Bayesian meta-analysis in Stan with *baggr*

1 December 2022

baggr

baggr (Bayesian aggregator): Bayesian meta-analysis in R with Stan

Overall goals:

1. Implement all basic meta-analysis models and tools
2. Focus on accessibility, model criticism and comparison
3. Help people avoid basic mistakes
4. Keep the framework flexible and extend to more models

(Probably) *not* our goal:

5. Build a package for people who already build their models in Stan

baggr v0.7 beta is on CRAN and GitHub

Features

- Continuous and categorical data
- "Hand-holding" approach (don't assume users know what they're doing)
- Meta-analytic workflow (data preparation, p.p. checking, LOO CV)
- Allows summary and individual-level data
- Meta-regression
- Plain language prior definitions (like `rstanarm`)

Development

Next (?):

Driven largely by suggestions from users.

- Port to command-line Stan
- More flexible (formula-driven?) definition of REs
- Modeling more parameters (mixtures, variances, quantiles)
- Working with counts and (parameteric/Cox) survival
  - Survival models already programmed by RAs at UChicago
- BYO? (part of the wider workflow effortit)

Some health-specific applications for m-a:

- Dealing with small samples (e.g. chemical risk assessment)
- "Model-based" meta-analysis (PK/PD modeling drug development)
- Network meta-analysis / ITC (health technology assessment)

Example (8 schools!)

We use a familiar summary-level dataset from Rubin (1981).
Individual-level data works just as well.

```
library(baggr)
schools
```

```
##      group tau se
## 1 School A  28 15
## 2 School B   8 10
## 3 School C  -3 16
## 4 School D   7 11
## 5 School E  -1  9
## 6 School F   1 11
## 7 School G  18 10
## 8 School H  12 18
```

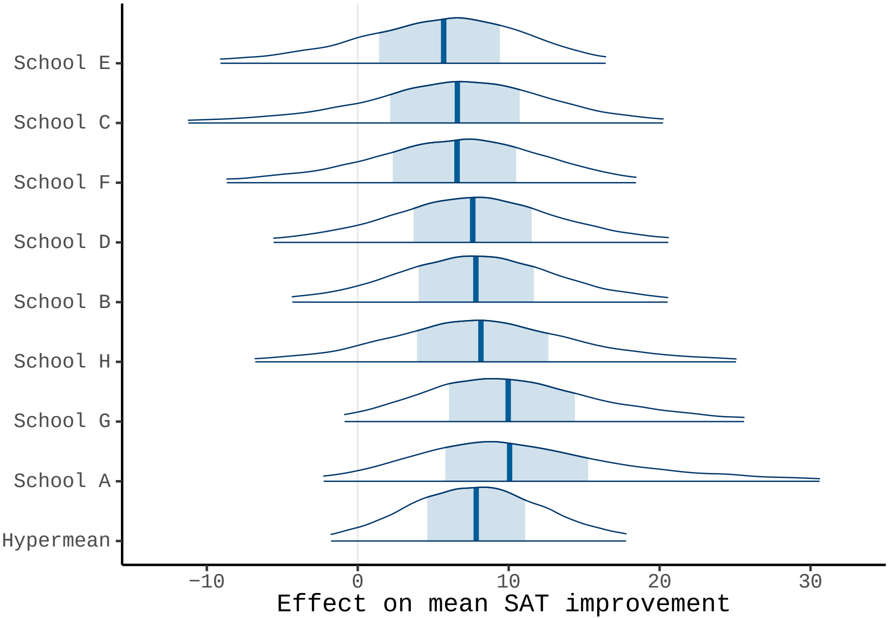
baggr syntax example: simple model workflow

```
# Fitting function is baggr()
fit <- baggr(schools, model = "rubin", pooling = "partial",
             prior_hypermean = normal(0, 25), prior_hypersd = uniform(0, 100))

# Semi-automatic prior choice, automatic model choice (with prompts):
eff_lab <- "mean SAT improvement"
fit_c <- baggr(schools, prior_hypersd = cauchy(0, 50), effect = eff_lab)
fit_n <- baggr(schools, prior_hypersd = normal(0, 10), effect = eff_lab)
fit_u <- baggr(schools, prior_hypersd = uniform(0, 100),
                        prior_hypermean = normal(-5, 10), effect = eff_lab)

# Plot results for one model, making use of bayesplot
baggr_theme_update(text = element_text(family = "mono"))
plot(fit_c, style = "areas", hyper = TRUE, order = TRUE)

# Compare fit to data
bgc <- baggr_compare("Cauchy" = fit_c, "Normal" = fit_n, "Uniform" = fit_u)
bgc; plot(bgc)
# Compare posterior predictive distribution (note ggplot)
effect_plot("Cauchy" = fit_c, "Normal" = fit_n, "Uniform" = fit_u) + xlim(-10, 30)

# Compare the impact of different prior choices
bgc_pooling <- baggr_compare(schools, prior_hypersd = normal(0, 10), what = "pooling")
plot(bgc_pooling)

# Use LOO cross-validation to choose between partial & full pooling
loo_p <- loocv(schools, pooling = "partial")
loo_f <- loocv(schools, pooling = "full")
loo_compare(loo_p, loo_f)
```

