Midterm Practice Problems

YSC2210 - Data Aanalysis and Visualisation with R

Academic Year 2021/2, Semester 2

1 Exam rules

Important: please read the following rules for the exam.

The exam is open-book and open-Internet with the exception that **any form of real-time communication** (e.g. email, Facebook or user forums) **and file exchanges** (e.g. Dropbox or RStudio Server Pro) **are strictly forbidden**. Do not look at another person's monitor. You must record your screen from the beginning of the exam until after you submitted your solutions to Canvas. You can only use one electronic device during the exam, and the screen of this device must be visible in the screen recording. **It is forbidden to share code with others**—before, during and after the exam—if this code can serve as help for other students during the exam (including students in future editions of this course). If your solution is based on ideas from a book, a web site or code you shared with another student prior to the exam (e.g. your team's homework solutions), you must clearly state the source in your submission.

Upload your R Markdown file to Canvas (under Assignments → Midterm exam) within 105 minutes. Your screen recording should be on Canvas soon afterwards.

Violations of these rules result in appropriate disciplinary procedures.

2 These problems are not representative of the exam's difficulty

I offer these problems for your practice only. They are not intended to be representative of the problems on the exam. However, these problems may aid you in your preparation.

3 R Markdown style

- Load all the packages you need for your solutions at the beginning of your R Markdown file, for example:

```
library(MASS)
library(scales)
library(tidyverse)
```

  Show the code, but do not show package start-up messages in the knitted file.

- Adhere to the tidyverse style (https://style.tidyverse.org/).
- Use code chunk options `fig.width`, `fig.height` and `out.width` to adjust figure dimensions. All parts of the figures should be clearly legible without appearing disproportionately large compared to the font size of the running text in the knitted R Markdown file.

4 Factors

Consider the following code chunk. Briefly explain the output from `table(answers)`. You may want to show the content of `answers` as part of your explanation.

```r
answers <- factor(c(rep("yes", 5), rep("no", 6)))

# On the next line, it is "Yes", not "yes"
answers <- factor(answers, levels = c("Yes", "no"))
table(answers)
```

```
## answers
## Yes  no
##   0   6
```

5 Relation between price, carat and colour of diamonds

(a) Recreate the plot in figure 1. It facets `ggplot2::diamonds` by colour and overlays a line of best fit to the log-transformed full data set on each facet.

- To obtain the same line in each facet (i.e. the best fit to *all* data), instead of a different line in each facet, pass the argument `data = select(diamonds, -color)` to `geom_smooth()`.
- Give the plot a title.
- Give credit to the source in the form of a caption (e.g. "R package 'ggplot2' ").
- Change the labels of both axes and the label of the fill legend as shown in figure 1.
- Log-transform both axes. Show minor breaks as suggested at https://ggplot2-book.org/scale-position.html#minor-breaks.
- Place a dollar sign in front of the y-axis tick labels.
- Choose a sequential ColorBrewer palette. Represent the 'neutral' point (i.e. a count of zero) by a light colour.
- Arrange the facets in two rows and four columns. Place the fill legend in the empty space in the bottom right corner.
- Use code chunk options `fig.width`, `fig.height` and `out.width` to adjust figure dimensions. All parts of the figure should be clearly legible without appearing disproportionately large compared to the font size of the running text in the knitted R Markdown file.

(b) What does the plot reveal about the data?

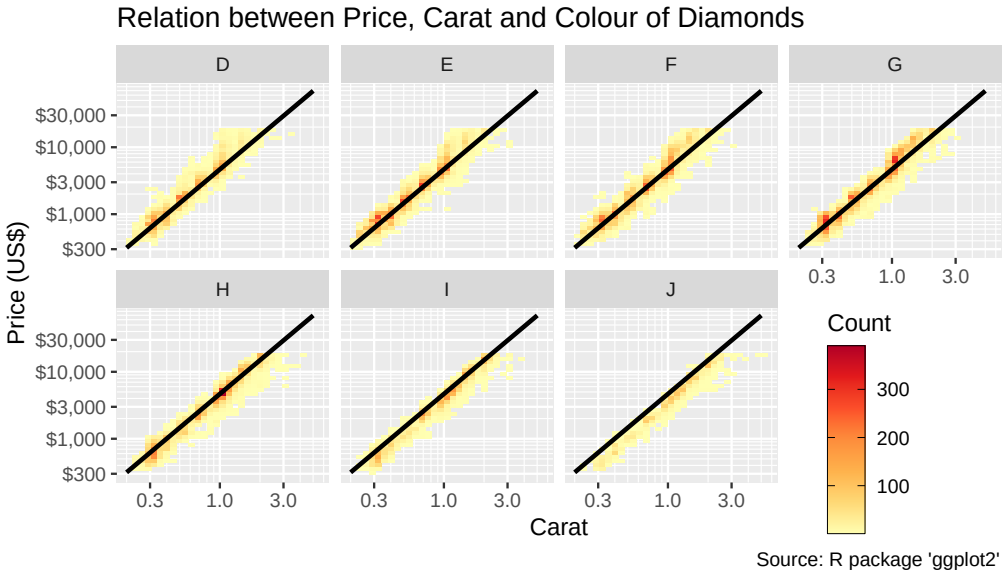

Figure 1: Facetted plot.

