

Exercise G: Poisson and STAN

Benedikt Ehinger

21. Januar 2017

Videos O: Hierarchical Models

Recommended Reading Material:

- DBDA Chapter 9

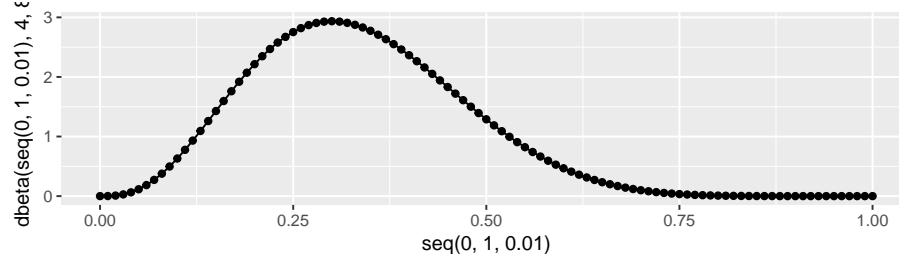
1 Hierarchical Bernoulli

We have a simple experiment where $n_{sub} = 20$ subjects were tested whether they can correctly classify whether a depicted image contains an animal or not. For this they were shown images of animals intermixed with control-stimuli without animals and decided using buttonpresses whether there was an animal present ($n_{trials} = 50$). We only analyze the number of correctly identified items (usually you would use the signal detection framework with false positives, false negatives etc. we only look at hits).

The data are generated according to the following code:

```
set.seed(3)
n_sub = 20
n_trials = 50
alpha = 4
beta = 8
theta_sub <- rbeta(n_sub, alpha, beta)
y_correct = rbinom(rep(1, n_sub), n_trials, theta_sub)
```

The population distribution looks like this:



The model is the following (again replace all %% with your own code):

```

model_bern <- "
data {
  %% # initialise the variables 'n_trials', 'n_sub' and 'y_correct'.
  # remember that y_correct is a list
}
parameters {
  %% # theta_sub is a list (size %) of real values between 0 and 1
  real<lower=0> alpha;
  real<lower=0> beta;
}
model {
  // Priors for Theta
  for(sub in 1:%){
    %% # Each theta_sub should be samples from the same beta
    # distribution with parameters alpha and beta

    %% # y_correct is sampled from a 'binomial' distribution.
    # Note that we could get partially rid for-loop by
    # vectorizing the assignment (check out the STAN manual)
  }
}"

```

And we can run the model as previously described with:

```

library(rstan)
samples <- stan(model_code=model_bern,
               data=list(n_trials=n_trials,
                         n_sub = n_sub,
                         y_correct=y_correct),
               iter=1000, chains=4)

```

1. Finish the model code and run the model.
2. Check for convergence (traceplot + Rhat).
3. Plot and interpret the parameters alpha and beta (pairs(samples, pars=c("beta", "alpha"))).
4. We want to look at 'shrinkage'. Compare the observed values (summary(samples)\$summary[1:n_sub,1]) with the maximum likelihood values (y_correct/n_trials)¹. Describe if and how you see shrinkage.
5. Calculate and visualize the population-beta-distribution for each alpha/-beta posterior value (in a case of 1000 iteration, this should result in 500 lines). You might want to use transparency/alpha².

¹These values represent individual model fits, i.e. if you would estimate the maximum likelihood θ for each subject independently

²Tip: use `extract(samples, pars=c("alpha", "beta"))` to get the values

6. change the value of alpha and beta in the simulation code both to 1. What do you observe in your previous plots and on shrinkage?