SIT741 Assignment 1

Unit Chair: Wei Luo

Due: 12 April 2019

Assignment 1 contributes to 25% of your final SIT741 mark. The full mark is 25. It must be completed individually, and submitted to CloudDeakin before the due date: 11 pm, 12/04/2019 (Week 6 Friday).

Learning goals

In this assignment, you will work on a real-world problem to consolidate your learning in the first five weeks, including organise your data as tidy data and perform simple statistical analyses. This activity also serves as scaffolding for the upcoming Assignment 2.

Please start early so that you can identify any skill/knowledge gap and seek support from the teaching staff and other students.

Background

In Australia, we have experienced extreme heat in the year 2019. With the inevitable rise of extreme weather events, it is crucial that we better understand its potential impact on our everyday life.

In November 2016, a storm in Victoria triggered an unexpected surge of emergency department visits at the local public hospitals. Some consequences of this weather event were captured in this news article:

`http://bit.ly/2gC8j6U`

Apart from such storms, various weather events may affect the demand for care at our emergency departments (EDs). In SIT741, you will use publicly available data to understand the relationship between weather patterns and ED demands. Your analysis could provide crucial knowledge for resource planning at our health care systems.

Assignment 1 will focus on the analysis of ED demand data.

Task 1: Obtaining ED demand data (4 points)

First, let's find data measuring ED demands. We will use the *emergency departments admissions and attendances* data set provided by the Department of Health of Western Australia:

`http://data.gov.au/dataset/emergency-department-admissisons-and-attendances`

Task 1.1 Download the data set using the link below.

`http://bit.ly/2nkCUEh`

Manual download is OK.

Task 1.2 Answer the following questions:

- How many rows are in the data?
- What data types are in the data?
- What time period does the data cover?
- What's the difference between "Attendance" and "Admissions"?
- What do the variables `Tri_1`, `Tri_2`, ... represent?

Hint: You may need to consult the relevant background document, for example, the government webpage here: https://ww2.health.wa.gov.au/About-us/Policy-frameworks/Information-Management/Mandatory-requirements/Emergency-Department-and-Emergency-Services-Patient-Level-Data-Collection-and-Reporting .

```
library(tidyverse)
library(lubridate)
heading <- names(read_csv("govhack3.csv", n_max = 0))
ed_att <- read_csv("govhack3.csv", skip = 1)
ed_att <- ed_att %>%
  filter(!is.na(Date))

# How many rows?
ed_att %>%
  count()
```

```
FALSE # A tibble: 1 x 1
FALSE       n
FALSE   <int>
FALSE 1   365
```

Variables types include datetime and count. It is encoded as 'chr' and 'int'; 'chr' need to be processed.

```
# What time period?

ed_att <- ed_att %>%
  mutate(Date = dmy(Date))

ed_att %>%
  summarise(start = min(Date),
            end = max(Date))
```

```
FALSE # A tibble: 1 x 2
FALSE   start      end
FALSE   <date>     <date>
FALSE 1 2013-07-01 2014-06-30
```

What's the difference between "Attendance" and "Admissions"?

Attendance is the number of patient visits; Admissions are those visits that result in hospitalisation.

What do the variables `Tri_1`, `Tri_2`, ... represent?

Attendance under different triage categories.

Task 2: Tidy data (5 points)

Task 2.1 Cleaning up columns

You may notice that the ED csv file has two rows of heading. This is quite common in data generated by BI reporting tools. Let's clean up the column names.

```
ed_data_link <- 'govhack3.csv'
top_row <- read_csv(ed_data_link, col_names = FALSE, n_max = 1)
second_row <- read_csv(ed_data_link, n_max = 1)

column_names <- second_row %>%
  unlist(., use.names=FALSE) %>%
  make.unique(., sep = "__") # double underscore

column_names[2:8] <- str_c(column_names[2:8], '0', sep='__')

daily_attendance <-
  read_csv(ed_data_link, skip = 2, col_names = column_names)
```

Now print out a list of healthcare facilities (hospitals) in the data set.

```
(
facilities <- top_row %>%
  unlist(., use.names=FALSE) %>%
  na.omit()
)
```

```
FALSE [1] "Royal Perth Hospital"
FALSE [2] "Fremantle Hospital"
FALSE [3] "Princess Margaret Hospital For Children"
FALSE [4] "King Edward Memorial Hospital For Women"
FALSE [5] "Sir Charles Gairdner Hospital"
FALSE [6] "Armadale/Kelmscott District Memorial Hospital"
FALSE [7] "Swan District Hospital"
FALSE [8] "Rockingham General Hospital"
FALSE [9] "Joondalup Health Campus"
FALSE attr(,"na.action")
FALSE  [1]  1  3  4  5  6  7  8 10 11 12 13 14 15 17 18 19 20 21 22 24 25 26 27
FALSE [24] 28 29 31 32 33 34 35 36 38 39 40 41 42 43 45 46 47 48 49 50 52 53 54
FALSE [47] 55 56 57 59 60 61 62 63 64
FALSE attr(,"class")
FALSE [1] "omit"
```

Task 2.2 Tidying data

1. Now we have a data frame. Answer the following questions for this data frame.

- Does each variable have its own column?
- Does each observation have its own row?
- Does each value have its own cell?

Duplicated Columns for most variables; Or multiple observations sharing a row.

2. Use spreading and/or gathering to transform the data frame into tidy data. The key is to put data from the same measurement source in a column and to put each observation in a row. Please answer the following questions.

