbc age_intime
     <dbl>     <dbl>  <dbl>      <dbl> <dbl>      <dbl>
 1     102       6.7    126       41.1   6.9         52
 2      85       4.8    137       36.1   7.1         86
 3      89       3.9    144       27.3   5.3         76
 4     112       4.2    142       38.1  15.7         55
 5      87       4.1    142       37.4   5.4         55
 6      NA       4.1    139       NA    NA           46
 7     131       3.9    138       31.4  10.4         76
 8     141       4.5    130       39.7  12.2         77
 9     288       3.5    137       34.9   7.2         57
10      95       6.5    125       34.3  16.8         56
# i 94,448 more rows
```

Q8. Exploratory data analysis (EDA)

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

Solution 8.1

- Length of ICU stay `los` vs demographic variables (race, insurance, marital_status, gender, age at intime)

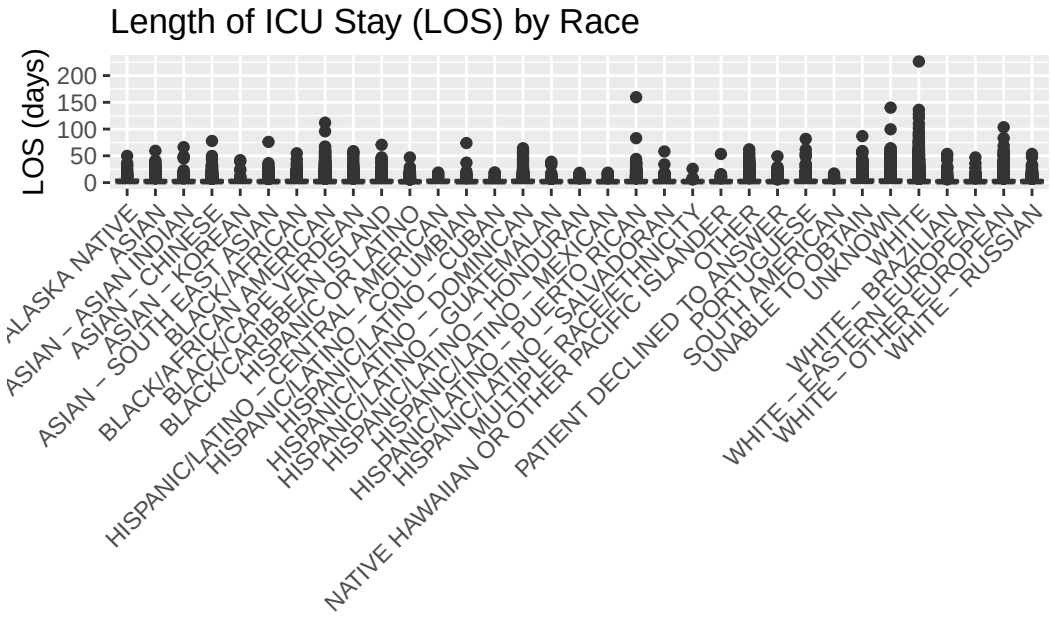
```
# Load necessary libraries
library(ggplot2)
library(dplyr)

# Create summary statistics for the ICU cohort
summary_stats <- mimic_icu_cohort %>%
```

```
  group_by(race, insurance, marital_status, gender) %>%
  summarise(
    mean_los = mean(los, na.rm = TRUE),
    median_los = median(los, na.rm = TRUE),
    sd_los = sd(los, na.rm = TRUE),
    IQR_los = IQR(los, na.rm = TRUE),
    .groups = 'drop'
  )

# Print summary statistics
print(summary_stats)
```

```
# A tibble: 1,078 x 8
   race       insurance marital_status gender mean_los median_los sd_los IQR_los
   <chr>      <chr>     <chr>          <chr>     <dbl>      <dbl>  <dbl>   <dbl>
 1 AMERICAN ~ ""        "DIVORCED"     M          2.44       2.44 NA       0
 2 AMERICAN ~ ""        "MARRIED"      M          6.30       6.30  6.52    4.61
 3 AMERICAN ~ "Medicai~ ""             F         16.9       16.9  NA       0
 4 AMERICAN ~ "Medicai~ "DIVORCED"     F         13.4       13.4  17.8    12.6
 5 AMERICAN ~ "Medicai~ "DIVORCED"     M          2.41       1.22  2.00    2.59
 6 AMERICAN ~ "Medicai~ "MARRIED"      F          2.08       2.08  1.29    0.914
 7 AMERICAN ~ "Medicai~ "MARRIED"      M          8.80       8.25  8.39   10.6
 8 AMERICAN ~ "Medicai~ "SINGLE"       F          1.45       1.33  0.738   1.00
 9 AMERICAN ~ "Medicai~ "SINGLE"       M          3.07       1.06  4.45    1.11
10 AMERICAN ~ "Medicar~ ""             F          3.12       4.11  1.74    1.51
# i 1,068 more rows
```

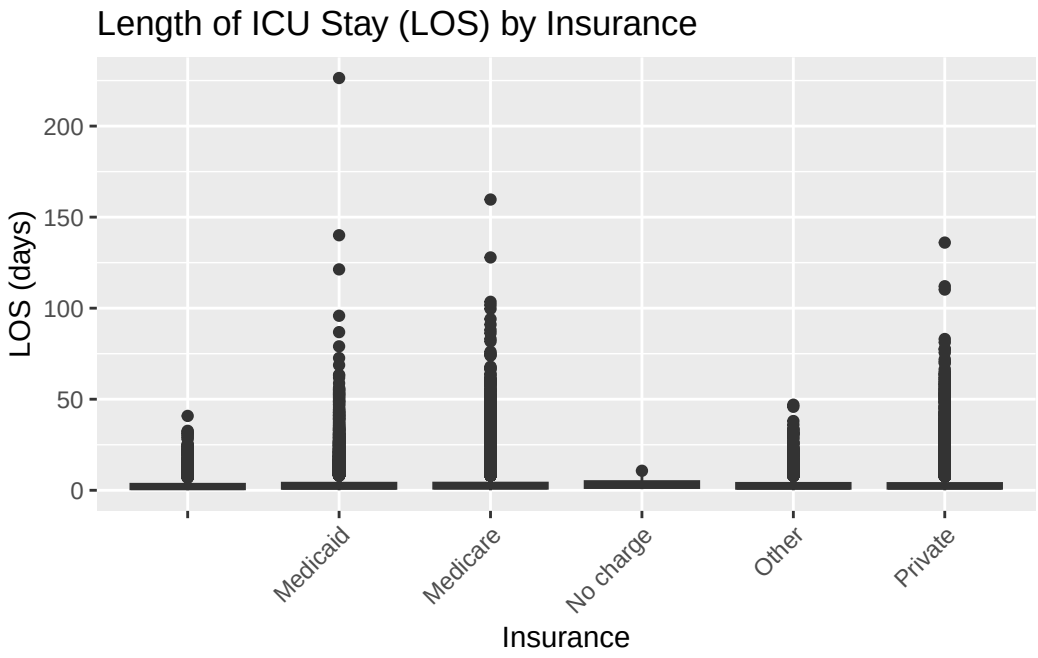
```
# Plot 1: Boxplot of los vs race
ggplot(mimic_icu_cohort, aes(x = race, y = los)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Length of ICU Stay (LOS) by Race", x = "Race",
       y = "LOS (days)")
```

