Lab 01A: Week 2 (Solutions)

Contents

The **specific aims** of this lab are:

- build experience in R, RStudio and generating R Markdown documents
- improve your statistical literacy
- generate discussion and provide an opportunity to practice *statistical thinking* and *communicating statistical concepts*
- practice chi-squared tests for categorical data

The unit **learning outcomes** addressed are:

- LO1 Formulate domain/context specific questions and identify appropriate statistical analysis.
- LO2 Extract and combine data from multiple data resources.
- LO3 Construct, interpret and compare numerical and graphical summaries of different data types including large and/or complex data sets.
- LO8 Create a reproducible report to communicate outcomes using a programming language.

1 Quick quiz

1 Quick quiz

See the Sway section of Ed.

Solution:

Source: Ben Jones

The average NFL player is about 25 years old, just over 6'2" in height, weighs a little more than 110kg and makes slightly less than US$1.5M in salary per year. Can you tell which distribution goes with which trait? See why the shape of the distribution matters a lot?

A. Height

B. Age

C. Weight

D. Salary

2 Group exercise

As a group (in breakout rooms on Zoom) discuss and brainstorm the following:

1. Select one of the data sets from below:

- what questions could you ask of the data?
- draft some visualisations you could use to answer these questions (you don't have to actually create them, just think about what you could possibly create)
- are there any properties of the data which might confound the question or make it difficult to answer?
- which of the questions you brainstormed were the hardest to answer visually?

2. Present the outcomes of your discussion to the rest of the class (your tutor might share a GoogleDoc for you to add your comments to).

2.1 Australian road fatalities

Australian road fatalities since 1989. Data sourced from the Australian Roads and Deaths Database (Bureau of Infrastructure Transport and Regional Economics 2018). You can download the data to June 2021 from here.

```
library("tidyverse") # loads readr, dplyr, ggplot2, ...
# If you download the data file first you can read it in using the
# readxl package if it's in your current working directory:
```

```
# readxl package if it's in your current working directory:
fdata = readxl::read_excel("ardd_fatalities_jun2021.xlsx", sheet = 2, skip = 4, na =
        c("","-9"), guess_max = 1e6)
```

```
glimpse(fdata)
```

```
Rows: 52,572
Columns: 23
$ `Crash ID`                       <dbl> 20214003, 20215045, 20215003…
$ State                            <chr> "SA", "WA", "WA", "NSW", "Ql…
$ Month                            <dbl> 6, 6, 6, 6, 6, 6, 6, 6, 6, 6…
$ Year                             <dbl> 2021, 2021, 2021, 2021, 2021…
$ Dayweek                          <chr> "Sunday", "Tuesday", "Sunday…
$ Time                             <dttm> 1899-12-31 00:00:00, 1899-1…
$ `Crash Type`                     <chr> "Single", "Single", "Single"…
$ `Bus Involvement`                <chr> "No", "No", "No", "No", "No"…
$ `Heavy Rigid Truck Involvement`  <chr> "No", "No", "No", "No", "No"…
$ `Articulated Truck Involvement`  <chr> "No", "No", "No", "No", "No"…
$ `Speed Limit`                    <chr> "110", "110", "110", "100", …
$ `Road User`                      <chr> "Passenger", "Driver", "Driv…
$ Gender                           <chr> "Male", "Male", "Male", "Mal…
$ Age                              <dbl> 24, 32, 27, 23, 21, 19, 19, …
$ `National Remoteness Areas`      <chr> "Remote Australia", NA, NA, …
$ `SA4 Name 2016`                  <chr> "South Australia - South Eas…
$ `National LGA Name 2017`         <chr> "Kangaroo Island (DC)", NA, …
$ `National Road Type`             <chr> "Sub-arterial Road", NA, NA,…
$ `Christmas Period`               <chr> "No", "No", "No", "No", "No"…
$ `Easter Period`                  <chr> "No", "No", "No", "No", "No"…
$ `Age Group`                      <chr> "17_to_25", "26_to_39", "26_…
$ `Day of week`                    <chr> "Weekend", "Weekday", "Weeke…
$ `Time of day`                    <chr> "Night", "Night", "Night", "…
```

Open the above Excel file and try to work out what each of the parameters are doing, e.g. `sheet = 2`, `skip = 4`, `na = c("","-9")` and `guess_max = 1e6`. The help for the `read_excel()` may also be usefuk `?read_excel`. Also the data dictionary linked from the Australian Roads and Deaths Database homepage. Check what happens if you don't include these parameters, do you get warning messages?