- How many spreading operations do you need?
- How many gathering operations do you need?
- Explain the steps.

```
daily_attendance <- daily_attendance %>%
  gather(key = index,
         value = value,
         -Date)

daily_attendance <- daily_attendance %>%
  separate(index,
           into = c("index",
                    "facility_id"),
           sep="__",
           remove=TRUE) %>%
  mutate(facility_id =
           as.numeric(facility_id) + 1) %>%
  mutate(facility_name =
           facilities[facility_id]) %>%
  select(Date,
         facility_name,
         index,
         value)

## Replacing N/A with 0
daily_attendance <- daily_attendance %>%
  mutate(value =
           as.numeric(value)) %>%
  replace_na(replace =
               list(value = 0))


daily_attendance <- daily_attendance %>%
  spread(index,
         value)

write_rds(daily_attendance, "perth_ed_data.rds")
```

gather -> separate -> spread

3. Are the variables having the expected variable types in R? Clean up the data types.

   Among others, datetime variables need to be processed.

```
daily_attendance <- daily_attendance %>%
  mutate(Date = dmy(Date))
```

4. Are there any missing values? Fix the missing data. Justify your actions.

   Cleaned up in the step before.

Task 3: Exploratory Data Analysis (5 points)

It is often a good idea to eyeball your data before fitting a model. The purpose is to understand the distribution of different measurements and their relations.

Task 3.1 Select a hospital

Select a hospital and create a data set for only that hospital. Print out the hospital's name, the total number of ED attendances and the total number of admissions.

```
selected_hospital <- 'Swan District Hospital'

one_hospital <- daily_attendance %>%
  filter(facility_name == selected_hospital) %>%
  select(-facility_name) %>%
  arrange(Date)

one_hospital %>%
  summarise(total_admission = sum(Admissions),
            total_attendance = sum(Attendance))
```

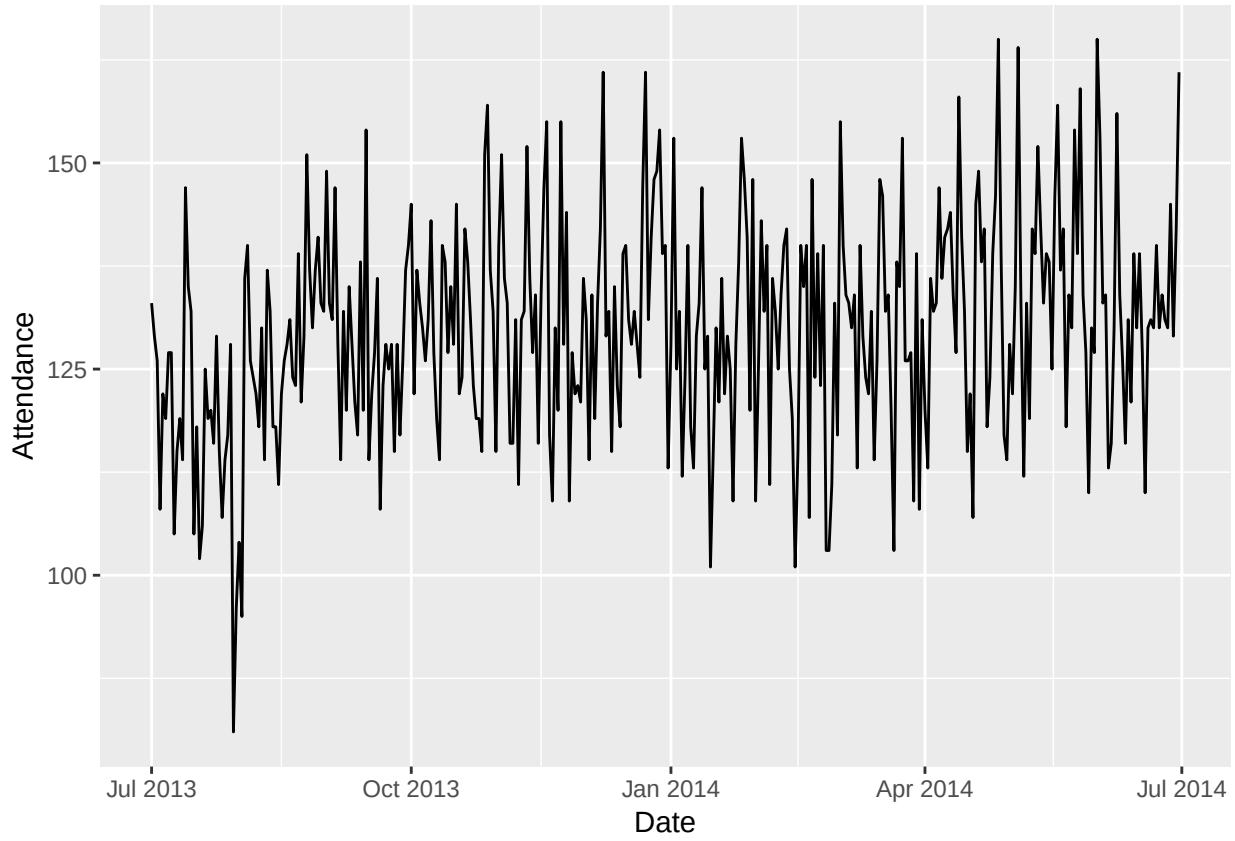
```
FALSE # A tibble: 1 x 2
FALSE   total_admission total_attendance
FALSE             <dbl>            <dbl>
FALSE 1            8993            47347
```

Task 3.2 For the hospital selected, if we want to compare the volume of ED demands across the year, which plot can we use? Show your plot and explain what the plot shows. (Hint: Which variables measure the ED demands?)

Line plot.

