BDA - Project brms library test

Eetu Suominen

Contents

Load packages

```
library(aaltobda)
library(cmdstanr)
library(brms)
library(ggplot2); theme_set(theme_grey())
library(grid)
library(gridExtra)
library(bayesplot)
library(ggdist)
theme_set(bayesplot::theme_default(base_family = "sans"))
library(rprojroot)
library(brms)
library(caret)
library(corrplot)
library(dplyr)
library(crosstable)
library(lme4)
SEED <- 614273
```

Scaler functions

```
min_max_scaler <- function(values){
  scaled_data = (values - min(values)) / (max(values) - min(values))
  return(scaled_data)
}

descaler <- function(values, max, min){
  descaled_data = values*(max-min) + min
  return(descaled_data)
}

standardize <- function(values){
  standardized_data = (values - mean(values)) / sd(values)
  return(standardized_data)
}

de_standardize <- function(values,sd_values, mean_values){
  de_standardized_data = values*sd_values + mean_values
  return(de_standardized_data)
}
```

Train / test data

```
train_data <- read.csv('./Data/train-data.csv')
test_data <- read.csv('./Data/test-data.csv')
```

Prios

```
pr1 = (prior(normal(0,1), class = "b", coef = "scaled_age"))


pr2 = (prior(normal(0,1), class = "b", coef = "scaled_age") +
```

```
       prior(student_t(3,0,1), class="sd", group="smoker"))

pr3 = (prior(normal(0,1), class = "b", coef = "scaled_age") +
       prior(normal(0,1), class="b", coef ="scaled_bmi") +
       prior(student_t(3,0,1), class="sd", group="smoker"))

pr4 = (prior(normal(0,1), class = "b", coef = "scaled_age") +
       prior(normal(0,1), class="b", coef ="scaled_bmi") +
       prior(student_t(3,0,1), class="sd", group="smoker") +
       prior(student_t(3,0,1), class="sd", group="region"))

pr5 = (prior(normal(0,1), class = "b", coef = "scaled_age") +
       prior(normal(0,1), class="b", coef ="scaled_bmi") +
       prior(normal(0,1), class="b", coef ="scaled_children") +
       prior(student_t(3,0,1), class="sd", group="smoker") +
       prior(student_t(3,0,1), class="sd", group="region"))

pr6_1 = prior(normal(0,1), class = 'b')

pr6_2 = (prior(normal(0,1), class = "b", coef = "scaled_age") +
         prior(normal(0,1), class = "b", coef = "sexmale") +
         prior(normal(0,1), class="b", coef ="scaled_bmi") +
         prior(normal(0,1), class="b", coef ="scaled_children") +
         prior(student_t(3,0,1), class="sd", group="smoker") +
         prior(student_t(3,0,1), class="sd", group="region"))
```

Models

Frequentist model

```
basic_model = lmer(formula = scaled_charges ~ scaled_age + scaled_bmi + children + sex + (1|smoker) + (
summary(basic_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: scaled_charges ~ scaled_age + scaled_bmi + children + sex + (1 |
##     smoker) + (1 | region)
##    Data: train_data
##
## REML criterion at convergence: 1481
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9954 -0.4558 -0.1684  0.2506  4.8627
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  smoker   (Intercept) 1.993573 1.41194
##  region   (Intercept) 0.001654 0.04067
##  Residual             0.247010 0.49700
## Number of obs: 1004, groups:  smoker, 2; region, 2
##
## Fixed effects:
##             Estimate Std. Error t value
```

```
## (Intercept)  0.53542    0.99923   0.536
## scaled_age   0.30231    0.01587  19.051
## scaled_bmi   0.16715    0.01611  10.377
## children     0.03916    0.01281   3.058
## sexmale      0.03119    0.03148   0.991
##
## Correlation of Fixed Effects:
##             (Intr) scld_g scld_b chldrn
## scaled_age   0.001
## scaled_bmi   0.001 -0.135
## children    -0.014 -0.061 -0.026
## sexmale     -0.016  0.020 -0.058 -0.014
```