```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```



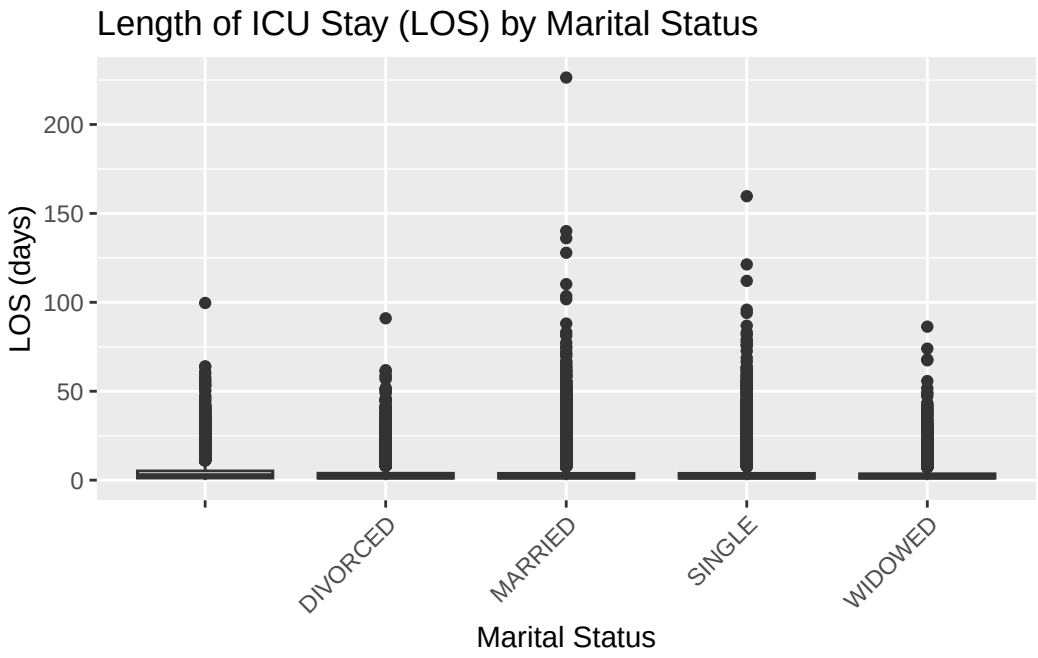

```
# Plot 2: Boxplot of los vs insurance
ggplot(mimic_icu_cohort, aes(x = insurance, y = los)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Length of ICU Stay (LOS) by Insurance",
       x = "Insurance", y = "LOS (days)")
```

```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```



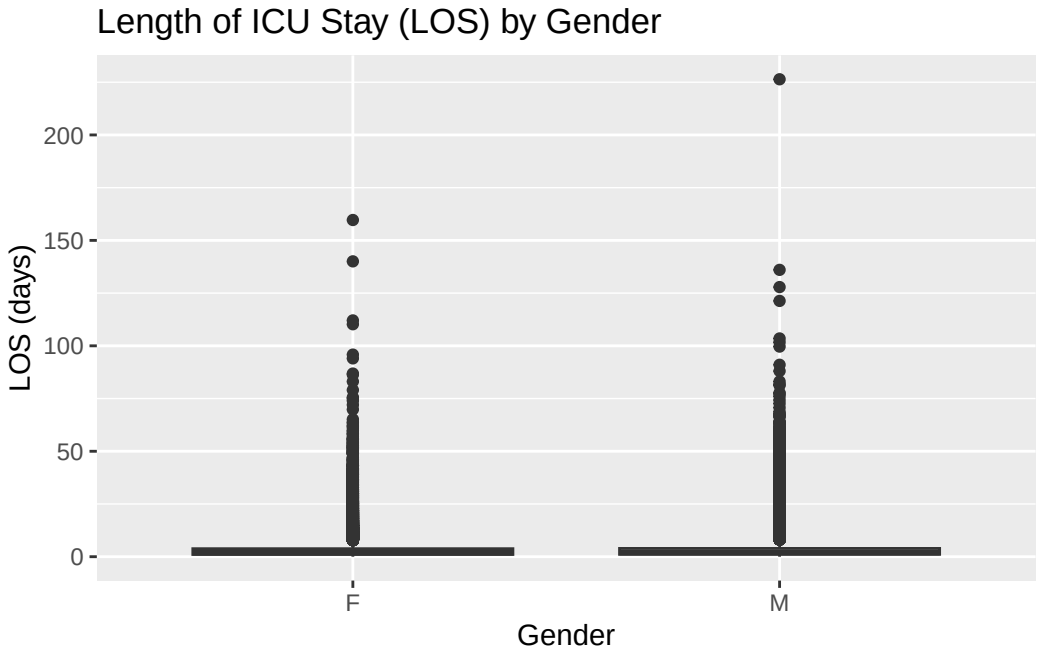