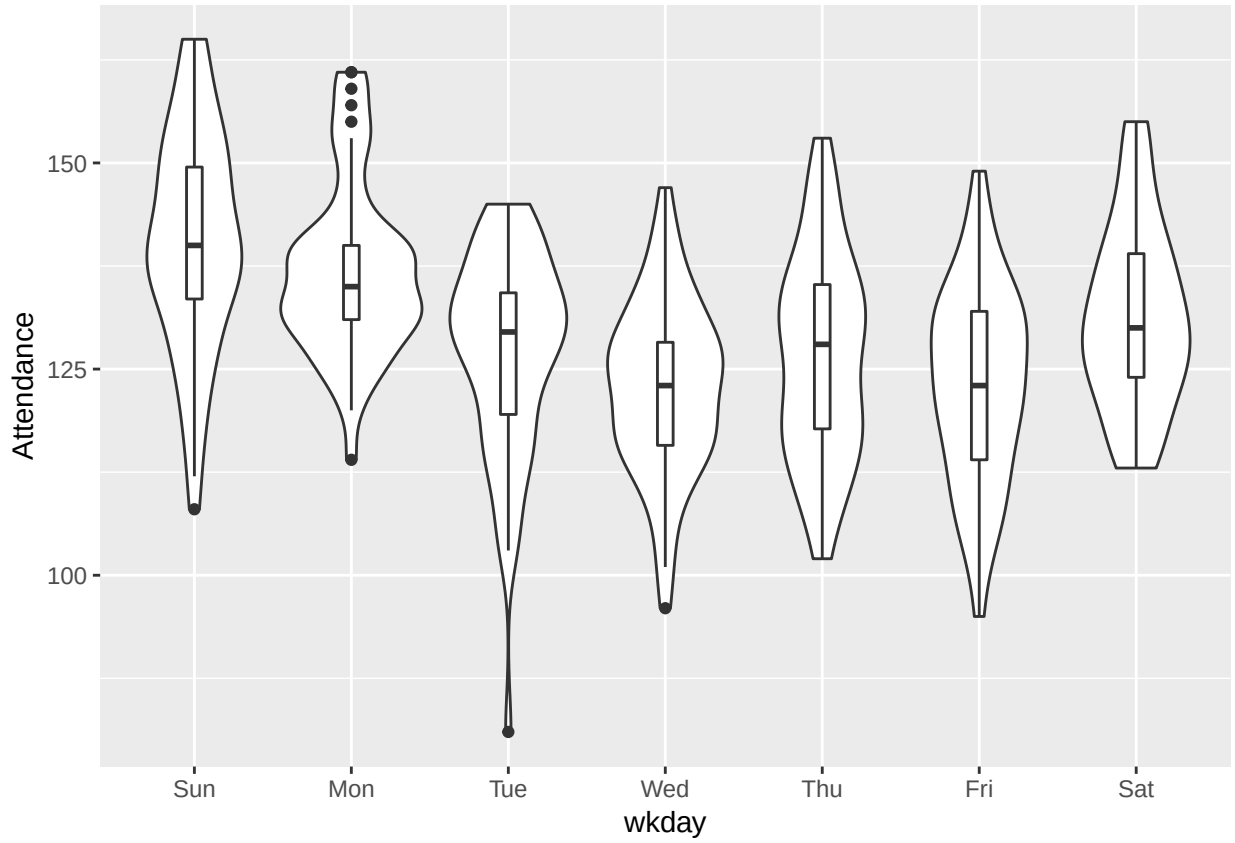
Showing the weekly pattern and change within a year.

```
one_hospital %>%
  ggplot() +
  geom_line(aes(x = Date,
                y = Attendance))
```