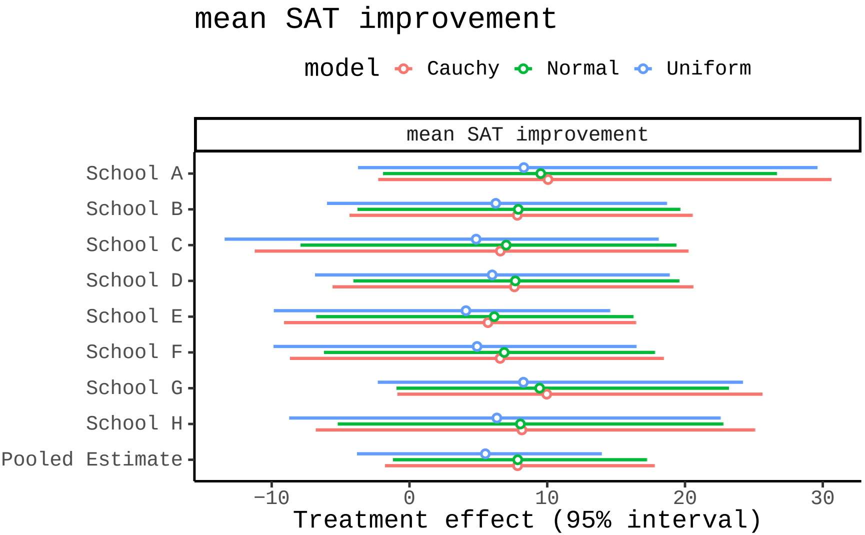
```
fit_c
```

```
## Model type: Rubin model with aggregate data
## Pooling of effects: partial
##
## Aggregate treatment effect (on mean SAT improvement):
## Hypermean (tau) =  7.9 with 95% interval -1.8 to 17.8
## Hyper-SD (sigma_tau) = 6.28 with 95% interval 0.22 to 19.58
## Total pooling (1 - I^2) = 0.8 with 95% interval 0.3 to 1.0
##
## Treatment effects on mean SAT improvement:
##          mean  sd   2.5%  50% 97.5% pooling
## School A 11.2 8.2  -2.26 10.1    31    0.83
## School B  7.9 6.2  -4.34  7.8    21    0.73
## School C  6.1 7.6 -11.22  6.6    20    0.85
## School D  7.6 6.4  -5.57  7.6    21    0.76
## School E  5.2 6.3  -9.09  5.7    16    0.70
## School F  6.2 6.7  -8.67  6.6    18    0.76
## School G 10.6 6.7  -0.87 10.0    26    0.73
## School H  8.4 7.7  -6.79  8.2    25    0.87
```