2.2 Cereal

This data is taken from the The Data and Story Library. Missing values are identified as those with a value of `-1`.

```
path = "https://github.com/DATA2002/data/raw/master/Cereal.csv"
cereal = readr::read_csv(path, na = "-1")
glimpse(cereal)
```

```
Rows: 77
Columns: 16
```

```
columns: 16
$ name     <chr> "100%_Bran", "100%_Natural_Bran", "All-Bran", "All-…
$ mfr      <chr> "N", "Q", "K", "K", "R", "G", "K", "G", "R", "P", "…
$ type     <chr> "C", "C", "C", "C", "C", "C", "C", "C", "C", "C", "…
$ calories <dbl> 70, 120, 70, 50, 110, 110, 110, 130, 90, 90, 120, 1…
$ protein  <dbl> 4, 3, 4, 4, 2, 2, 2, 3, 2, 3, 1, 6, 1, 3, 1, 2, 2, …
$ fat      <dbl> 1, 5, 1, 0, 2, 2, 0, 2, 1, 0, 2, 2, 3, 2, 1, 0, 0, …
$ sodium   <dbl> 130, 15, 260, 140, 200, 180, 125, 210, 200, 210, 22…
$ fiber    <dbl> 10.0, 2.0, 9.0, 14.0, 1.0, 1.5, 1.0, 2.0, 4.0, 5.0,…
$ carbo    <dbl> 5.0, 8.0, 7.0, 8.0, 14.0, 10.5, 11.0, 18.0, 15.0, 1…
$ sugars   <dbl> 6, 8, 5, 0, 8, 10, 14, 8, 6, 5, 12, 1, 9, 7, 13, 3,…
$ potass   <dbl> 280, 135, 320, 330, NA, 70, 30, 100, 125, 190, 35, …
$ vitamins <dbl> 25, 0, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, …
$ shelf    <dbl> 3, 3, 3, 3, 3, 1, 2, 3, 1, 3, 2, 1, 2, 3, 2, 1, 1, …
$ weight   <dbl> 1.00, 1.00, 1.00, 1.00, 1.00, 1.00, 1.00, 1.33, 1.0…
$ cups     <dbl> 0.33, 1.00, 0.33, 0.50, 0.75, 0.75, 1.00, 0.75, 0.6…
$ rating   <dbl> 68.40297, 33.98368, 59.42551, 93.70491, 34.38484, 2…
```

3 Projects in RStudio

One of the hallmarks of good science is reproducibility. R Markdown documents help this goal, but it's not so helpful if you can't find the file you just created, or the data set it refers to. It's vital to have an appropriate folder structure on your computer to keep your various analyses in. The structure below is a suggestion:

```
DATA2002/
  |- Labs/
    |- Lab00/
      |- Lab00.rmd
      |- Lab00.html
    |- Lab1a/
      |- Lab1a.rproj
      |- Lab1a.rmd
      |- Lab1a.html
      |- data/
        |- FILE_NAME.csv
  |- Assignment/
    |- Assignment.rproj
    |- Assignment.rmd
    |- Assignment.html
```

Some key elements:

- There are `.rproj` files in some folders, these store information about "RStudio projects." `File > New Project` will let you create a new project. I recommend creating a new project for each lab and for each module report.
- Once you're in a project, the working directory is wherever that `.rproj` file is stored and you can refer to files relative to that working directory.

- You can switch between projects using the top right drop down menu in the RStudio interface. You can also have multiple projects open at the same time - for example you could have the `Lab1a.rproj` project open in one RStudio window and the `Assignment.rproj` open in another window. When you do this the two instances of RStudio don't know about each other, i.e. an object available in one RStudio window is not accessible in the other RStudio window.

You can find out more about R projects here.

4 Exercises

4.1 Tablet devices

Tablet devices are an increasingly important component of the global electronics market. According to a market intelligence research company, the use of tablet devices can be classified into the following user segments.

User Segment	2012 percentages	Current survey frequency
Business-Professional	69%	102
Goverment	21%	32
Education	7%	12
Home	3%	4
Total	100%	150

Do the data provide sufficient evidence to indicate that the figures obtained in the current survey agree with the percentages in 2012?

Some R code to help with the calculations:

```
y_i = c(102, 32, 12, 4)
p_i = c(0.69, 0.21, 0.07, 0.03)
n = sum(y_i)
e_i = n * p_i
```

Solution:

The calculation is summarised in the following table:

User Segment	Obs. freq. y_i	Exp. prob. p_i	Exp. freq. $e_i = np_i$	$\frac{(y_i - e_i)^2}{e_i}$
Business-Prof.	102	0.69	$150 \times 0.69 = 103.5$	
Goverment	32	0.21	$150 \times 0.21 = 31.5$	
Education	12	0.07	$150 \times 0.07 = 10.5$	