```
# Plot 3: Boxplot of los vs marital status
ggplot(mimic_icu_cohort, aes(x = marital_status, y = los)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Length of ICU Stay (LOS) by Marital Status",
       x = "Marital Status", y = "LOS (days)")
```

```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```



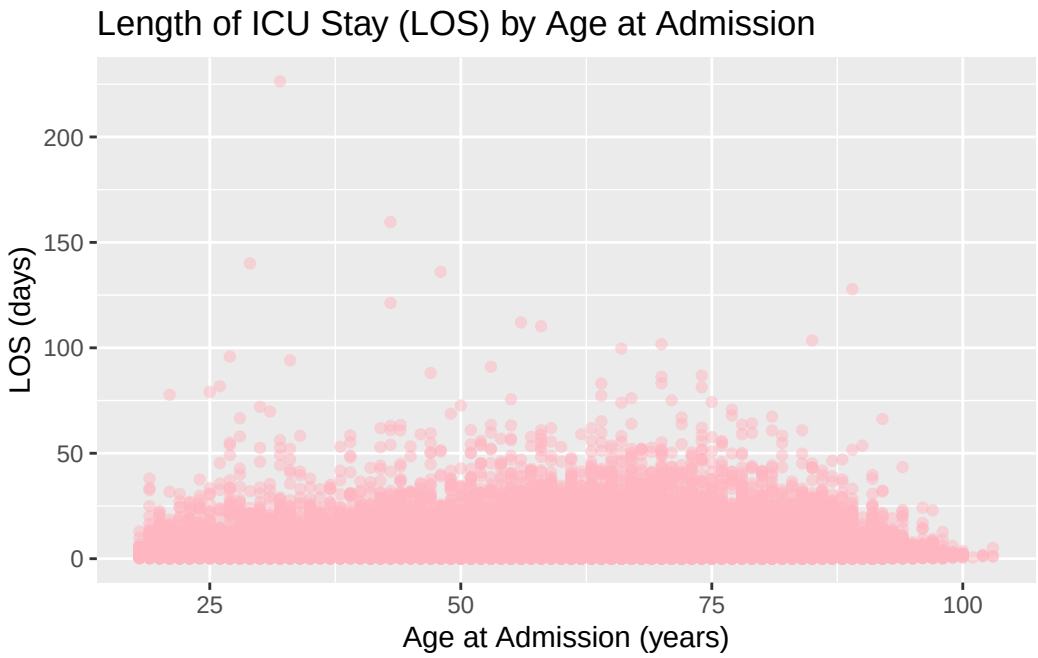

```
# Plot 4: Boxplot of los vs gender
ggplot(mimic_icu_cohort, aes(x = gender, y = los)) +
  geom_boxplot() +
  labs(title = "Length of ICU Stay (LOS) by Gender",
       x = "Gender", y = "LOS (days)")
```

```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```




```
# Plot 5: Scatter plot of los vs age at intime (age_intime)
ggplot(mimic_icu_cohort, aes(x = age_intime, y = los)) +
  geom_point(color="lightpink",alpha = 0.5) +
  labs(title = "Length of ICU Stay (LOS) by Age at Admission",
       x = "Age at Admission (years)", y = "LOS (days)")
```