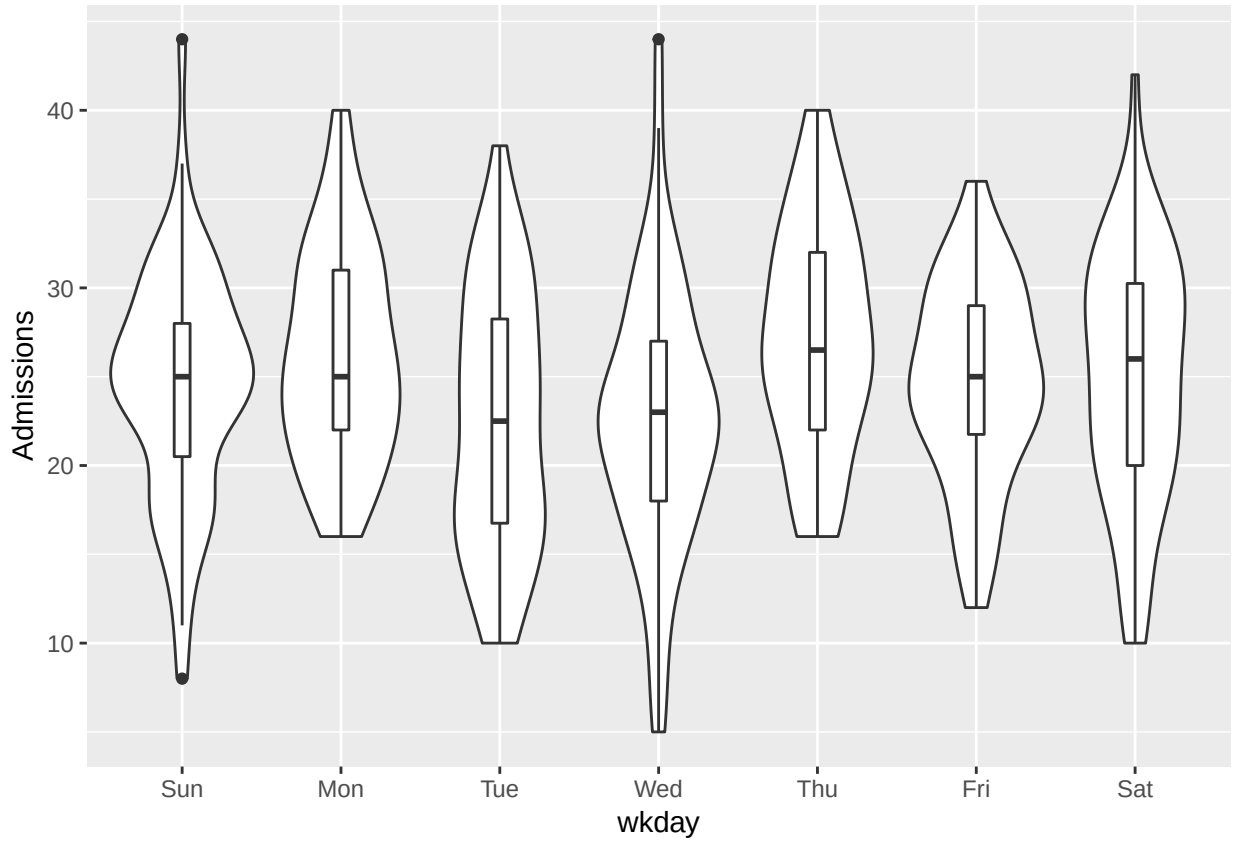


Task 3.3 How do the ED demands change during a week? Show it visually.

```
one_hospital <- one_hospital %>%
  mutate(wkday = wday(Date,
                      label = TRUE))

one_hospital %>%
  ggplot(aes(x = wkday,
             y = Attendance)) +
  geom_violin() +
  geom_boxplot(width=.1)
```




```
one_hospital %>%
  ggplot(aes(x = wkday,
             y = Admissions)) +
  geom_violin() +
  geom_boxplot(width=.1)
```




```
# Or similarly for other counts
```

Task 3.4 Which distributions are appropriate for modelling the ED demand? Which variables meet the assumptions for the Poisson distribution? (For simplicity, here we will make a "naive" assumption that counts on consecutive days are independent. We will relax this assumption later in the unit.)

```
one_hospital %>%
  summarise_each(c('Attendance',
                   str_c('Tri', 1:5, sep='_')),
                 funs = c('mean', 'var')) # Showing mean and variance are similar only for Tri_2
```

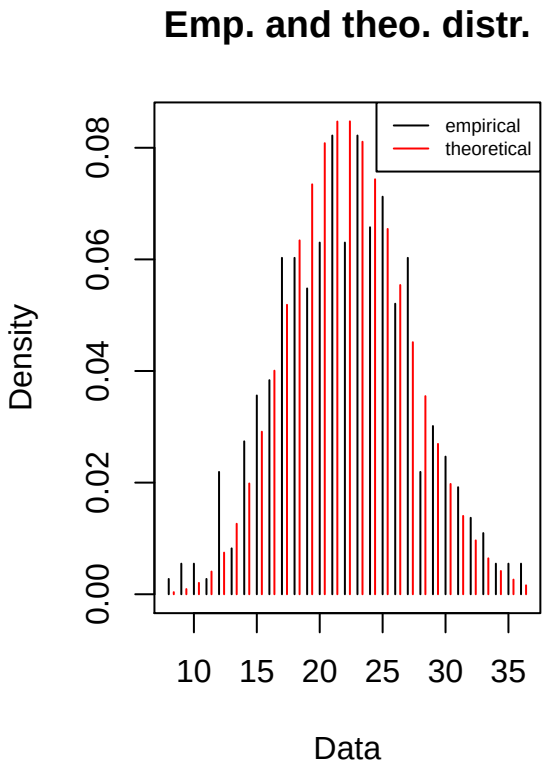
```
FALSE # A tibble: 1 x 12
FALSE   Attendance_mean Tri_1_mean Tri_2_mean Tri_3_mean Tri_4_mean Tri_5_mean
FALSE             <dbl>      <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
FALSE 1            130.      0.299       22.0       49.7       49.3       7.43
FALSE # ... with 6 more variables: Attendance_var <dbl>, Tri_1_var <dbl>,
FALSE #   Tri_2_var <dbl>, Tri_3_var <dbl>, Tri_4_var <dbl>, Tri_5_var <dbl>
```