Train the Bayesian models

```
model_1 = brm(
  scaled_charges ~ scaled_age,
  data  = train_data,
  prior = pr1,
  cores = 4,
  iter = 4000,
  file = "./Models/model_1"
)

model_2 = brm(
  scaled_charges ~ scaled_age + (1|smoker),
  data  = train_data,
  prior = pr2,
  cores = 4,
  iter = 4000,
  file = "./Models/model_2"
)

model_3 = brm(
  scaled_charges ~ scaled_age + scaled_bmi + (1|smoker),
  data  = train_data,
  prior = pr3,
  cores = 4,
  iter = 4000,
  file = "./Models/model_3"
)

model_4 = brm(
  scaled_charges ~ scaled_age + scaled_bmi + (1|smoker) + (1|region),
  data  = train_data,
  prior = pr4,
  cores = 4,
  iter = 4000,
  file = "./Models/model_4"
)

model_5 = brm(
  scaled_charges ~ scaled_age + scaled_bmi + scaled_children + (1|smoker) + (1|region),
  data  = train_data,
  prior = pr5,
```

```
  cores = 4,
  iter = 4000,
  file = "./Models/model_5"
)

model_6_1 = brm(
  scaled_charges ~ scaled_age + sex + scaled_bmi + scaled_children +  smoker + region,
  data  = train_data,
  prior = pr6_1,
  cores = 4,
  iter = 4000,
  file = "./Models/model_6_1"
)

model_6_2 = brm(
  scaled_charges ~ scaled_age + sex + scaled_bmi + scaled_children +  (1|smoker) + (1|region),
  data  = train_data,
  prior = pr6_2,
  cores = 4,
  iter = 4000,
  file = "./Models/model_6_2"
)
```