```
Warning: Removed 14 rows containing missing values or values outside the scale range
(`geom_point()`).
```

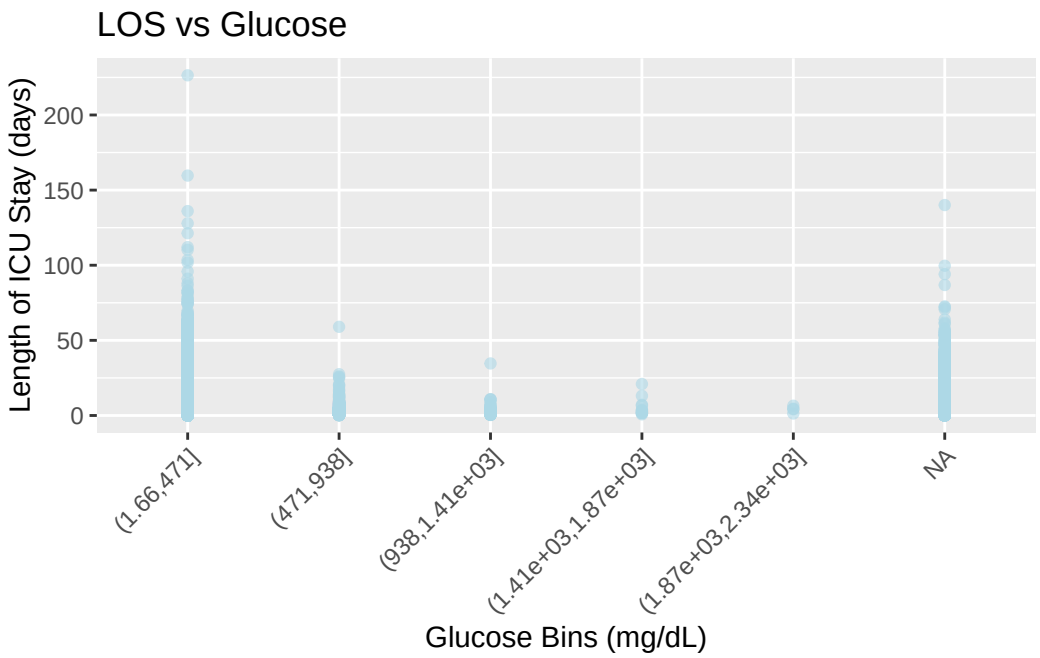


Solution 8.2

- Length of ICU stay `los` vs the last available lab measurements before ICU stay

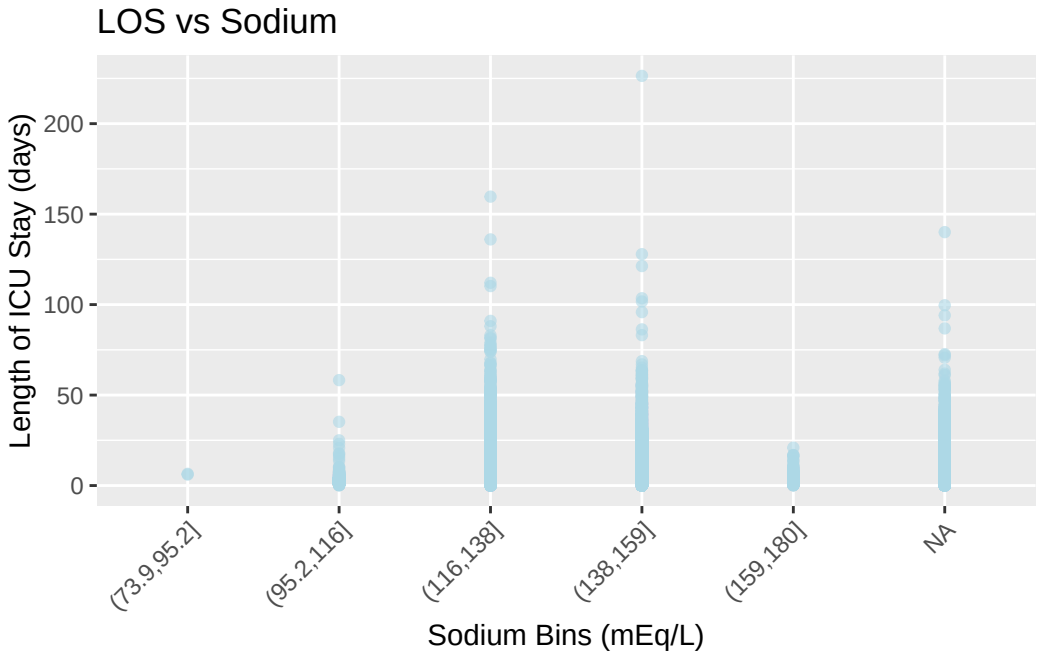
```
# Load necessary libraries
library(ggplot2)
library(dplyr)

# Binning lab measurements for box plot representation
mimic_icu_cohort <- mimic_icu_cohort %>%
  mutate(glucose_bin = cut(glucose, breaks = 5),
         sodium_bin = cut(sodium, breaks = 5),
         potassium_bin = cut(potassium, breaks = 5),
         wbc_bin = cut(wbc, breaks = 5),
         creatinine_bin = cut(creatinine, breaks = 5))

# Plot 1: Boxplot of LOS vs Glucose bins
ggplot(mimic_icu_cohort, aes(x = glucose_bin, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se = FALSE, color = "blue") +
  labs(title = "LOS vs Glucose", x = "Glucose Bins (mg/dL)",
       y = "Length of ICU Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
`geom_smooth()` using formula = 'y ~ x'

Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_smooth()`).

Warning: Removed 14 rows containing missing values or values outside the scale range
(`geom_point()`).
```




```r
# Plot 2: Boxplot of LOS vs Sodium bins
ggplot(mimic_icu_cohort, aes(x = sodium_bin, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se = FALSE, color = "blue") +
  labs(title = "LOS vs Sodium", x = "Sodium Bins (mEq/L)",
       y = "Length of ICU Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

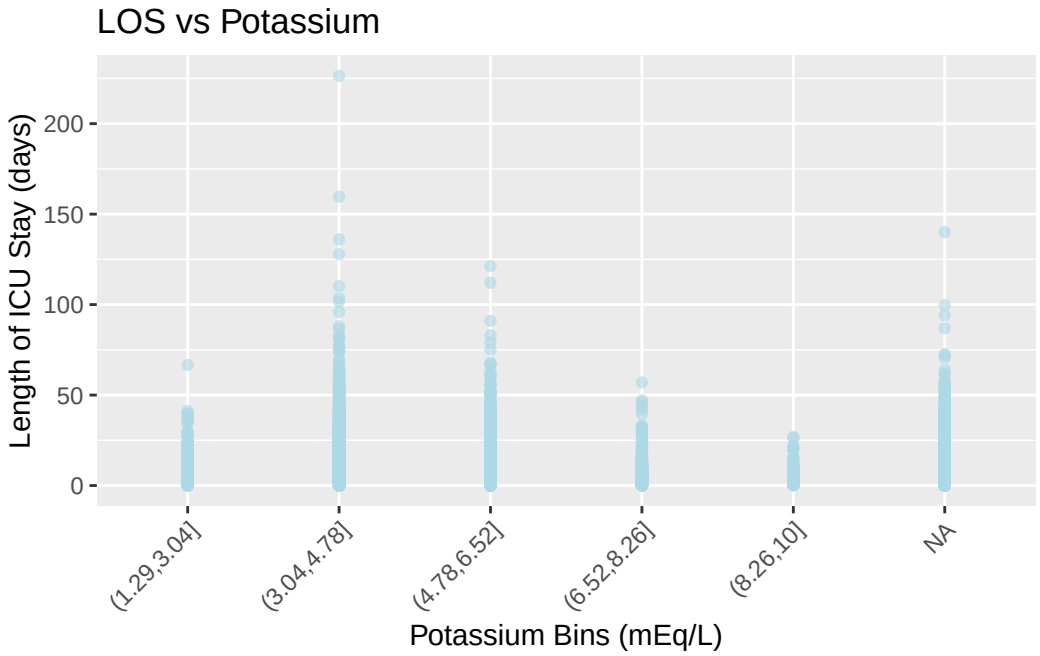
```
`geom_smooth()` using formula = 'y ~ x'

Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_smooth()`).
Removed 14 rows containing missing values or values outside the scale range
(`geom_point()`).
```




```
# Plot 3: Boxplot of LOS vs Potassium bins
ggplot(mimic_icu_cohort, aes(x = potassium_bin, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se = FALSE, color = "blue") +
  labs(title = "LOS vs Potassium", x = "Potassium Bins (mEq/L)",
       y = "Length of ICU Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
`geom_smooth()` using formula = 'y ~ x'
```

