Bayesian Data Analysis – Assignment 8

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 7. The maximum amount of points from this assignment is 6.

Reading instructions: Chapter 7 in BDA3, see **here**. Also read the paper on PSIS-LOO that can be found **here** or **here**.

Grading instructions: The grading will be done in peergrade. All grading questions and evaluations for assignment 8 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.

Model assessment: LOO-CV for factory data with Stan (6p)

Use leave-one-out cross-validation (LOO-CV) to assess the predictive performance of the pooled, separate and hierarchical Gaussian models for the factory dataset (see the second exercise in Assignment 7). To read in the data, just use:

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

PSIS-LOO is a recently developed method for approximating the exact LOO and is thus not in BDA3. For more information, see the lecture slides and the original paper **here** or **here**.

Use Stan for fitting the models, and the `loo` R package for computing the approximate LOO-CV given the posterior samples provided by Stan. You can install the package as

```
> install.packages(loo)
```

Python users can use PSIS-LOO implementation in ArviZ library.

The report should include the following parts.

1. Fit the models with Stan as instructed in Assignment 7. To use the `loo` or `psisloo` functions, you need to compute the log-likelihood values of each observation for every posterior draw (i.e. an S-by-N matrix, where S is the number of posterior draws and $N = 30$ is the total number of observations). This can be done in the `generated quantities` block in the Stan code; for a demonstration, see the Gaussian linear model `lin.stan` in the R Stan examples that can be found **here**.

2. Compute the PSIS-LOO elpd values and the $\hat{k}$-values for each of the three models.

   Hint! It will be convenient to visualize the $\hat{k}$-values for each model so that you can easily see how many of these values fall in the range $\hat{k} > 0.7$ to assess the reliability of the PSIS-LOO estimate for each model. You can read more about the theoretical guarantees for the accuracy of the estimate depending on $\hat{k}$ from the original article (see **here** or **here**), but regarding this assignment, it suffices to understand that if all the $\hat{k}$-values are $\hat{k} \lesssim 0.7$, the PSIS-LOO estimate can be considered to be reliable, otherwise there is a concern that it may be biased (too optimistic, overestimating the predictive accuracy of the model).

3. Compute the effective number of parameters p_{eff} for each of the three models.

   Hint! The estimated effective number of parameters in the model can be computed from equation (7.15) in the book, where $\text{elpd}_{\text{loo-cv}}$ is the PSIS-LOO value (sum of the LOO log densities) and lpd is given by equation (7.5) in the book.

4. Assess how reliable the PSIS-LOO estimates are for the three models based on the $\hat{k}$-values.

5. An assessment of whether there are differences between the models with regard to the $\text{elpd}_{\text{loo-cv}}$, and if so, which model should be selected according to PSIS-LOO.