Convergence diagnostics

```
summary(model_1)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: scaled_charges ~ scaled_age
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
##
## Population-Level Effects:
##            Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      0.01      0.03    -0.05     0.06 1.00     8042     5866
## scaled_age     0.32      0.03     0.25     0.38 1.00     7826     6445
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.97      0.02     0.93     1.01 1.00     8282     6286
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_1 = loo(model_1)
loo_1
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
```

```
##          Estimate   SE
## elpd_loo  -1391.5 30.6
## p_loo         3.9  0.3
## looic      2783.1 61.2
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
summary(model_2)
```

```
## Warning: There were 114 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See http://mc-stan.org/misc/
## warnings.html#divergent-transitions-after-warmup
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: scaled_charges ~ scaled_age + (1 | smoker)
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
##
## Group-Level Effects:
## ~smoker (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.45      0.69     0.58     3.25 1.01      913      366
##
## Population-Level Effects:
##            Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      0.45      0.95    -1.63     2.32 1.00     1809     1445
## scaled_age     0.33      0.02     0.29     0.36 1.00     2686      859
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.53      0.01     0.50     0.55 1.00     5358     4568
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_2 = loo(model_2)
loo_2
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -782.9 33.9
## p_loo         6.8  0.6
## looic      1565.7 67.8
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
summary(model_3)
```

```
## Warning: There were 100 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See http://mc-stan.org/misc/
## warnings.html#divergent-transitions-after-warmup
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: scaled_charges ~ scaled_age + scaled_bmi + (1 | smoker)
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
##
## Group-Level Effects:
## ~smoker (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.42      0.64     0.59     3.09 1.00     2136     2325
##
## Population-Level Effects:
##            Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      0.43      0.97    -1.67     2.31 1.00     2250     2397
## scaled_age     0.31      0.02     0.27     0.34 1.00     5815     4193
## scaled_bmi     0.16      0.02     0.13     0.19 1.00     5988     5524
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.50      0.01     0.48     0.52 1.00     6431     4890
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_3 = loo(model_3)
loo_3
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -732.7 34.2
## p_loo         7.4  0.7
## looic      1465.3 68.3
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
summary(model_4)
```

```
## Warning: There were 152 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See http://mc-stan.org/misc/
## warnings.html#divergent-transitions-after-warmup
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
```

```
## Formula: scaled_charges ~ scaled_age + scaled_bmi + (1 | smoker) + (1 | region)
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
##
## Group-Level Effects:
## ~region (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     0.36      0.46     0.01     1.64 1.00     1692     2251
##
## ~smoker (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.44      0.68     0.60     3.31 1.00     3219     2787
##
## Population-Level Effects:
##            Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      0.43      0.98    -1.71     2.38 1.00     2906     2209
## scaled_age     0.30      0.02     0.27     0.34 1.00     6686     5017
## scaled_bmi     0.17      0.02     0.14     0.20 1.00     6699     5442
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.50      0.01     0.48     0.52 1.00     7489     5234
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_4 = loo(model_4)
loo_4
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -731.7 34.2
## p_loo         8.5  0.7
## looic      1463.4 68.3
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
summary(model_5)
```

```
## Warning: There were 215 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See http://mc-stan.org/misc/
## warnings.html#divergent-transitions-after-warmup
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: scaled_charges ~ scaled_age + scaled_bmi + scaled_children + (1 | smoker) + (1 | region)
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
```

```
##
## Group-Level Effects:
## ~region (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     0.35      0.43     0.01     1.57 1.00     1734     2567
##
## ~smoker (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.40      0.62     0.58     3.01 1.00     2995     3170
##
## Population-Level Effects:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept           0.41      0.95    -1.60     2.32 1.00     2402     2511
## scaled_age          0.30      0.02     0.27     0.33 1.00     6579     5535
## scaled_bmi          0.17      0.02     0.14     0.20 1.00     6828     4793
## scaled_children     0.05      0.02     0.02     0.08 1.00     6709     5314
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.50      0.01     0.48     0.52 1.00     5924     4764
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_5 = loo(model_5)
loo_5
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -728.1 34.2
## p_loo         9.5  0.8
## looic      1456.2 68.5
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
summary(model_6_1)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: scaled_charges ~ scaled_age + sex + scaled_bmi + scaled_children + smoker + region
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
##
## Population-Level Effects:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept          -0.39      0.03    -0.44    -0.33 1.00    15197     6475
## scaled_age          0.30      0.02     0.27     0.33 1.00    13909     6425
## sexmale             0.03      0.03    -0.03     0.09 1.00    15086     5317
```

```
## scaled_bmi          0.17      0.02     0.14     0.20 1.00    13181     6680
## scaled_children     0.05      0.02     0.02     0.08 1.00    13261     5480
## smokeryes           1.99      0.04     1.92     2.07 1.00    15187     5769
## regionsouth        -0.07      0.03    -0.13    -0.00 1.00    12716     6365
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.50      0.01     0.48     0.52 1.00    13769     5851
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_6_1 = loo(model_6_1)
loo_6_1
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -728.3 34.1
## p_loo        10.3  0.9
## looic      1456.7 68.2
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
summary(model_6_2)
```

```
## Warning: There were 273 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See http://mc-stan.org/misc/
## warnings.html#divergent-transitions-after-warmup
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: scaled_charges ~ scaled_age + sex + scaled_bmi + scaled_children + (1 | smoker) + (1 | regi
##    Data: train_data (Number of observations: 1004)
##   Draws: 4 chains, each with iter = 4000; warmup = 2000; thin = 1;
##          total post-warmup draws = 8000
##
## Group-Level Effects:
## ~region (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     0.37      0.45     0.01     1.62 1.00     1132     2123
##
## ~smoker (Number of levels: 2)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.45      0.67     0.59     3.29 1.01      921      240
##
## Population-Level Effects:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept           0.43      1.01    -1.63     2.49 1.01     1131      228
## scaled_age          0.30      0.02     0.27     0.33 1.00     4839     5625
## sexmale             0.03      0.03    -0.03     0.09 1.00     4929     5484
```

```
## scaled_bmi          0.17      0.02     0.14     0.20 1.00     5750     5747
## scaled_children     0.05      0.02     0.02     0.08 1.00     2650     4988
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.50      0.01     0.48     0.52 1.00     3861     5317
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
loo_6_2 = loo(model_6_2)
loo_6_2
```

```
##
## Computed from 8000 by 1004 log-likelihood matrix
##
##          Estimate   SE
## elpd_loo   -728.5 34.2
## p_loo        10.5  0.9
## looic      1457.1 68.3
## ------
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

Loo comparison

