Assignment instructions

Anynomous

1 Introduction

In addition to R-markdown, Quarto can be used to write the assignment reports. This template contains essentially the same information as the old R-markdown template but we illustrate how you can use Quarto for the assignments.

Some useful resources to get started with Quarto (also an example of a list):

- Getting started with Quarto and Rstudio from the official webpage
- A comprehensive user guide from the official webpage
- Markdown basics
- Quarto FAQ for R-markdown users
- Awesome Quarto - list by Mickaël Canouil

To create your assignment, you can use the assignment-specific templates (recommended, see e.g. the links at the top of assignment 1) or remove the formatting instructions and use this file as a template. Keep the header (the first lines of this file between two lines of —) as it sets the author name to be anonymous, and you can set the title to match the assignment number.

As with R-markdown, you can use the text editor of your choice, but RStudio's editor is probably the easiest and you can choose the formatting (e.g. section headings, bolding, lists, figures, etc.) from the toolbar. Switching between the `source` and `visual` mode allows the quick preview of your formatting.

Note The report should be anonymous and submitted to peergrade.io as `assignmentX.pdf`. Aalto JupyterHub has everything installed and you should be able to render the templates to pdf without any further set-up, but if there are problems contact the TAs or get more information on this from the Quarto documentation. Alternatively, if you have problem with creating a PDF file, start by creating an HTML file and the just print the HTML to a PDF. You may also use other software to create the report PDF, but follow the general instructions in this file (see the pdf version of the template file).

2 Loaded packages

Below are examples of how to load packages that are used in the assignment. After installing the `aaltobda` package (this is pre-installed in Aalto JupyterHub), you need to also load it in the beginning of every notebook where you want to use it with `library()` function (also in Aalto JupyterHub):

```
# To install aaltobda, see the General information in the assignment.

library(aaltobda)
```

3 Including source code

In general, all code needed to produce the essential parts needs to be included, so that it is possible to see, for peer reviewers (and TAs), where errors may have happened.

You can always look at the open rubrics to see how and what is asked for in each exercise.

Try to avoid printing an excessive amount of code and think about what is essential for showing how did you get the result.

Write clear code. The code is also part of your report and clarity of the report affects your score. If the code is not self-explanatory, add comments. In a notebook, you can interleave explaining text and code.

If in doubt additional source code can be included in an appendix.

4 Format instructions

All exercises in the assignment should start with a header fully specifying that it is exercise X, e.g.: (use # in quarto / rmd for a header):

5 Exercise 1)

Subtasks in each assignments should be numbered and use header (use ## for a sub-header).

5.1 a)

For each subtask include necessary textual explanation, equations, code and figures so that the answer to the question flows naturally. You can think what kind of report would you like to review, and what kind of information would make it easier where there is error (if there are errors).

6 Code

In Quarto, code is inserted in a same way as in R-markdown. In fact, Quarto can also render R-markdown documents.

```
5 + 5
```

This R code is evaluated when running the notebook or when rendering to PDF.

If you want to show and run the code, but the output is very long or messy and you prefer to hide the output from the rendered report you can use option `#|`
`results: false`. This is useful especially later as Stan may output many lines.
Note that in Quarto, cell options are specified with the #|-syntax.

```
5 + 5
```

If you want to use some code in the notebook, but think it's not helpful for the reviewers you can exclude it from the generated PDF with option `#| include: false`. You will see the next block in the qmd-file, but not in the generated PDF.

See more on the cell options from Quarto documentation.

