Bayesian Data Analysis – Assignment 9

General information

- The recommended tool in this course is R (with the IDE R-Studio). You can download R **here** and R-Studio **here**. There are tons of tutorials, videos and introductions to R and R-Studio online. You can find some initial hints **here**.
- You can write the report with your preferred software, but the outline of the report should follow the instruction in the R markdown template that can be found **here**.
- Report all results in a single, **anonymous** *.pdf -file and return it to **peergrade.io**.
- Many of the exercises can be checked using the R package `markmyassignment`. Information on how to install and use the package can be found **here**.
- The course has its own R package with data and functionality to simplify coding. To install the package just run the following:
  1. `install.packages("remotes")`
  2. `remotes::install_github("avehtari/BDA_course_Aalto", subdir = "rpackage")`
- Many of the exercises can be checked automatically using the R package `markmyassignment`. Information on how to install and use the package can be found **here**.
- Additional self study exercises and solutions for each chapter in BDA3 can be found **here**.
- We collect common questions regarding installation and technical problems in a course Frequently Asked Questions (FAQ). This can be found **here**.
- If you have any suggestions or improvements to the course material, please feel free to create an issue or submit a pull request to the public repository!!

Information on this assignment

This exercise is related to Chapter 9. The maximum amount of points from this assignment is 6.

Note! This assignment build upon assignment 7, so be sure that assignment 7 is correct before you start with this assignment.

Reading instructions: Chapter 9 in BDA3, see **here**.

Grading instructions: The grading will be done in peergrade. All grading questions and evaluations for assignment 9 can be found **here**

Reporting accuracy: As many significant digits as justified by the Monte Carlo error and posterior accuracy.

Installing and using `rstan`: See the Stan demos on how to use Stan from R. The university Ubuntu desktops have the necessary libraries installed so there should be no need to install anything. To install Stan on your laptop, see the instructions below.

In R, install package `rstan`. Installation instructions on Linux, Mac and Windows can be found at `https://github.com/stan-dev/rstan/wiki/RStan-Getting-Started`. Additional useful packages are `loo`, `bayesplot` and `shinystan` (but you don't need these in this exercise). For Python users, the `Arviz` library may be relevant.

Stan manual can be found at `http://mc-stan.org/documentation/`. From this website, you can also find a lot of other useful material about Stan.

To use markmyassignment for this assignment, run the following code in R:

```
> library(markmyassignment)
> exercise_path <-
+   "https://github.com/avehtari/BDA_course_Aalto/blob/master/exercises/tests/ex9.yml"
> set_assignment(exercise_path)
> # To check your code/functions, just run
> mark_my_assignment()
```

Decision analysis for the factory data (3p)

This exercise is an example of a decision analysis (DA). In a broad context, this means optimizing over different decisions that lead to different outcomes that all have different utilities. In a Bayesian context, this means using posterior distributions to make decisions.

In this exercise, you work as a data analyst in the company that owns the six machines that have produced the data in the `factory` dataset. To access the data, just use:

```
> library(aaltobda)
> data("factory")
```

Your task is to decide whether or not to buy a new (7th) machine for the company. The decision should be based on our best knowledge about the machines.

The following is known about the production process:

- The given data contains quality measurements of single products from the six machines that are ordered from the same seller. (columns: different factories, rows: measurements)
- Customers pay 200 euros for each product.
  - If the quality of the product is below 85, the product cannot be sold
  - All the products that have sufficient quality are sold.
- Raw-materials, the salary of the machine user and the usage cost of the machine for each product cost 106 euros in total.
  - Usage cost of the machine also involves all investment and repair costs divided by the number of products a machine can create. So there is no need to take the investment cost into account as a separate factor.
- The only thing the company owner cares about is money. Thus, as a utility function, use the profit of a new product from a machine.

As noticed in the previous assignment, the hierarchical model fits best with the dataset, so use it to compute the utilities. The assumptions for the hierarchical model are the same as in assignment 7. If you did things correctly then, solving the assignment only requires you to change the "generated quantities"-block in the Stan-code to compute the correct predictive samples for products of all 7 (= 6 + 1) machines.

Your task is the following:

1. For each of the six machines, compute and report the expected utility of one product of that machine. Below is a test case on how the utility function should work (and that you can test with `markmyassignment`. **Note!** This is just a test case to test that your utility function works. In the report, you should report the expected utility using your posterior draws from Stan.

   ```
   > y_pred <- c(123.80, 85.23, 70.16, 80.57, 84.91)
   > utility(draws = y_pred)
   ```

   ```
   [1] -26
   ```

2. Rank the machines based on the expected utilities (in other words order the machines from worst to best) and explain briefly what the utility values tell about the quality of these machines. E.g. Tell which machines are profitable and which are not (if any).
3. Compute and report the expected utility of the products of a new (7th) machine.
4. Based on your analysis, discuss briefly whether the company owner should buy a new (7th) machine.
5. As usual, remember to include the source code (for both Stan and R)!