Home	4	0.03	$150 \times 0.03 = 4.5$
Total	150	1	150

Since $e_4 = 4.5 < 5$, we combine the last two classes. The revised table is

User Segment	Obs. freq. y_i	Exp. prob. p_i	Exp. freq. $e_i = np_i$	$\frac{(y_i - e_i)^2}{e_i}$
Business-Prof.	102	0.69	$150 \times 0.69 = 103.5$	$\frac{(102-103.5)^2}{103.5} = 0.0217$
Goverment	32	0.21	$150 \times 0.21 = 31.5$	$\frac{(32-31.5)^2}{31.5} = 0.0079$
Education and home	16	0.10	$150 \times 0.10 = 15$	$\frac{(16-15)^2}{15} = 0.0667$
Total	150	1	150	0.0963

```
y_i = c(102, 32, 12 + 4)
p_i = c(0.69, 0.21, 0.07 + 0.03)
n = sum(y_i)
e_i = n * p_i
t0 = sum((y_i - e_i)^2/e_i)
t0
```

```
[1] 0.09634231
```

```
pval = pchisq(t0, 2, lower.tail = FALSE)
pval
```

```
[1] 0.9529707
```

```
chisq.test(y_i, p = p_i)
```

```

    Chi-squared test for given probabilities

data:  y_i
X-squared = 0.096342, df = 2, p-value = 0.953
```

The chi-squared goodness-of-fit test is

1. **Hypotheses:** $H_0 : p_1 = 0.69,\ p_2 = 0.21,\ p_3 = 0.1$ vs H_1 : at least one of the equalities does not hold.
2. **Assumption:** $e_i = np_i \geq 5$.
3. **Test statistic:** $T = \sum_{i=1}^{3} \frac{(Y_i - e_i)^2}{e_i}$ Under H_0, $T \sim \chi^2_2$ approx.
4. **Observed test statistic:** $t_0 = 0.0963$
5. **p-value:** $P(\chi^2_2 \geq 0.0963) = 0.953$
6. **Decision:** Since the p-value is (much) greater than 0.05, we no not reject the null hypothesis

5. **Decision:** Since the p-value is (much) greater than 0.05, we do not reject the null hypothesis. Hence, the current data is consistent with the distribution of tablet devices in 2012.

4.2 Smoking rates

A study of patients with insulin-dependent diabetes was conducted to investigate the effects of cigarette smoking on renal and retinal complications. Before examining the results of the study, a researcher expects that the proportions of four different subgroups are as follow:

Subgroup	Proportion
Nonsmokers	0.50
Current Smokers	0.20
Tobacco Chewers	0.10
Ex-smokers	0.20

Of 100 randomly selected patients, there are 44 nonsmokers, 24 current smokers, 13 tobacco chewers and 19 ex-smokers. Should the researcher revise his estimates? Use 0.01 as the level of significance.

```
y_i = c(  ,  ,  ,  )
p_i = c(0.5, 0.2, 0.1, 0.2)
n = sum(  )
e_i = n * p_i
```

Solution:

The calculation is summarised in the following table:

Selection i	Observed freq. y_i	Expected freq. $e_i = np_i$	Diff. $(y_i - e_i)$	Test stat. $\frac{(y_i-e_i)^2}{e_i}$
Nonsmokers	44	$100 \times 0.5 = 50$	-6	$\frac{(-6)^2}{50} = 0.72$
Current Smokers	24	$100 \times 0.2 = 20$	4	$\frac{4^2}{20} = 0.80$
Tobacco Chewers	13	$100 \times 0.1 = 10$	3	$\frac{3^2}{10} = 0.90$
Ex-smokers	19	$100 \times 0.2 = 20$	-1	$\frac{(-1)^2}{20} = 0.05$
Total	100	100	0	$t_0 = 2.47$

```
y_i = c(44, 24, 13, 19)
p_i = c(0.5, 0.2, 0.1, 0.2)
n = sum(y_i)
```

```
e_i = n * p_i
t0 = sum((y_i - e_i)^2/e_i)
t0
```

```
[1] 2.47
```

```
pval = pchisq(t0, 3, lower.tail = FALSE)
pval
```

```
[1] 0.4807372
```

```
chisq.test(y_i, p = p_i)
```

```
    Chi-squared test for given probabilities

data:  y_i
X-squared = 2.47, df = 3, p-value = 0.4807
```

1. Hypothesis: H_0: $p_1 = .5,\ p_2 = .2,\ p_3 = .1,\ p_4 = .2$ vs H_1: At least one equality do not hold.
2. Assumptions: $e_i = np_i \geq 5$.
3. Test statistic: $T = \sum_{i=1}^{k} \frac{(Y_i - e_i)^2}{e_i}$. Under H_0, $T \sim \chi^2_3$ approx.
4. Observed test statistic: $t_0 = 2.47$
5. p-value: $P(\chi^2_3 \geq 2.47) = 0.481$
6. Decision: Since the p-value is greater than 0.01, we do not reject the null hypothesis. The data is consistent with the proportions estimated by the researcher.

5 Australian road fatalities

Answer the following questions about the Australian road fatalities data.