```
loo_compare(loo_1,loo_2,loo_3,loo_4,loo_5,loo_6_1,loo_6_2)
```

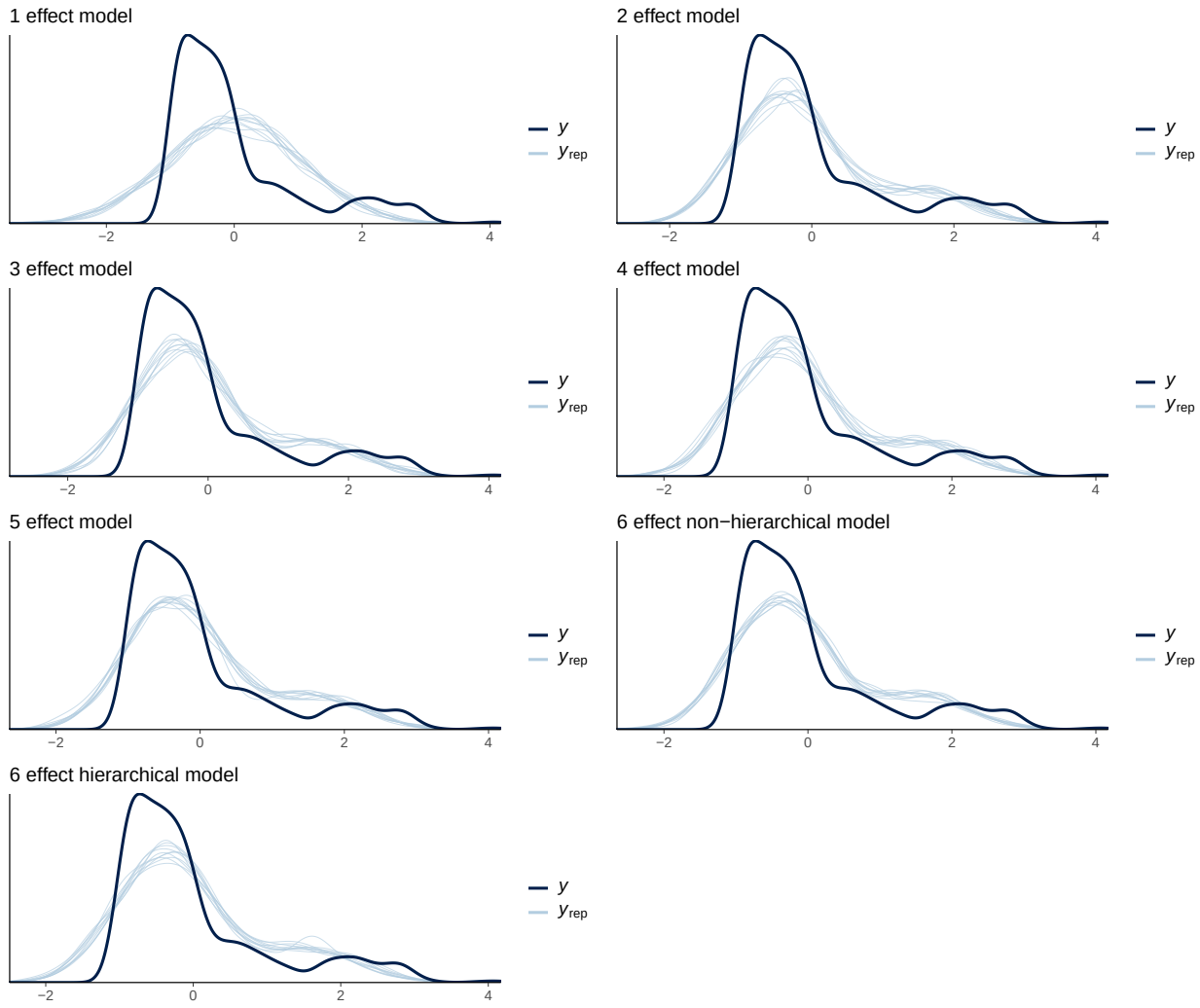
```
##           elpd_diff se_diff
## model_5      0.0       0.0
## model_6_1   -0.3       1.0
## model_6_2   -0.5       1.0
## model_4     -3.6       2.9
## model_3     -4.6       3.7
## model_2    -54.8      10.7
## model_1   -663.5      36.7
```

Posterior predictive checks