7 Plots

Include plots, with a specific width and height for the figure. We can also add label and caption for the plot:

```
data("drowning") # Access the data in aaltobda package
plot(drowning$year, drowning$drownings)
```

Or using `qplot()` from `ggplot2` package:

```
library(ggplot2)
# see themes at https://ggplot2.tidyverse.org/reference/ggtheme.html
theme_set(theme_classic())
qplot(drowning$year, drowning$drownings) +
  labs(x="Year", y="Drownings")
```

Or using `ggplot()` from `ggplot2` package with pipe `|>`

```
drowning |>
  ggplot(aes(x=year, y=drownings)) +
    geom_point() +
    labs(x='Year', y='Number of drownings')
```

Or using `ggplot()` from `ggplot2` package without pipe. In the following code bloc `eval=FALSE` is used to show the code, but not display the same plot again.

```
ggplot(data=drowning, aes(x=year, y=drownings)) +
  geom_point() +
  labs(x='Year', y='Number of drownings')
```

You can then refer the figure using `@yourlabel`-syntax: **?@fig-drowning1**, **?@fig-drowning2**. Figure labels should start with `fig-` prefix. If you label equations or tables, they should start with`eq-` and `tbl-` prefixes respectively.

8 Images

You can include an existing image (e.g. scanned copy of pen and paper equations). We will also add a label for cross-referencing.

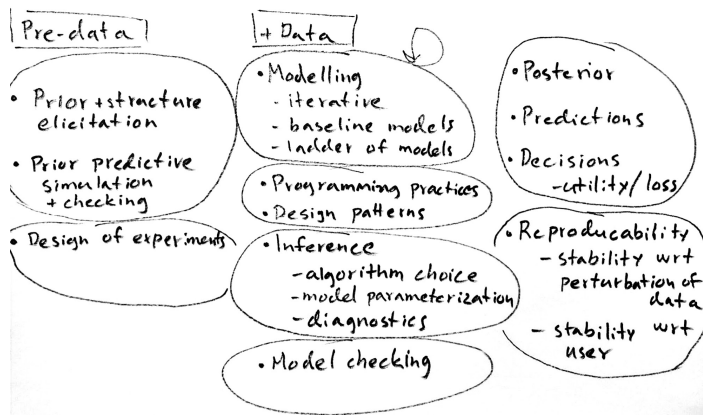See Figure 1 for illustration of parts of Bayesian workflow.

Figure 1: Parts of Bayesian workflow

9 Equations

You can write equations using LaTeX syntax, or you can include them as images if, for example, you use Microsoft Equations.

In Markdown, equations can easily be formulated using LaTeX in line as $f(k) = \binom{n}{k}p^k(1-p)^{n-k}$. Or use the math environment as follows:

$$
\begin{array}{ccc}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23}.
\end{array}
$$

The above example illustrated also multicolumn 'array'. Alternative way to make multiline equations with alignment is to use 'aligned' as follows:

$$
\begin{aligned}
y &\sim \mathrm{normal}(\mu, 1) \\
\mu &\sim \mathrm{normal}(0, 1).
\end{aligned}
$$

Labeling equations allows to refer them later in the text. For example:

$$
p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{1}
$$

Posterior distribution of θ is given by Equation 1 .

If you are new to LaTeX equations, you could use the latext4technics equation editor to create LaTeX equations to include in the report.

A short introduction to equations in LaTeX can be found at https://www.overleaf.com/learn/latex/Mathematical_expressions.

10 Tables

You can use `knitr::kable` to add formatted tables. Captioning and labeling works similarly as with plots.

Compare this to raw output:

```
# Raw output
head(drowning)
```

Table 1: First six rows of the drowning data.

```
library(knitr)
kable(head(drowning), col.names=c("Year","Drownings"))
```

Table 2: First five rows of the mtcars data.

```
kable(head(mtcars), digits=1)
```

It is also possible to control the number of digits, which is helpful to improve readability:

Refer the table in the usual way: see Table 1.

11 Language

The language used in the course is English. Hence the report needs to be written in English.

12 Jupyter Notebook and other report formats

You are allowed to use any format to produce your report, such as Jupyter Notebook, as long as you follow the formatting instructions in this template. Using Quarto with Jupyter Lab is also possible. See getting started guide for Jupyter Lab.