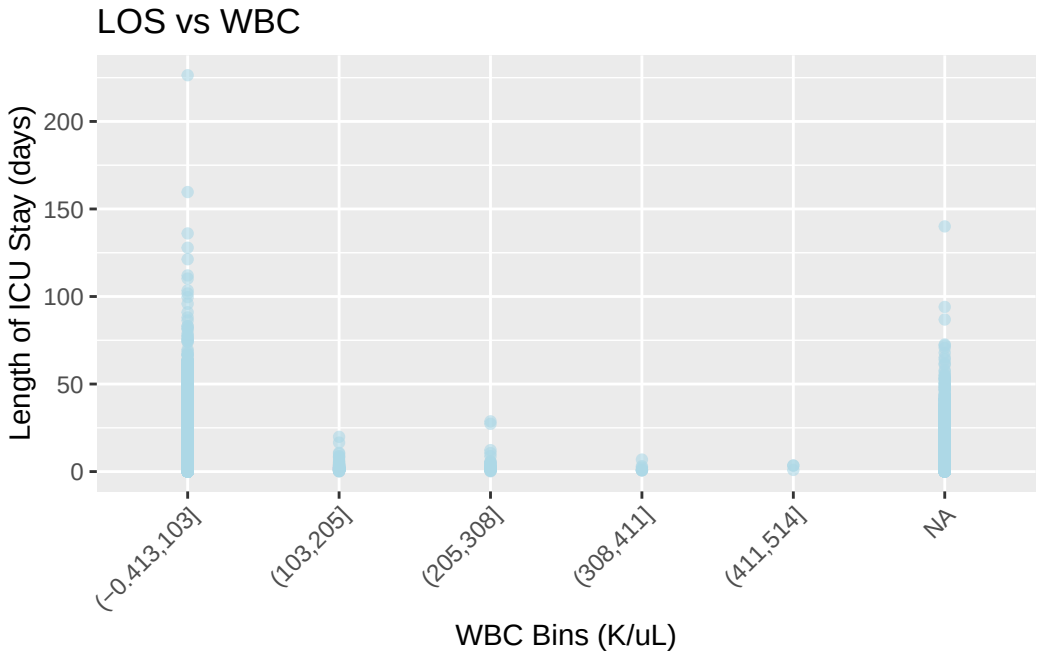
```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_smooth()`).
Removed 14 rows containing missing values or values outside the scale range
(`geom_point()`).
```




```
# Plot 4: Boxplot of LOS vs WBC bins
ggplot(mimic_icu_cohort, aes(x = wbc_bin, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se = FALSE, color = "blue") +
  labs(title = "LOS vs WBC", x = "WBC Bins (K/uL)",
       y = "Length of ICU Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
`geom_smooth()` using formula = 'y ~ x'
```

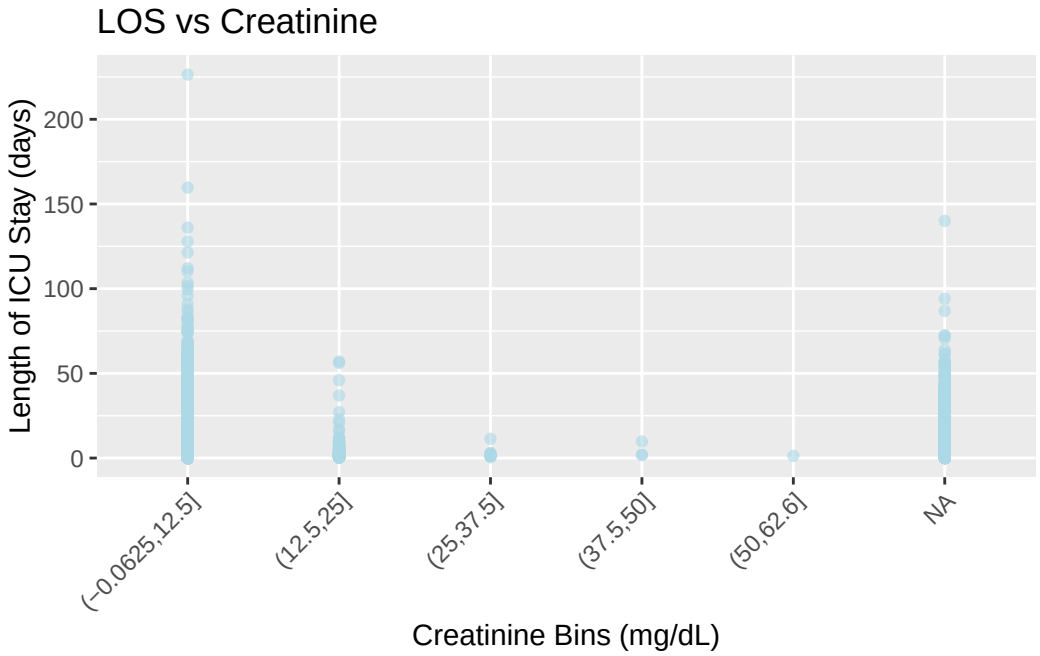
```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_smooth()`).
Removed 14 rows containing missing values or values outside the scale range
(`geom_point()`).
```




```
# Plot 5: Boxplot of LOS vs Creatinine bins
ggplot(mimic_icu_cohort, aes(x = creatinine_bin, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se = FALSE, color = "blue") +
  labs(title = "LOS vs Creatinine", x = "Creatinine Bins (mg/dL)",
       y = "Length of ICU Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
`geom_smooth()` using formula = 'y ~ x'
```

```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_smooth()`).
Removed 14 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Solution 8.3

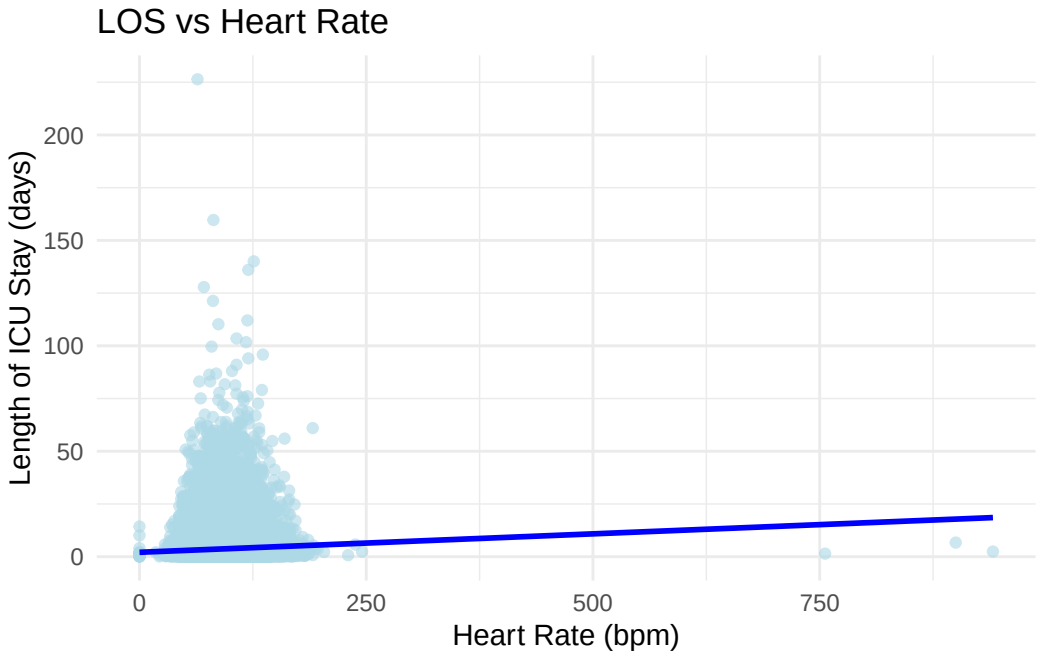
- Length of ICU stay `los` vs the first vital measurements within the ICU stay