```
p1 <- pp_check(model_1) +
  ggtitle("1 effect model")
p2 <- pp_check(model_2) +
  ggtitle("2 effect model ")
p3 <- pp_check(model_3) +
  ggtitle("3 effect model ")
p4 <- pp_check(model_4) +
  ggtitle("4 effect model ")
p5 <- pp_check(model_5) +
  ggtitle("5 effect model")
p6_1 <- pp_check(model_6_1) +
  ggtitle("6 effect non-hierarchical model")
p6_2 <- pp_check(model_6_2) +
```

```
  ggtitle("6 effect hierarchical model")
```

```
grid.arrange(p1, p2, p3, p4, p5, p6_1, p6_2, nrow=4)
```



Predictive performance assessment with test data

```
pp_1 = posterior_predict(model_1,newdata=test_data)
pp_2 = posterior_predict(model_2,newdata=test_data)
pp_3 = posterior_predict(model_3,newdata=test_data)
pp_4 = posterior_predict(model_4,newdata=test_data)
pp_5 = posterior_predict(model_5,newdata=test_data)
pp_6_1 = posterior_predict(model_6_1,newdata=test_data)
pp_6_2 = posterior_predict(model_6_2,newdata=test_data)
pp_lm = predict(basic_model, newdata=test_data)

rs_1 = bayes_R2(model_1, newdata = test_data)
rs_2 = bayes_R2(model_2, newdata = test_data)
rs_3 = bayes_R2(model_3, newdata = test_data)
rs_4 = bayes_R2(model_4, newdata = test_data)
```

```r
rs_5 = bayes_R2(model_5, newdata = test_data)
rs_6_1 = bayes_R2(model_6_1, newdata = test_data)
rs_6_2 = bayes_R2(model_6_2, newdata = test_data)

sprintf("Model 1 R-squared: %f", median(rs_1))
```

```
## [1] "Model 1 R-squared: 0.087849"
```

```r
sprintf("Model 2 R-squared: %f", median(rs_2))
```

```
## [1] "Model 2 R-squared: 0.708724"
```

```r
sprintf("Model 3 R-squared: %f", median(rs_3))
```

```
## [1] "Model 3 R-squared: 0.730025"
```

```r
sprintf("Model 4 R-squared: %f", median(rs_4))
```

```
## [1] "Model 4 R-squared: 0.730129"
```

```r
sprintf("Model 5 R-squared: %f", median(rs_5))
```

```
## [1] "Model 5 R-squared: 0.731718"
```

```r
sprintf("Model 6_1 R-squared: %f", median(rs_6_1))
```

```
## [1] "Model 6_1 R-squared: 0.730025"
```

```r
sprintf("Model 6_2 R-squared: %f", median(rs_6_2))
```

```
## [1] "Model 6_2 R-squared: 0.730666"
```

```r
# descaling the data
c_sd = sd(train_data$charges)
c_mean = mean(train_data$charges)

# de-scaled values
pp_1_des = de_standardize(colMeans(pp_1), c_sd, c_mean)
pp_2_des = de_standardize(colMeans(pp_2), c_sd, c_mean)
pp_3_des = de_standardize(colMeans(pp_3), c_sd, c_mean)
pp_4_des = de_standardize(colMeans(pp_4), c_sd, c_mean)
pp_5_des = de_standardize(colMeans(pp_5), c_sd, c_mean)
pp_6_1_des = de_standardize(colMeans(pp_6_1), c_sd, c_mean)
pp_6_2_des = de_standardize(colMeans(pp_6_2), c_sd, c_mean)
pp_lm_des = de_standardize(pp_lm, c_sd, c_mean)
```

Calculate prediction errors

```r
calculate_rmse <- function(true,predicted){
  rmse = sqrt(mean((true - predicted)^2))
  return(rmse)
}

calculate_r2 <- function(true, predicted){
  rss = sum((true - predicted)^2)
  tss = sum((true - mean(true))^2)
  return(1 - rss/tss)
}
```