1. How are missing values recorded, and why might they occur?
2. How many fatalities occurred since 1989? How many fatal crashes have there been since 1989?
3. What is the most common hour of the day for a fatal crash?
4. What is the most common day of the week for a fatal crash?
5. What is the most common month for a fatal crash?
6. Are fatal crashes uniformly distributed across the months of the year? Filter the data down to one year (e.g. 2019) to do this test. You should write out a full hypothesis test and make an appropriate conclusion.

Step 1: Create a new R Project (e.g. call it Lab01). Step 2: download the fatalities to June 2021 Excel

Step 1: Create a new R Project (e.g. call it Lab01). Step 2: download the fatalities to June 2021 Excel file and save it into the R project folder. Step 3: Open a new R Markdown file.

To get things moving, here's some code that imports the data and makes all the required edits at once. Your tutor will help explain the steps.

```
# fatalities data
fdata = readxl::read_excel("ardd_fatalities_jun2021.xlsx",
                           sheet = 2,
                           skip = 4,
                           na = c("","-9"),
                           guess_max = 1e6) %>%
  janitor::clean_names()

# crash data
cdata = fdata %>%
  select(-road_user, -gender, -age, -age_group) %>%
  distinct() %>%
  group_by(crash_id) %>%
  slice(1) %>%
  ungroup() %>%
  mutate(hour = lubridate::hour(time))
```

You might want to investigate the **lubridate** package which provides a range of functions for working with dates and times.

As you're working through this question, spend some time making the output in your compiled HTML file "presentable," i.e. remove any spurious output (messages or warnings) using the chunk options, turn on code folding, include sufficient commentary as text such that you don't need to read the code to know what is going on, if you generate plots, edit the axis labels so that they are meaningful (i.e. not just the raw variable name).

1. The missing numeric values in the data set are denoted by '-9.' This might occur when police can not determine specific details about the fatality, such as the age of the killed person or the speed limit at the crash site. Text entries may contain 'Unknown' if police could not determine those details of the crash. Source: the data dictionary available here.

2. When importing the data, note that we have specified the missing value identifier as `na = "-9"` and asked the `read_excel()` function to inspect more of the data than the default in order to guess the column types. We also cleaned the column names using the `clean_names()` function from the janitor package.

```
fdata = readxl::read_excel("ardd_fatalities_jun2021.xlsx", sheet = 2, skip = 4,
    na = c("", "-9"), guess_max = 1e+06) %>%
    janitor::clean_names()
# glimpse(fdata)
```

Each entry in this data set is a unique fatality, so we can count the number of fatalities by counting the number of rows: 52572. There may be more than one fatality per crash, so to identify the number of

fatal crashes, we need to reduce the data frame removing individual specific data and then applying the `distinct()` function to keep only distinct rows:

```
cdata = fdata %>%
    select(-road_user, -gender, -age, -age_group) %>%
    distinct()
nrow(cdata)
```

```
[1] 47329
```

```
cdata %>%
    select(crash_id) %>%
    n_distinct()
```

```
[1] 47328
```

This is almost the same as checking to see how many distinct `crash_id` entries there are in the data set. There is one inconsistency where we have two rows in the crash data for the same crash ID:

```
cdata %>%
    group_by(crash_id) %>%
    filter(n() > 1)
```

```
# A tibble: 2 × 19
# Groups:   crash_id [1]
  crash_id state month  year dayweek  time                crash_type
     <dbl> <chr> <dbl> <dbl> <chr>    <dttm>              <chr>
1 20164024 SA        3  2016 Thursday 1899-12-31 08:15:00 Single

2 20164024 SA        3  2016 Thursday 1899-12-31 08:15:00 Single
# … with 12 more variables: bus_involvement <chr>,
#   heavy_rigid_truck_involvement <chr>,
#   articulated_truck_involvement <chr>, speed_limit <chr>,
#   national_remoteness_areas <chr>, sa4_name_2016 <chr>,
#   national_lga_name_2017 <chr>, national_road_type <chr>,
#   christmas_period <chr>, easter_period <chr>, day_of_week <chr>,
#   time_of_day <chr>
```

In this crash one fatality had a `speed_limit` of 100 while the other had a `speed_limit` of 60. This could be an error, or perhaps they were traveling on opposite sides of the road and the crash occurred near a change in the speed limit. Let's just keep one of them by slicing the data:

```
cdata = cdata %>%
    group_by(crash_id) %>%
    slice(1) %>%
    ungroup()
nrow(cdata)
```

```
[1] 47328
```

We can say that there were 47328 fatal crashes since 1989.