```
# Load necessary libraries
library(ggplot2)

# Plot 1: Scatter plot of LOS vs Heart Rate
ggplot(mimic_icu_cohort, aes(x = heart_rate, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se = FALSE, color = "blue") +
  labs(title = "LOS vs Heart Rate", x = "Heart Rate (bpm)",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

```
Warning: Removed 100 rows containing non-finite outside the scale range
(`stat_smooth()`).
```

```
Warning: Removed 100 rows containing missing values or values outside the scale range
(`geom_point()`).
```

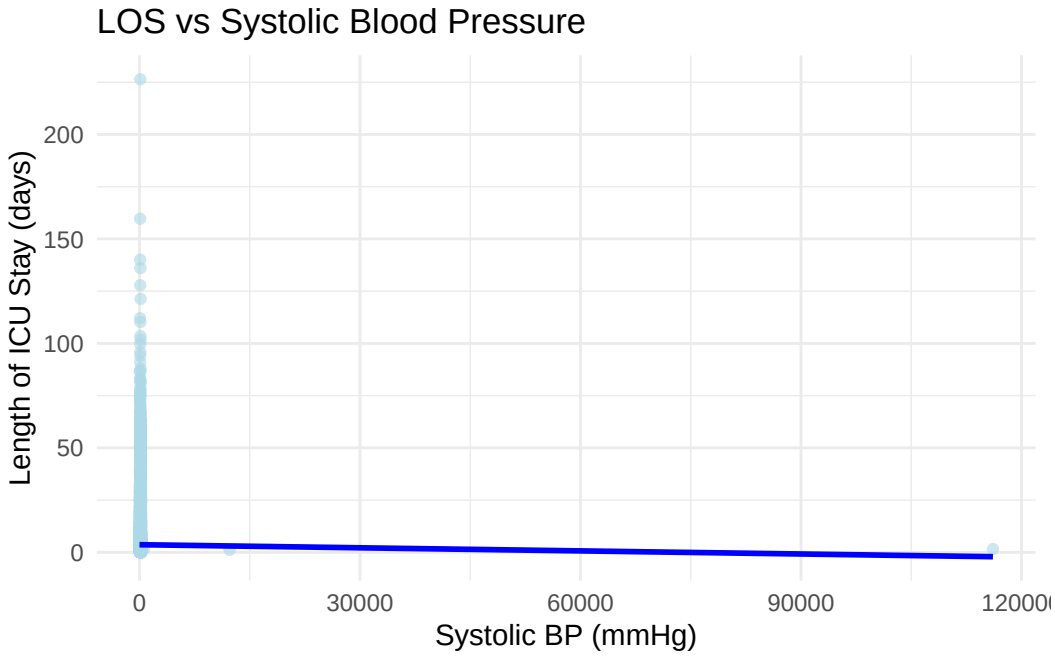



```
# Plot 2: Scatter plot of LOS vs Systolic Blood Pressure
ggplot(mimic_icu_cohort, aes(x = non_invasive_bloodpressue_systolic, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se= FALSE, color = "blue") +
  labs(title = "LOS vs Systolic Blood Pressure", x = "Systolic BP (mmHg)",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

```
Warning: Removed 1384 rows containing non-finite outside the scale range
(`stat_smooth()`).
```