```
rmse_1 = calculate_rmse(test_data$charges, (pp_1_des))
rmse_2 = calculate_rmse(test_data$charges, (pp_2_des))
rmse_3 = calculate_rmse(test_data$charges, (pp_3_des))
rmse_4 = calculate_rmse(test_data$charges, (pp_4_des))
rmse_5 = calculate_rmse(test_data$charges, (pp_5_des))
rmse_6_1 = calculate_rmse(test_data$charges, (pp_6_1_des))
rmse_6_2 = calculate_rmse(test_data$charges, (pp_6_2_des))
rmse_lm = calculate_rmse(test_data$charges, (pp_lm_des))


r2_1 = calculate_r2(test_data$charges, (pp_1_des))
r2_2 = calculate_r2(test_data$charges, (pp_2_des))
r2_3 = calculate_r2(test_data$charges, (pp_3_des))
r2_4 = calculate_r2(test_data$charges, (pp_4_des))
r2_5 = calculate_r2(test_data$charges, (pp_5_des))
r2_6_1 = calculate_r2(test_data$charges, (pp_6_1_des))
r2_6_2 = calculate_r2(test_data$charges, (pp_6_2_des))
r2_lm = calculate_r2(test_data$charges, (pp_lm_des))


print("RMSE:")
```

```
## [1] "RMSE:"
```

```
sprintf("Model_1: %f", rmse_1)
```

```
## [1] "Model_1: 11184.732318"
```

```
sprintf("Model_2: %f", rmse_2)
```

```
## [1] "Model_2: 6529.636789"
```

```
sprintf("Model_3: %f", rmse_3)
```

```
## [1] "Model_3: 6240.216918"
```

```
sprintf("Model_4: %f", rmse_4)
```

```
## [1] "Model_4: 6225.925695"
```

```
sprintf("Model_5: %f", rmse_5)
```

```
## [1] "Model_5: 6203.878391"
```

```
sprintf("Model_6_1: %f", rmse_6_1)
```

```
## [1] "Model_6_1: 6235.945777"
```

```
sprintf("Model_6_2: %f", rmse_6_2)
```

```
## [1] "Model_6_2: 6227.857655"
```

```
sprintf("Model_lm: %f", rmse_lm)
```

```
## [1] "Model_lm: 6231.468982"
```

```
print("R^2 scores:")
```

```
## [1] "R^2 scores:"
```

```
sprintf("Model_1: %f", r2_1)
```

```
## [1] "Model_1: 0.059054"
```

```
sprintf("Model_2: %f", r2_2)
```

```
## [1] "Model_2: 0.679305"
```

```
sprintf("Model_3: %f", r2_3)
```

```
## [1] "Model_3: 0.707104"
```

```
sprintf("Model_4: %f", r2_4)
```

```
## [1] "Model_4: 0.708444"
```

```
sprintf("Model_5: %f", r2_5)
```

```
## [1] "Model_5: 0.710506"
```

```
sprintf("Model_6_1: %f", r2_6_1)
```

```
## [1] "Model_6_1: 0.707505"
```

```
sprintf("Model_6_2: %f", r2_6_2)
```

```
## [1] "Model_6_2: 0.708263"
```

```
sprintf("Model_lm: %f", r2_lm)
```

```
## [1] "Model_lm: 0.707925"
```

```
p1 = ggplot(test_data, aes(x=charges)) + geom_density() + theme_gray() + ggtitle("Test data")
p2 = ggplot() + geom_density(aes(x=pp_1_des)) + theme_gray() + ggtitle("1 effect model") + labs(x = "cha
p3 = ggplot() + geom_density(aes(x=pp_2_des)) + theme_gray() + ggtitle("2 effect model") + labs(x = "cha
p4 = ggplot() + geom_density(aes(x=pp_3_des)) + theme_gray() + ggtitle("3 effect model") + labs(x = "cha
p5 = ggplot() + geom_density(aes(x=pp_4_des)) + theme_gray() + ggtitle("4 effect model") + labs(x = "cha
p6 = ggplot() + geom_density(aes(x=pp_5_des)) + theme_gray() + ggtitle("5 effect model") + labs(x = "cha
p7 = ggplot() + geom_density(aes(x=pp_6_1_des)) + theme_gray() + ggtitle("6 effect non-hierarchical mod
p8 = ggplot() + geom_density(aes(x=pp_6_2_des)) + theme_gray() + ggtitle("6 effect hierarchical model")
p9 = ggplot() + geom_density(aes(x=pp_lm_des)) + theme_gray() + ggtitle("Linear Mixed Model") + labs(x =