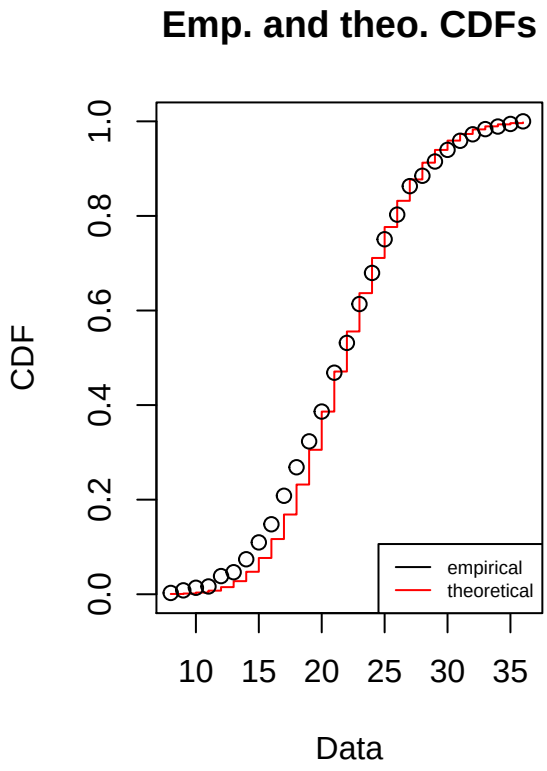
```
library(skimr)
skim(one_hospital)
```

```
FALSE Skim summary statistics
```

```
FALSE  n obs: 365
FALSE  n variables: 9
FALSE
FALSE -- Variable type:Date -------------------------------------------------------------------------------
FALSE  variable missing complete   n        min        max     median n_unique
FALSE      Date       0      365 365 2013-07-01 2014-06-30 2013-12-30      365
FALSE
FALSE -- Variable type:factor -----------------------------------------------------------------------------
FALSE  variable missing complete   n n_unique                          top_counts
FALSE     wkday       0      365 365        7 Mon: 53, Sun: 52, Tue: 52, Wed: 52
FALSE  ordered
FALSE     TRUE
FALSE
FALSE -- Variable type:numeric ----------------------------------------------------------------------------
FALSE    variable missing complete   n   mean    sd p0 p25 p50 p75 p100     hist
FALSE  Admissions       0      365 365  24.64  6.77  5  20  25  29   44 <U+2581><U+2582><U+2586><U+2587
FALSE  Attendance       0      365 365 129.72 13.29 81 121 130 138  165 <U+2581><U+2581><U+2582><U+2586
FALSE       Tri_1       0      365 365   0.3   0.94  0   0   0   0    4 <U+2587><U+2581><U+2581><U+2581
FALSE       Tri_2       0      365 365  22.01  5.28  8  18  22  25   36 <U+2581><U+2583><U+2585><U+2587
FALSE       Tri_3       0      365 365  49.74  8.79 26  44  50  56   80 <U+2581><U+2583><U+2586><U+2587
FALSE       Tri_4       0      365 365  49.35  9.69 26  43  49  55   82 <U+2581><U+2583><U+2587><U+2587
FALSE       Tri_5       0      365 365   7.43  4.5   0   5   7  10   28 <U+2583><U+2587><U+2585><U+2582
```

```r
library(magrittr)
library(fitdistrplus)

estimate <- one_hospital %$%
  fitdist(data = Tri_2,
          distr = "pois")

estimate %>%
  plot
```

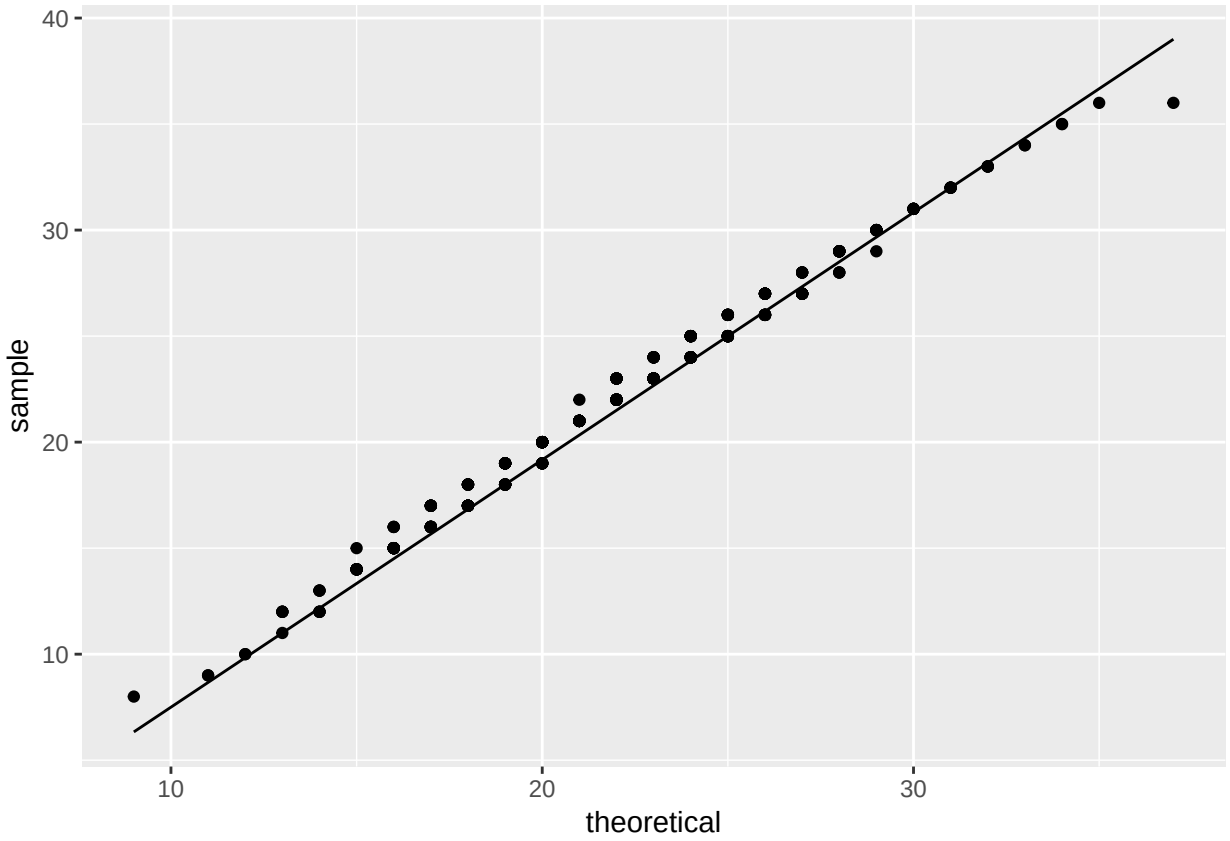





```
estimate %>%
  summary
```

```
FALSE Fitting of the distribution ' pois ' by maximum likelihood
FALSE Parameters :
FALSE        estimate Std. Error
FALSE lambda 22.00822  0.2455534
FALSE Loglikelihood:  -1131.469   AIC:  2264.938   BIC:  2268.838
```

```
one_hospital %>%
  ggplot(aes(sample = Tri_2)) +
  stat_qq(distribution = stats::qpois, dparams = estimate$estimate) +
  stat_qq_line(distribution = stats::qpois, dparams = estimate$estimate)
```



Only Tri_2 has the mean similar to the variance.