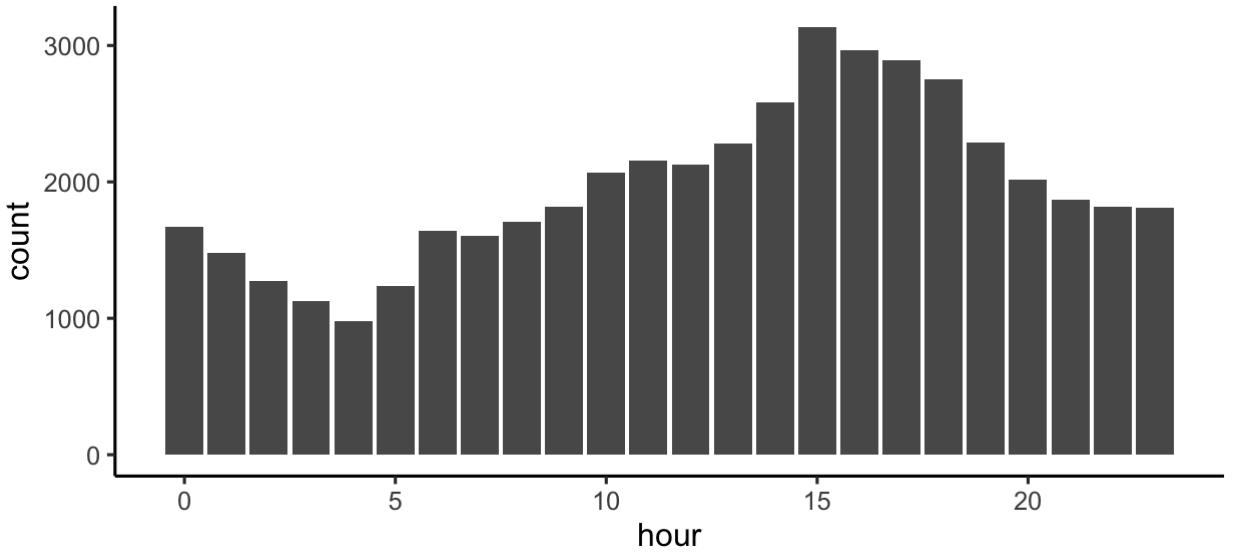
2. To work out the most common hour of the day for a fatal crash let's work with the `time` column

3. To work out the most common hour of the day for a fatal crash let's work with the `time` column. Note that when R imported the time it assigned a default date with it `1899-12-31`, we can ignore the date part of the date-time object and instead use the `hour()` function from the **lubridate** package to extract the hour.

```
cdata = cdata %>%
    mutate(hour = lubridate::hour(time))
```

We can now tabulate or visualise the `hour` column.

```
# cdata %>% count(hour)
cdata %>%
    ggplot() + aes(x = hour) + geom_bar()
```

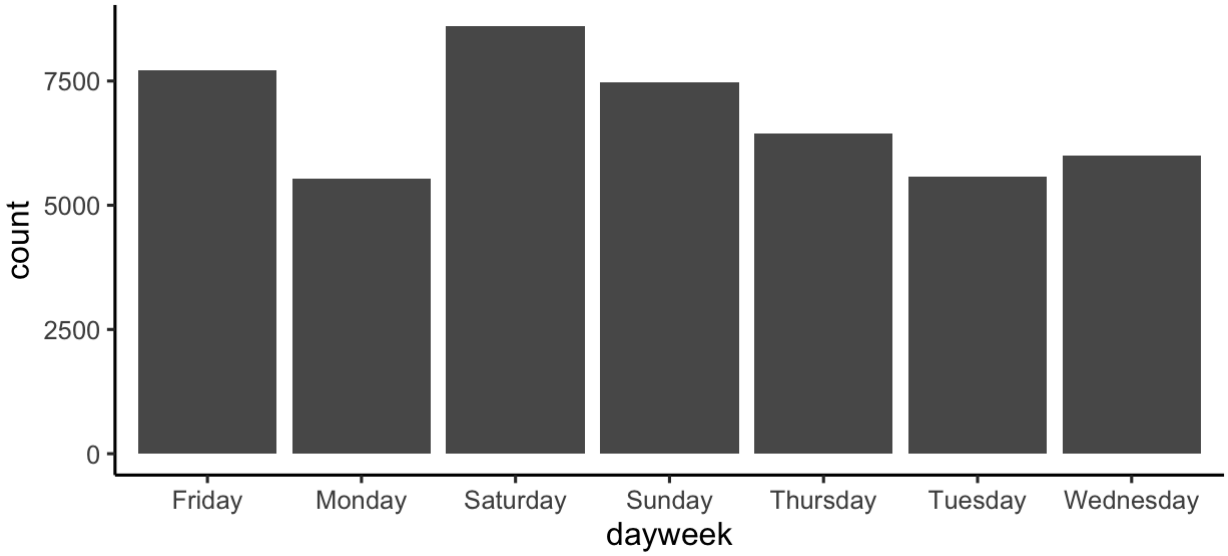


We can see that the most common hour of the day for fatalities is 3pm. Does this necessarily mean that it is most dangerous to drive at 3pm?