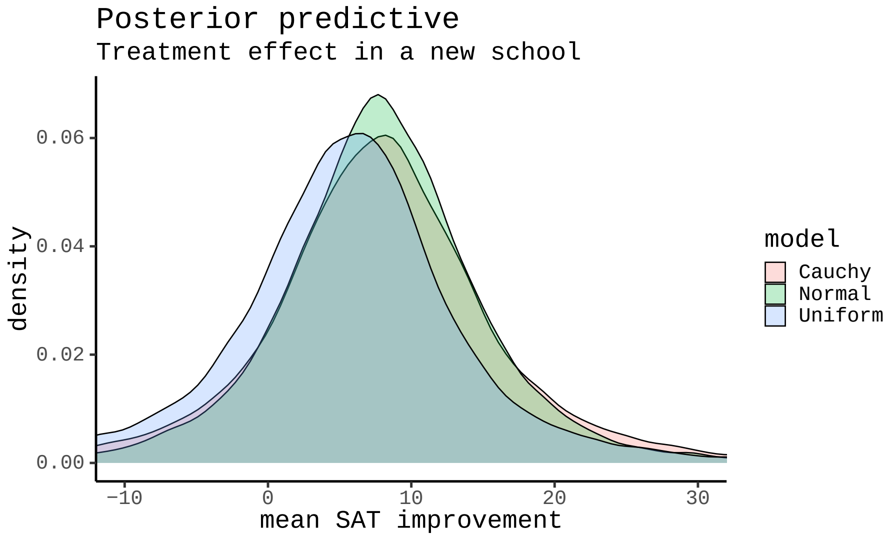
```r
plot(fit_c, style = "areas", hyper = TRUE, order = TRUE)
```



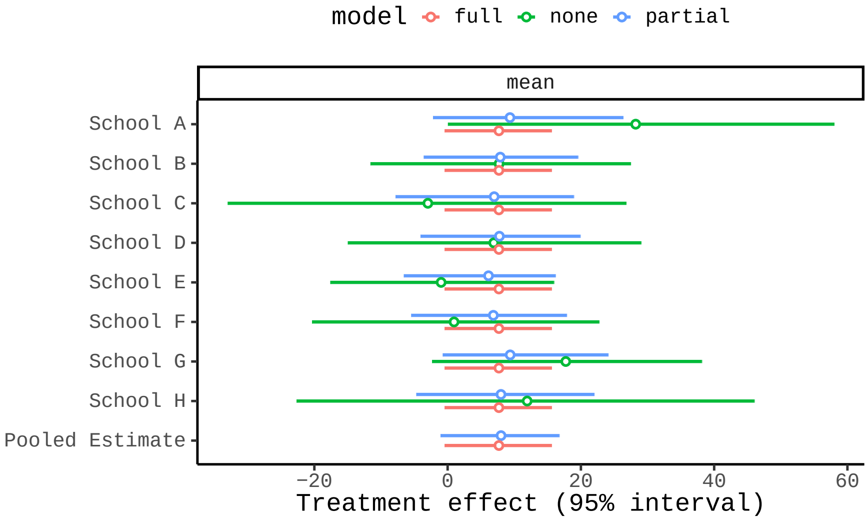
```r
bgc <- baggr_compare("Cauchy" = fit_c, "Normal" = fit_n, "Uniform" = fit_u)
bgc; plot(bgc)
```



```r
effect_plot("Cauchy" = fit_c, "Normal" = fit_n, "Uniform" = fit_u) +
  coord_cartesian(xlim = c(-10, 30)) +
  ggtitle("Posterior predictive", "Treatment effect in a new school")
```



```
bgc_pooling <- baggr_compare(schools, prior_hypersd = normal(0, 10),
                             what = "pooling")
plot(bgc_pooling)
```



```
loo_p <- loocv(schools, pooling = "partial")
loo_f <- loocv(schools, pooling = "full")
loo_compare(loo_p, loo_f)
```

```
## Comparison of cross-validation
##
##                      ELPD ELPD SE
## Model 1 - Model 2 -0.754    0.33
```

Additional commands:

```
treatment_effect(fit)
group_effects(fit)
effect_draw(fit)
forest_plot(fit)
pooling(fit)
```