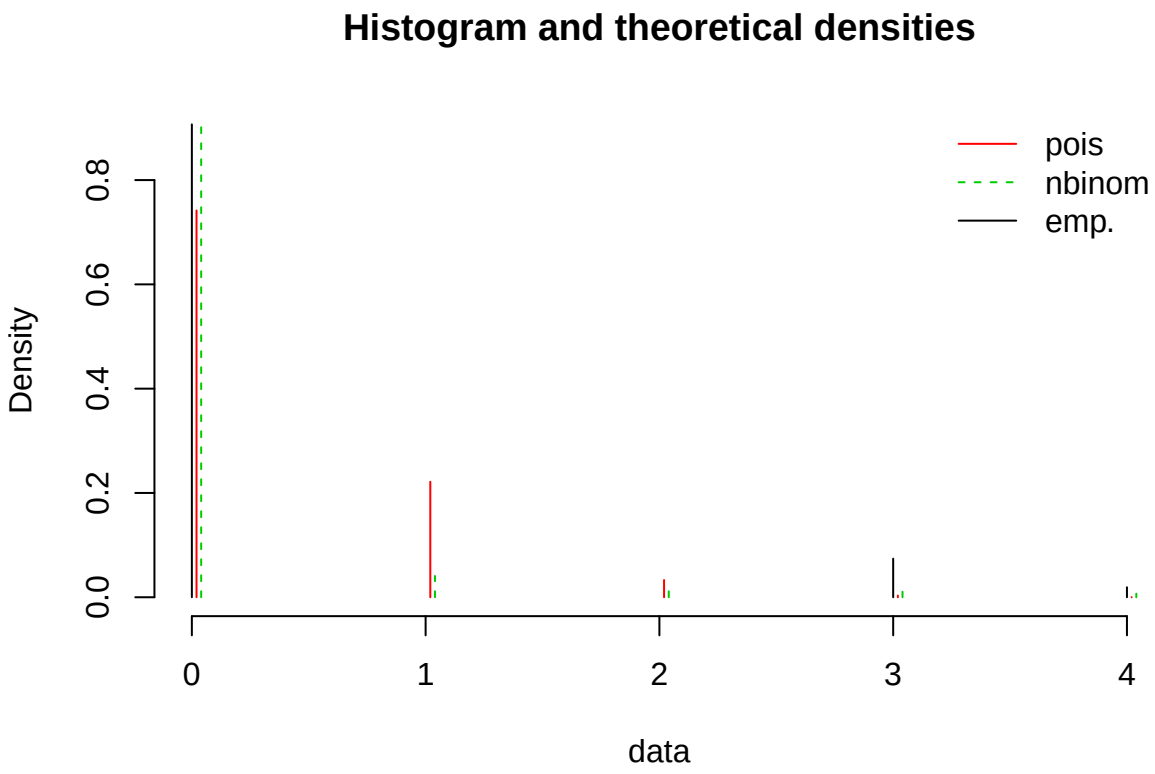
Task 4: Fitting distributions (5 points)

As you may see in the previous step, although we are dealing with count data, a Poisson distribution may not provide a good fit. Actually, unconditional Poisson distribution is too restrictive for most real-world applications. In this task, we will fit a couple of distributions to the Triage 1 attendance.