4. What is the most common day of the week for a fatal crash?

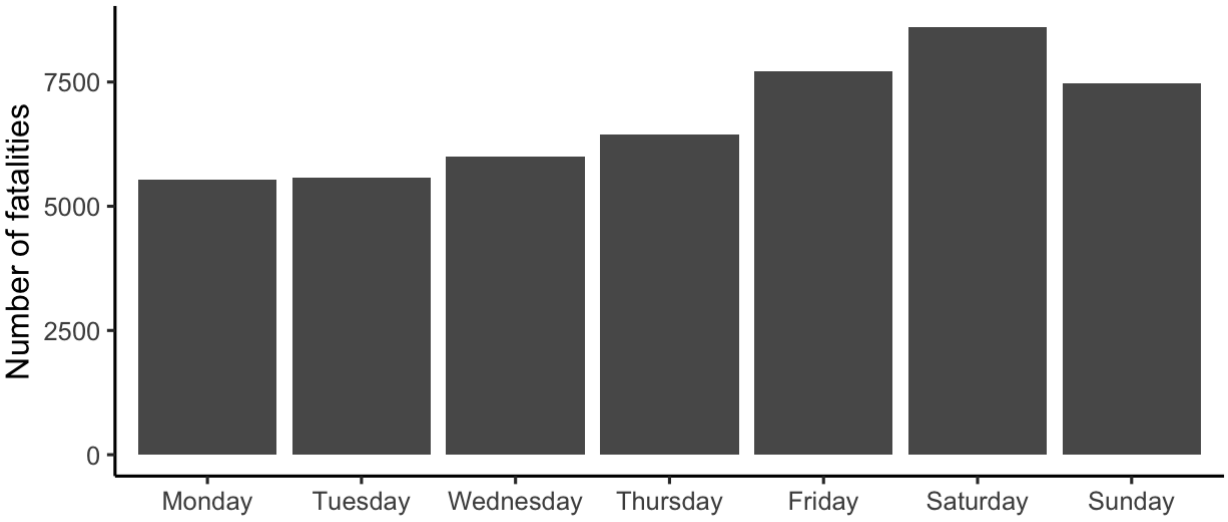
```
# cdata %>% count(dayweek)
cdata %>%
```

```
    ggplot() + aes(x = dayweek) + geom_bar()
```



Saturday is the most common day for fatal crashes. Note that the bar chart isn't great because the days are in alphabetical order. Let's fix that:

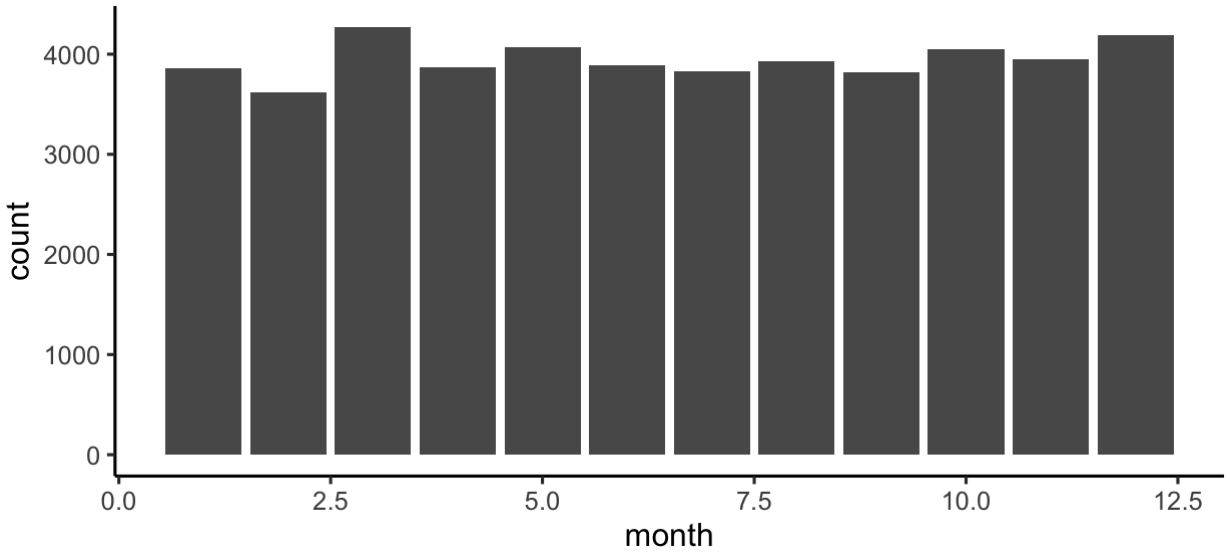
```
cdata = cdata %>%
    mutate(dayweek = factor(dayweek, levels = c("Monday", "Tuesday", "Wednesday",
        "Thursday", "Friday", "Saturday", "Sunday")))
# cdata %>% count(dayweek)
cdata %>%
    ggplot() + aes(x = dayweek) + geom_bar() + labs(y = "Number of fatalities",
    x = "")
```



It is now a lot easier to interpret the plot.