6 Relation between brain mass and body mass of animals

The **MASS** package contains a data frame `Animals` with brain masses and body masses for 28 species. This data frame has an unusual format, but it is possible to convert it into a tibble as follows:

```
masses <- as_tibble(Animals, rownames = "species")
```

(a) There are two categories of animals in `masses`:

- Dinosaurs: 'Dipliodocus' (presumably a misspelling of Diplodocus), 'Triceratops' and 'Brachiosaurus'.
- Mammals: all other animals in `masses`.

Add a column `masses$group` that is a factor with levels `Mammal` and `Dinosaur` (in this order).

(b) Make a plot similar to figure 2:

- Show mammals and dinosaurs using different colours and shapes.
- Change the labels of both axes.
- Drop the label for the combined legend of colours and shapes. It is self-evident that mammals and dinosaurs are 'groups'.
- Give the plot a title.
- Give credit to the source in the form of a caption (e.g. "R package 'MASS' ").
- Ensure that the axis tick labels are at consecutive integer powers of 10 and that they appear in the format 10^{-1}, 10^{0}, ...
- Remove all minor grid lines.
- Fix the aspect ratio with `coord_fixed()`. Explain why `coord_fixed()` is a sensible choice for these data.
- Choose the ColorBrewer palette 'Dark2'.
- Adjust the symbol sizes in the legend so that the symbols are easily legible.
- Use code chunk options `fig.width` and `out.width` to adjust figure dimensions. All parts of the figure should be clearly legible without appearing disproportionately large compared to the font size of the running text in the knitted R Markdown file.

(c) What does the plot reveal about the data?

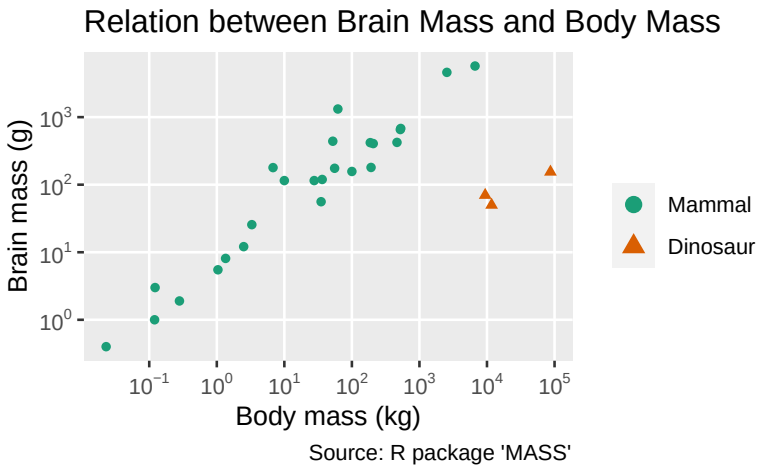

Figure 2: Figure made with **ggplot2**.

7 ASEAN tourism

The spreadsheet at https://michaelgastner.com/DAVisR_data/intra_asean_destination_2018.csv contains information about intra-ASEAN tourism in 2018. The CSV file has two columns:

- `country`: destination
- `million_visits`: the number of visits to `country`

Present the data as a bar chart similar to figure 3.

- Give the plot a title.
- Give credit to the source (ASEAN Statistical Yearbook 2019) in the form of a caption.
- Change the labels of both axes as shown in figure 3.
- Make the bars horizontal.
- Sort countries so that the most (least) visited country is at the top (bottom).
- Show the percentage to the right of each bar.
- Use code chunk options `fig.width`, `fig.height` and `out.width` to adjust figure dimensions. All parts of the figure should be clearly legible without appearing disproportionately large compared to the font size of the running text in the knitted R Markdown file.

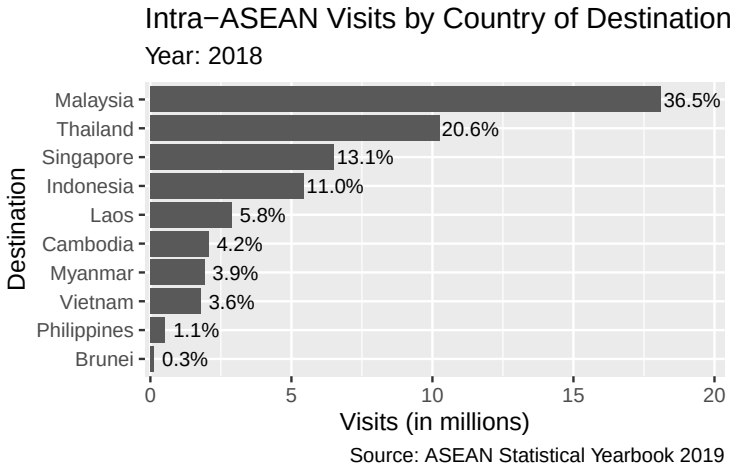

Figure 3: Bar plot made with **ggplot2**.