```
Warning: Removed 1384 rows containing missing values or values outside the scale range
(`geom_point()`).
```

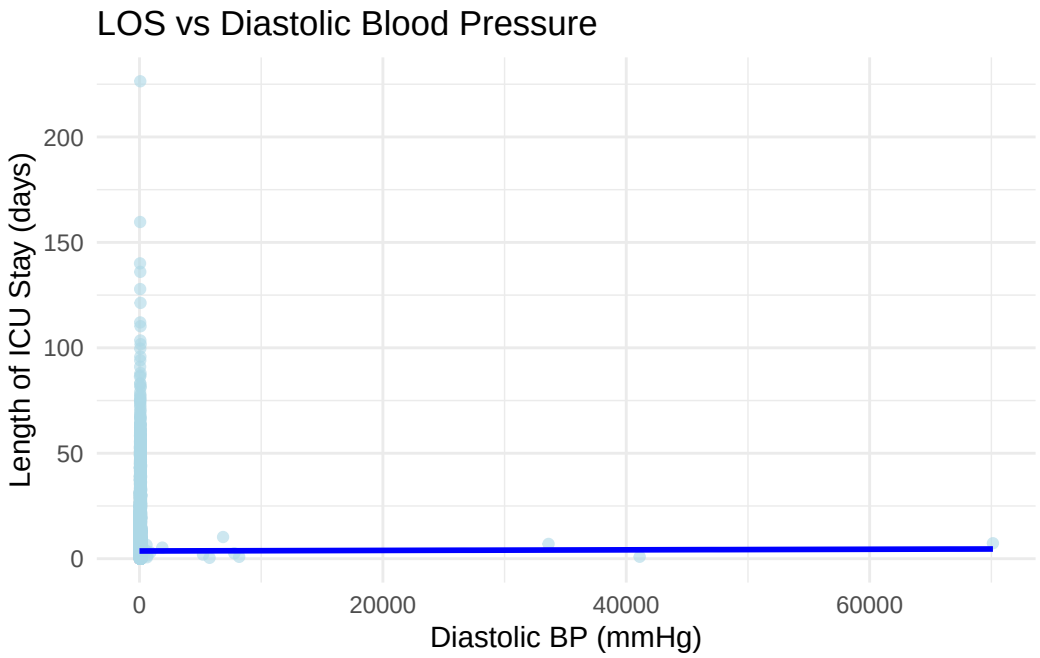



```
# Plot 3: Scatter plot of LOS vs Diastolic Blood Pressure
ggplot(mimic_icu_cohort,
       aes(x = non_invasive_bloodpressure_diastolic, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se=FALSE, color = "blue") +
  labs(title = "LOS vs Diastolic Blood Pressure",
       x = "Diastolic BP (mmHg)",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

```
Warning: Removed 1389 rows containing non-finite outside the scale range
(`stat_smooth()`).
```

```
Warning: Removed 1389 rows containing missing values or values outside the scale range
(`geom_point()`).
```

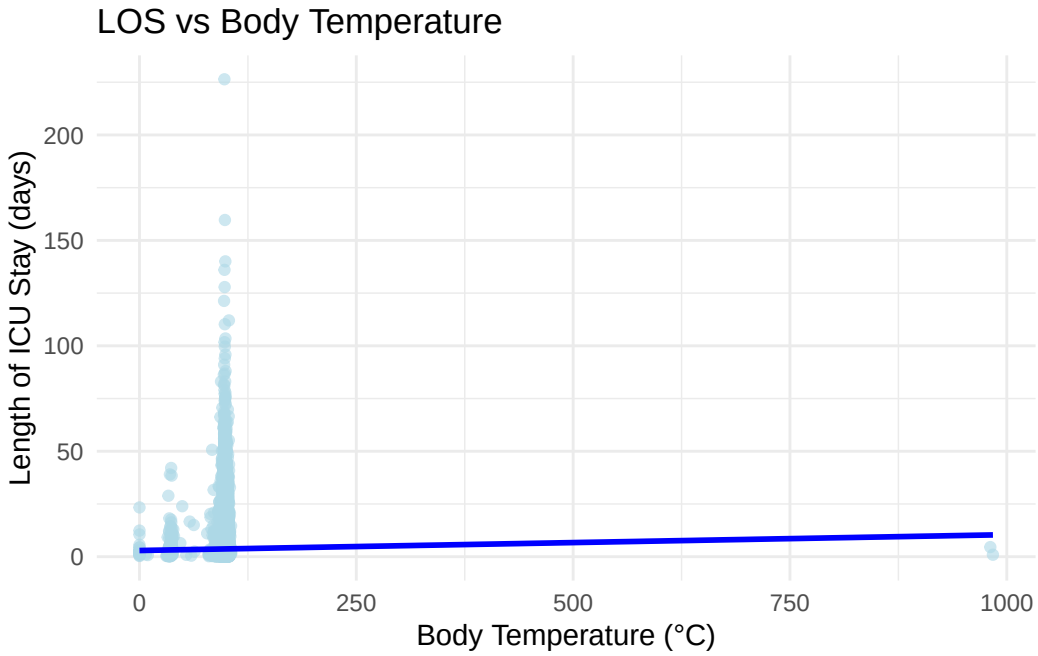



```
# Plot 4: Scatter plot of LOS vs Body Temperature
ggplot(mimic_icu_cohort, aes(x = temperature_farhenheit, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se=FALSE, color = "blue") +
  labs(title = "LOS vs Body Temperature", x = "Body Temperature (°C)",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

```
Warning: Removed 1689 rows containing non-finite outside the scale range
(`stat_smooth()`).
```