grid.arrange(p1, p2, p3, p4, p5, p6, p7, p8, p9, nrow = 3, top = textGrob("Posterior predictions with t
```

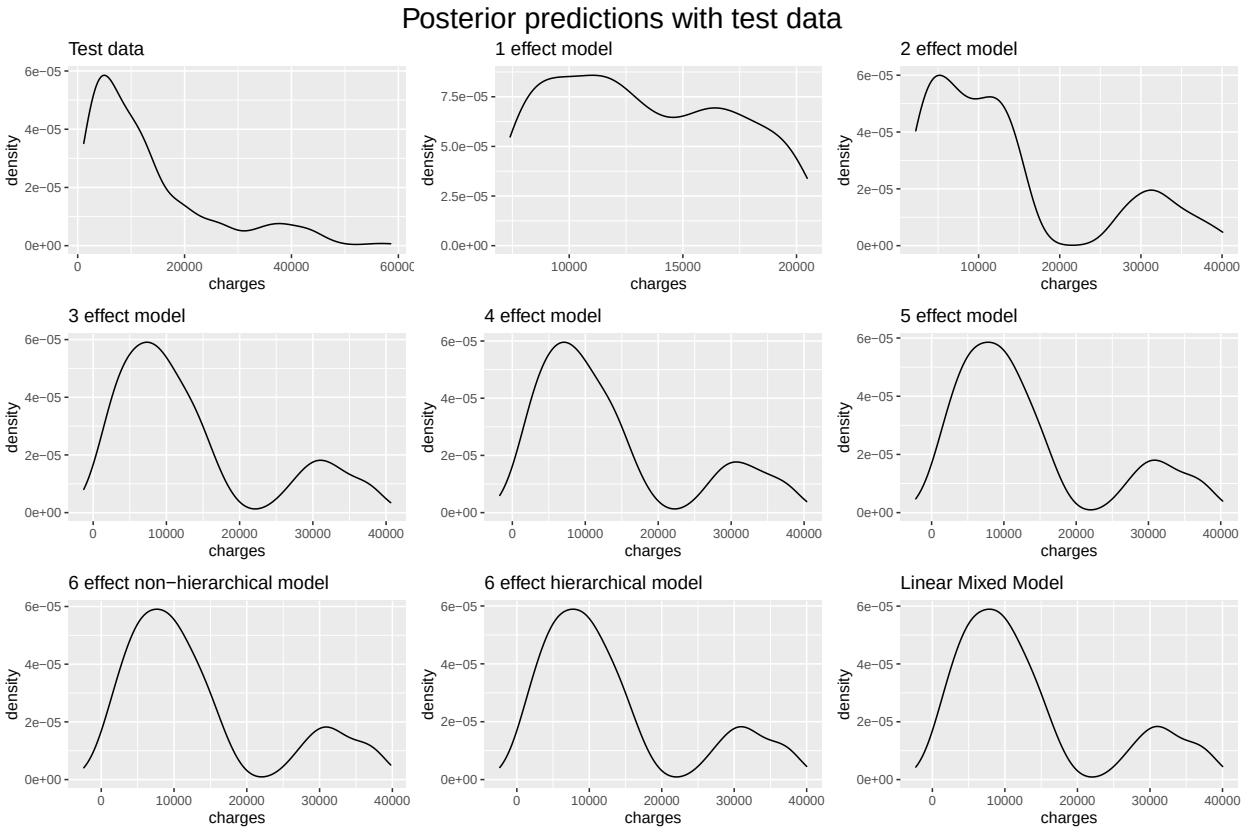


Figure 1: A nice image.