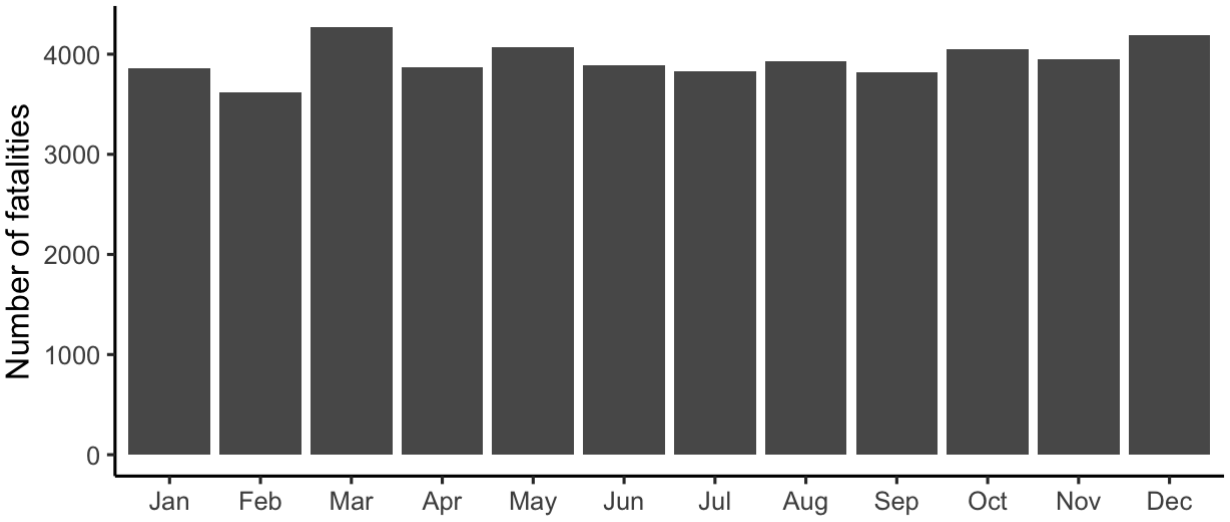
5. What is the most common month for a fatal crash?

```
# cdata %>% count(month)
cdata %>%
    ggplot() + aes(x = month) + geom_bar()
```



Let's create a nicer variable with month names rather than just numbers.

```
cdata = cdata %>%
    mutate(month_named = factor(month, levels = 1:12, labels = month.abb))
cdata %>%
    ggplot() + aes(x = month_named) + geom_bar() + labs(y = "Number of fatalities",
    x = "")
```



6. *Filter down to the year 2019 and perform a test to see if fatal crashes uniformly distributed across the months of the year?* This is a little unfair as not all months have the same number of days, we could do an adjustment, but let's just leave it as is for the purpose of this test.

```
mcount = cdata %>%
    filter(year == 2019) %>%
    dplyr::count(month_named)
```

```
mcount
```

```
# A tibble: 12 × 2
   month_named     n
   <fct>       <int>
 1 Jan           109
 2 Feb            87
 3 Mar           101
 4 Apr            93
 5 May            96
 6 Jun            86
 7 Jul            80
 8 Aug            94
 9 Sep            84
10 Oct            93
11 Nov            79
12 Dec            98
```

To test if fatal crashes are uniformly distributed across the months of the year:

1. $H_0 : p_1 = p_2 = p_3, \ldots, p_{12} = \frac{1}{12}$ vs H_1 : At least one of the equalities does not hold.

2. Assumption: $e_i = np_i \geq 5$

```
mcount = mcount %>%
    mutate(expected = (1/12) * sum(n))
mcount$expected >= 5
```

```
 [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

All expected cell counts are larger than 5. The independence assumption is likely satisfied as car crashes tend to be independent of each other. This may not be completely true, for example if a car crash caused a traffic jam that led to a further crash, but it's at least approximately true.

3. Test statistic: $T = \sum_{i=1}^{12} \frac{(Y_i - e_i)^2}{e_i}$ Under H_0, $T \sim \chi^2_{11}$ approx.

```
(Tstat = sum(((mcount$n - mcount$expected)^2)/mcount$expected))
```

```
[1] 9.432727
```

4. Observed test statistic: $t_0 = 9.43$.

5. p-value: $P(\chi^2_{11} \geq 9.43) = 0.582$.

