BDA - Assignment 8

Anonymous

Contents

Exercise 1)

```
library(posterior)
library(cmdstanr)
library(aaltobda)
data("factory")
```

Separate model

```
stan_data <- list(
  y = factory,
  N = 5,
  J=6
)
mod_sep <- cmdstan_model("BDA/factory_sep_1.stan")
mod_sep$print()
```

```
## data {
##   int<lower=0> N;
##   int<lower=0> J;
##   vector[J] y[N];
## }
##
## parameters {
##   vector[J] mu;
##   vector<lower=0>[J] sigma;
## }
```

```
##
## model {
##   // priors
##   for (j in 1:J){
##     mu[j] ~ normal(0, 10);
##     sigma[j] ~ gamma(1,1);
##   }
##
##   // likelihood
##   for (j in 1:J)
##     y[,j] ~ normal(mu[j], sigma[j]);
## }
##
## generated quantities {
##   real ypred;
##   vector[N] log_lik;
##   // Compute predictive distribution
##   // for the first machine
##   ypred = normal_rng(mu[6], sigma[6]);
##   for (i in 1:N){
##     log_lik[i] = normal_lpdf(y[i] | mu, sigma);
##
##   }
## }
```

```
fit_sep <- mod_sep$sample(data = stan_data)
```

Pooled model

```
mod_pooled <- cmdstan_model("BDA/factory_pooled_1.stan")
mod_pooled$print()
```

```
## data {
## int<lower=0> N;
## int<lower=0> J;
## vector[J] y[N];
## }
## parameters {
## real mu;
## real<lower=0> sigma;
## }
## model {
##   // prior
##   mu ~ normal(0, 10);
##   sigma ~ gamma(1,1);
##
##   for (j in 1:J)
##     y[,j] ~ normal(mu, sigma);
## }
## generated quantities {
##   real ypred;
```

```
##   vector[J] log_lik[N];
##   // Compute predictive distribution
##   // for the seventh machine
##   ypred = normal_rng(mu, sigma);
##   for (i in 1:N) {
##     for (j in 1:J) {
##       log_lik[i,j] = normal_lpdf(y[i,j] | mu, sigma);
##     }
##   }
## }
```

```r
fit_pooled <- mod_pooled$sample(data = stan_data)
```

Hierarchical model

```r
mod_hier <- cmdstan_model("BDA/factory_hier_1.stan")
mod_hier$print()
```

```
## data {
## int<lower=0> N;
## int<lower=0> J;
## vector[J] y[N];
## }
## parameters {
## real mu_P;
## real<lower=0> sigma_P;
## real<lower=0> sigma;
## vector[J] theta;
## }
## model {
##   // prior
##   mu_P ~ normal(0, 10);
##   sigma_P ~ gamma(1,1);
##   sigma ~ gamma(1,1);
##
##   for (j in 1:J){
##     theta[j] ~ normal(mu_P, sigma_P);
## }
##   for (j in 1:J)
##     y[,j] ~ normal(theta[j], sigma);
## }
## generated quantities {
##   real ypred;
##   vector[J] log_lik[N];
##   real theta_pred= normal_rng(mu_P, sigma_P);
##   // Compute predictive distribution
##   // for the seventh machine
##   ypred = normal_rng(theta[6], sigma);
##   for (i in 1:N) {
##     for (j in 1:J) {
##       log_lik[i,j] = normal_lpdf(y[i,j] | theta[j], sigma);
```

```
##     }
##   }
##
## }
```

```
fit_hier <- mod_hier$sample(data = stan_data)
```

Exercise 2 + 3)

The PSIS-LOO elpd values, the $\hat{k}$ values, and the effective number of parameters p_eff (`p_loo`) are computed with built-in functions from `loo` library.

Separate model

```
fit_sep$loo()
```

```
## Warning: Some Pareto k diagnostic values are too high. See help('pareto-k-diagnostic') for details.
```

```
##
## Computed from 4000 by 5 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -196.0 12.4
## p_loo        19.1  2.1
## looic       392.0 24.9
## ------
## Monte Carlo SE of elpd_loo is NA.
##
## Pareto k diagnostic values:
##                          Count Pct.    Min. n_eff
## (-Inf, 0.5]   (good)     1     20.0%   321
##  (0.5, 0.7]   (ok)       0      0.0%   <NA>
##    (0.7, 1]   (bad)      4     80.0%   44
##    (1, Inf)   (very bad) 0      0.0%   <NA>
## See help('pareto-k-diagnostic') for details.
```

Pooled model

```
fit_pooled$loo()
```

```
##
## Computed from 4000 by 30 log-likelihood matrix
##
##          Estimate  SE
## elpd_loo   -134.6 4.9
## p_loo         2.6 0.9
```

```
## looic       269.3 9.8
## ------
## Monte Carlo SE of elpd_loo is 0.1.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

Hierarchical model

```
fit_hier$loo()
```

```
## Warning: Some Pareto k diagnostic values are slightly high. See help('pareto-k-diagnostic') for deta
```

```
##
## Computed from 4000 by 30 log-likelihood matrix
##
##          Estimate  SE
## elpd_loo   -129.4 4.7
## p_loo         8.5 1.7
## looic       258.9 9.3
## ------
## Monte Carlo SE of elpd_loo is 0.1.
##
## Pareto k diagnostic values:
##                          Count Pct.    Min. n_eff
## (-Inf, 0.5]   (good)     26    86.7%   390
##  (0.5, 0.7]   (ok)        4    13.3%   188
##    (0.7, 1]   (bad)       0     0.0%   <NA>
##    (1, Inf)   (very bad)  0     0.0%   <NA>
##
## All Pareto k estimates are ok (k < 0.7).
## See help('pareto-k-diagnostic') for details.
```

Exercise 4)

Based on the calculated $\hat{k}$ values above, we can see that the separate model has all values much higher than 0.7, while pooled and hierarchical model have reasonably low values. Therefore, the two latter model are more reliable and separate PSIS-LOO estimates is not very reliable.

Exercise 5)

According to the values returned above, the hierarchical model has the highest elpd_loo value, i.e. highest PSIS-LOO value, at around -130, while the pooled has slightly lower value, at -135 and the separate model has significantly low -197 PSIS-LOO value.

Thus, there are differences between the models and the hierarchical model should be selected.