Bayesian data analysis – exercise 6

This assignment is related to Chapters 10 and 11.

The maximum amount of points from this assignment is 6. In addition to the correctness of the answers, the overall quality and clearness of the report is evaluated.

Report all results to a single, **anonymous** *.pdf -file and return it to peergrade.io. Include also source code to the report (either as an attachment or as a part of the answer). By anonymity it is meant that the report should not contain your name or student number.

1. Generalized linear model: Bioassay with Stan (6 points)

Replicate the computations for the bioassay example of section 3.7 (BDA3) using Stan.

Information and hints

- See the Stan demos on how to use Stan from R or Python.
- 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`.
- In Python, you can use Stan by importing the module `pystan`. To install the Python interface to your own laptop, follow instructions here `http://pystan.readthedocs.io/en/latest/getting_started.html`.
- Use uniform prior as in the book $p(\alpha, \beta) \propto 1$.
- You can use Stan's default settings for the number of chains, samples per chain and warm-up length.
- Use $\hat{R}$ to assess convergence. Note that Stan gives you these values automatically; in R type `print(fit)` or simply `fit` and in Python `print fit`. Here `fit` is the fit object returned by function `stan`. Report the $\hat{R}$ values both for α and β and draw conclusions about the convergence of the chains. **This means that you should briefly explain how to interpret the obtained $\hat{R}$ values**.
- Plot the samples for α and β (scatter plot) and compare to the Figure 3.3b in BDA3 to verify visually that your code works. You can also compute and visualize the density in a grid (as in Fig. 3.3a) but this is not mandatory.
- 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.
- Useful Stan function: `binomial_logit`