```
(pv = 1 - pchisq(Tstat, df = 11))
```

```
[1] 0.5820152
```

6. Decision: Since the p-value is quite large (p = 0.582), we do not reject the null hypothesis at the 5% level of significance. We conclude that there is no significant difference between the number of fatal crashes across the months of the year hence the observed pattern of crashes are

consistent with the hypothesis of a uniform distribution across the months of the year.

We could also do it using `chisq.test()`

```
chisq.test(mcount$n)
```

```
    Chi-squared test for given probabilities

data:  mcount$n
X-squared = 9.4327, df = 11, p-value = 0.582
```

Do you get the same result if you look over multiple years? Why do you think increasing the sample size changes the result?

6 For after the lab

6.1 Recap

R functions used:

- `chisq.test()` for performing a chi-square test
- `readr::read_csv()` for importing data from a CSV file
- `dplyr::glimpse()` glimpse a data frame
- `janitor::clean_names()` for cleaning the column names
- `dplyr::select()` select or remove columns from a data frame
- `dplyr::distinct()` keep only unique rows in a data frame
- `dplyr::group_by()` group a data frame by one or more variables and `dplyr::ungroup()` undoes the grouping operation
- `dplyr::slice()` slice rows out of a data frame
- `dplyr::mutate()` create or overwrite columns in a data frame
- `lubridate::hm()` takes a character vector that looks like `"hh:mm"` and converts it into a time vector
- `lubridate::hour()` takes a time vector and extracts just the hour component

6.2 Pollution

To deal with the water pollution problem, three proposals are suggested:

1. Remove the industrial plant
2. Relocate the industrial plant to the river mouth; and
3. Build a sewage plant.

Thirty government officials are interviewed and their opinions are given below. Test, at the 5% level of significance, the null hypothesis that there is no preference among the three proposals. To facilitate the working, you may complete the following table:

Proposal	Observed y_i	Expected e_i	$(y_i - e_i)$	$\frac{(y_i - e_i)^2}{e_i}$
Remove the plant	6			
Relocate the plant to river mouth	9			
Build a sewage plant	15			
Total	30			$t_0 =$

```
y_i = c(6, 9, 15)
p_i = c(  ,  ,  )
n = sum(  )
e_i = n * p_i
```

Solution:

```
y_i = c(6, 9, 15)
p_i = c(1, 1, 1)/3
n = sum(y_i)
e_i = n * p_i
t0 = sum((y_i - e_i)^2/e_i)
t0
```

```
[1] 4.2
```

```
pval = pchisq(t0, 2, lower.tail = FALSE)
pval
```

```
[1] 0.1224564
```

```
chisq.test(y_i, p = p_i)
```

```

    Chi-squared test for given probabilities

data:  y_i
X-squared = 4.2, df = 2, p-value = 0.1225
```

Calculation is summarised in the following table:

Proposal i	Observed freq. y_i	Expected freq. np_i	Diff. $(y_i - e_i)$	$\frac{(y_i-e_i)^2}{e_i}$
1.Remove the plant	6	$30 \times 0.33 = 10$	-4	$\frac{(-4)^2}{10} = 1.6$
2.Relocate the plant	9	$30 \times 0.33 = 10$	-1	$\frac{(-1)^2}{10} = 0.1$
3.Build a sewage plant	15	$30 \times 0.33 = 10$	5	$\frac{5^2}{10} = 2.5$
Total	30	30	0	$t_0 = 4.2$

The chi-squared goodness-of-fit test is

1. Hypothesis: H_0: $p_1 = 0.33,\ p_2 = 0.33,\ p_3 = 0.33$ vs H_1: At least one equality does not hold.
2. Assumptions: $e_i = np_i \geq 5$.
3. Test statistic: $T = \sum_{i=1}^{k} \frac{(Y_i - e_i)^2}{e_i}$. Under H_0, $T \sim \chi^2_2$ approx.
4. Observed test statistic: $t_0 = 4.2$
5. P-value: $P(\chi^2_2 > 4.2) = 0.122$
6. Decision: Since the p-value is greater than 0.05, we do not reject the null hypothesis. The data is consistent with the null hypothesis of the proportions of preferences for the three proposals.

7 Data dictionary

7.1 Cereal

Nutritional information for breakfast cereals ($n = 77$)

- `name` Name of cereal
- `mfr` Manufacturer
  - 1 = American Home Foods
  - 2 = General Mills
  - 3 = Kellogg's
  - 4 = Nabisco
  - 5 = Post

    - 5 = Post
    - 6 = Quaker Oats
    - 7 = Ralston Purina
- `type` Type of cereal
    - 1 = cold
    - 2 = hot
- `calories` Calories per serving
- `protein` Protein grams
- `fat` Fat grams
- `sodium` Sodium millimeters
- `fiber` Fiber
- `carbo` Carbohydrates
- `sugar` Sugar
- `Potass` Potassium
- `vitamin` Vitamins
- `shelf` Shelf position in store
    - 1 = bottom
    - 2 = middle
    - 3 = top
- `weight` Weight (grams)
- `cups` Cups in serving
- `rating` Taste rating

References

Bureau of Infrastructure Transport and Regional Economics. 2018. “Australian Road Deaths Database.” https://bitre.gov.au/statistics/safety/fatal_road_crash_database.aspx.