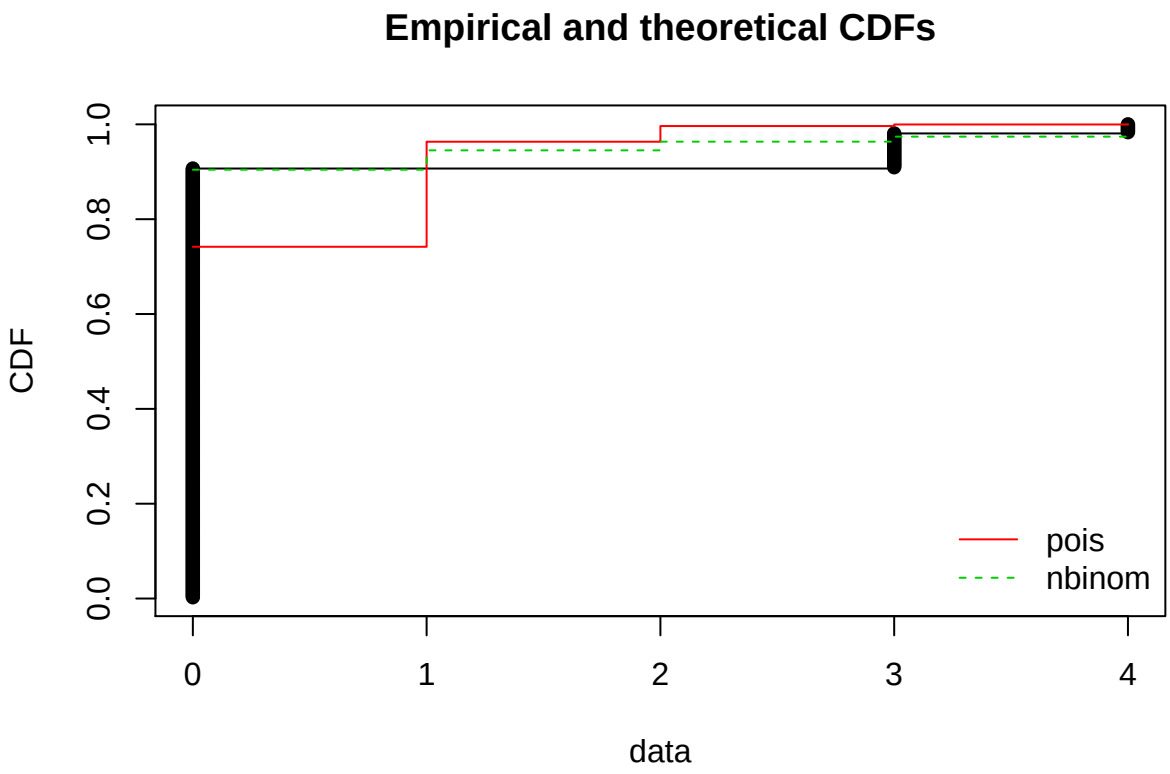
Task 4.1: Fitting distributions

Fit a Poisson distribution and a negative binomial distribution on `Tri_1`. You may use functions provided by the package `fitdistrplus`.

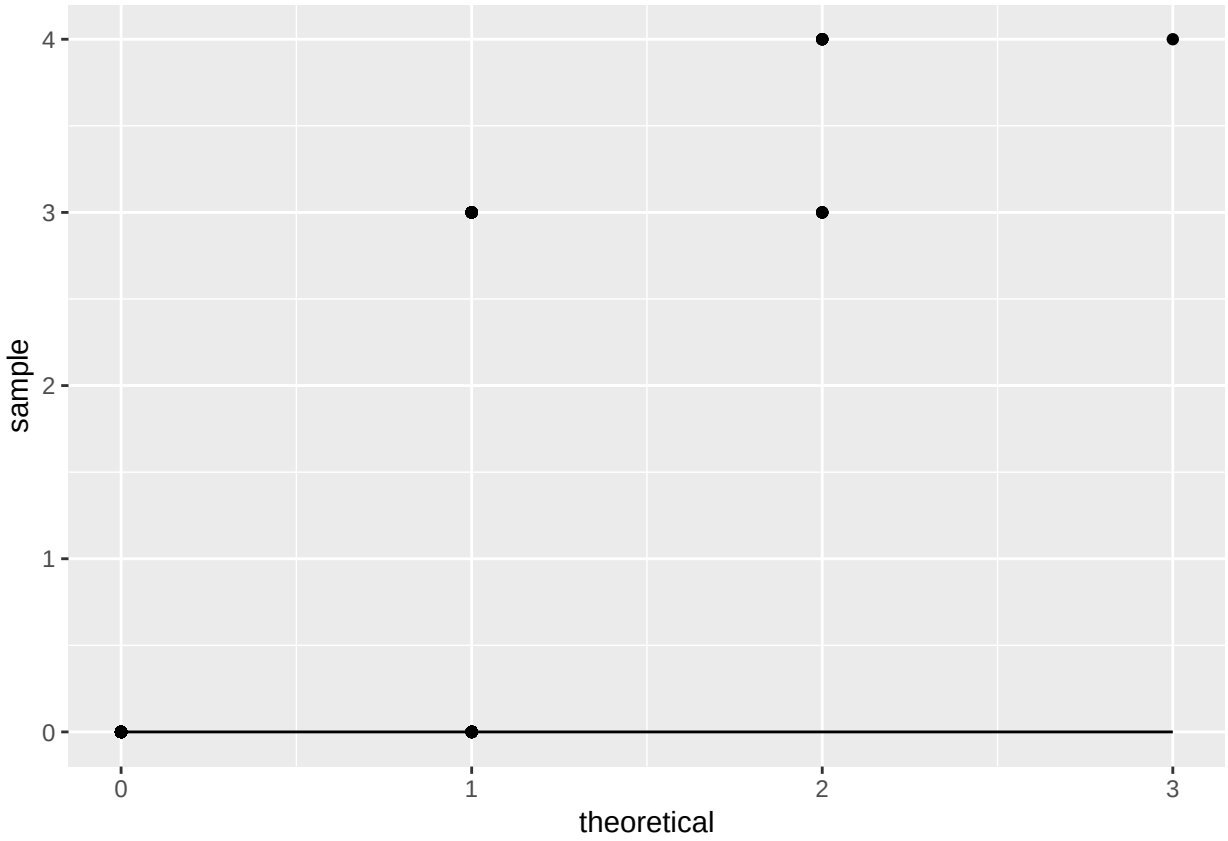
```
fit_P <- fitdist(one_hospital$Tri_1, "pois")

fit_NB <- fitdist(one_hospital$Tri_1, "nbinom")

denscomp(list(fit_P, fit_NB))
```