```
Warning: Removed 1689 rows containing missing values or values outside the scale range
(`geom_point()`).
```

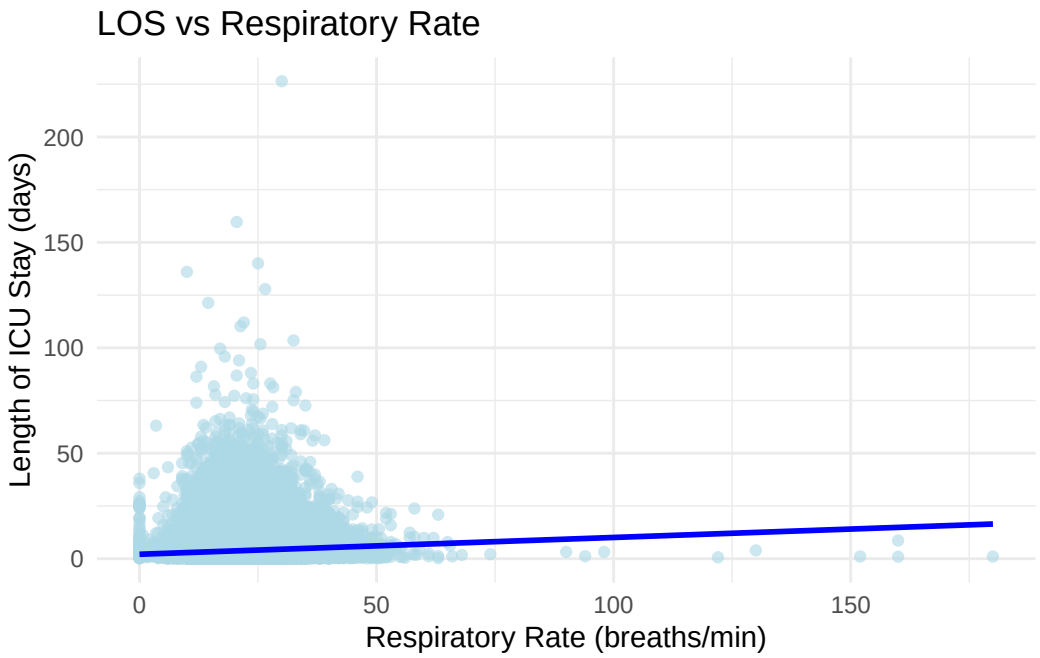



```
# Plot 5: Scatter plot of LOS vs Respiratory Rate
ggplot(mimic_icu_cohort, aes(x = respiratory_rate, y = los)) +
  geom_point(color="lightblue",alpha = 0.6) +
  geom_smooth(method = "lm",se=FALSE, color = "blue") +
  labs(title = "LOS vs Respiratory Rate", x = "Respiratory Rate (breaths/min)",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

```
Warning: Removed 212 rows containing non-finite outside the scale range
(`stat_smooth()`).
```

```
Warning: Removed 212 rows containing missing values or values outside the scale range
(`geom_point()`).
```

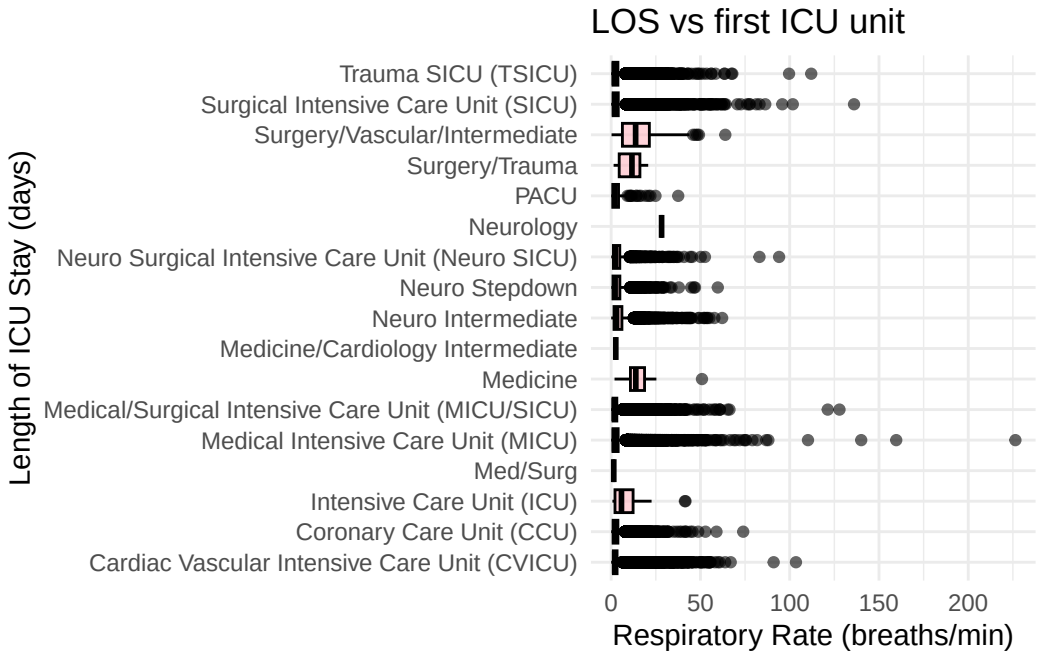


Solution 8.4

- Length of ICU stay `los` vs first ICU unit

```
ggplot(mimic_icu_cohort, aes(x = los, y = first_careunit)) +
  geom_boxplot(color="black",fill="lightpink",alpha = 0.6) +

  labs(title = "LOS vs first ICU unit", x = "Respiratory Rate (breaths/min)",
       y = "Length of ICU Stay (days)") +
  theme_minimal()
```

```
Warning: Removed 14 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```


LOS vs first ICU unit
Length of ICU Stay (days)
Trauma SICU (TSICU)
Surgical Intensive Care Unit (SICU)
Surgery/Vascular/Intermediate
Surgery/Trauma
PACU
Neurology
Neuro Surgical Intensive Care Unit (Neuro SICU)
Neuro Stepdown
Neuro Intermediate
Medicine/Cardiology Intermediate
Medicine
Medical/Surgical Intensive Care Unit (MICU/SICU)
Medical Intensive Care Unit (MICU)
Med/Surg
Intensive Care Unit (ICU)
Coronary Care Unit (CCU)
Cardiac Vascular Intensive Care Unit (CVICU)
0
50
100
150
200
Respiratory Rate (breaths/min)