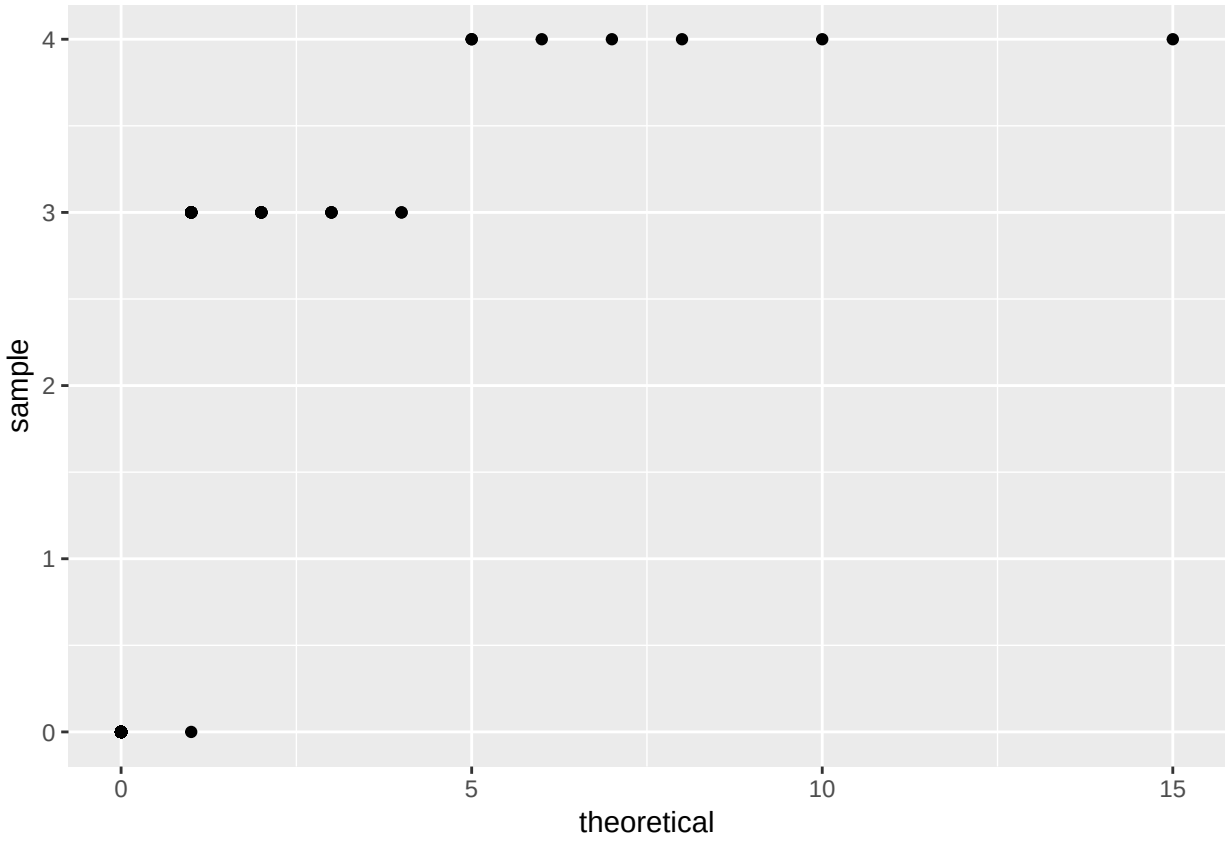

```
cdfcomp(list(fit_P, fit_NB))
```




```
one_hospital %>%
  ggplot(aes(sample = Tri_1)) +
  stat_qq(distribution = stats::qpois, dparams = fit_P$estimate) +
  stat_qq_line(distribution = stats::qpois, dparams = fit_P$estimate)
```




```
one_hospital %>%
  ggplot(aes(sample = Tri_1)) +
  stat_qq(distribution = stats::qnbinom, dparams = fit_NB$estimate) +
  stat_qq_line(distribution = stats::qnbinom, dparams = fit_NB$estimate)
```



Task 4.2: Compare distributions

Compare the log-likelihood of two fitted distributions.

```
fit_P$loglik
```

```
FALSE [1] -311.3558
```

```
fit_NB$loglik
```

```
FALSE [1] -190.8864
```

Which distribution fit the data better? Why?

NB has a greater log-likelihood, and better fit of the data.

Task 5: Research question (3 points)

There are more than one ways to fit a distribution to a set of numbers. Produce a short literature review on different distribution fitting methods, showing the pros and cons of each method.

moment-based Generalised Lambda distribution ...

Task 6: Ethics question (2 points)

During your work, have you identified any issues that have ethical implications? Does it concern security or privacy? How do you mitigate the risk?

> Low count in Tri_1 attendance present potential privacy concerns. Risk has been mitigated though censoring low counts to 0.

Task 7: Reflection (1 point)

Answer the following questions:

1. What help did you receive from other students? What did you learn from them?
2. Please estimate the mark that you will receive for assignment 1. Please provide both a point estimate and an interval estimate (a confidence interval). You don't need to provide a mathematical model, but please explain how do you use conditional information to reach the estimates. Based on the conditional information, explain what you would have done differently to improve that mark?

What to submit

By the due date, you are required to submit the following files to the assignment Dropbox in CloudDeakin.

1. An MS Word or PDF file containing your answers to all the assignment questions.
2. An R Notebook file `Assignment1_submission.Rmd` filled in with the script for your calculations. The file should be able to run. Include sufficient comments so that the script can be understood by the marker. Indicate all the packages that need to be installed separately.

Marking criteria

Your submission will be marked using the following criteria.

- Showing good effort through completed tasks.
- Applying statistical thinking to understand the problems and to identify solutions.
- Applying statistical programming skills to obtain data and to process them for data analysis.
- Applying visualisation techniques to discover distribution patterns and relationships among variables.
- Demonstrating creativity and resourcefulness in solutions.
- Showing attention to details through a good quality assignment report.