Model comparison with Bayes factor

Bruno Nicenboim / Shravan Vasishth

2020-03-16

Model comparison

Model comparison using the Bayes factor

Comparison of two different models

Model comparison

There are two perspectives on model comparison:

- a (prior) predictive perspective based on the Bayes factor using marginal likelihoods
- a (posterior) predictive perspective based on cross-validation.

Model comparison using the Bayes factor

Marginal likelihood

Bayes' rule can be written with reference to a specific statistical model $\mathcal{M}_1$.

$$p(\theta \mid D, \mathcal{M}_1) = \frac{p(\theta \mid \mathcal{M}_1)p(D \mid \theta, \mathcal{M}_1)}{p(D \mid \mathcal{M}_1)} \tag{1}$$

Here D refers to the data and θ is a vector of parameters.

$P(D \mid \mathcal{M}_1)$ is the marginal likelihood, and is a single number that tells you the likelihood of the observed data D given the model $\mathcal{M}_1$

The likelihood is evaluated for every possible parameter value, weighted by the prior plausibility and summed together.

A simple example:

- Model 1

```
l1 <- function(p) dbinom(80, 100, p) * dbeta(p, 4, 2)
(ml1 <- integrate(l1, 0, 1)[[1]])
```

```
## [1] 0.02
```

A simple example:

- Model 2

```
l2 <- function(x, y) {
  dbbinom(80, 100, x, y) * dlnorm(x, 0, 100) *
    dlnorm(y, 0, 100)
}
(ml2 <- rmutil::int2(l2, a = c(0, 0), eps = 1e-04, max = 12))
```

```
## [1] 0.00000833
```

A simple example:

- Model 3

```
l3 <- function(p) dbinom(80, 100, p) * dbeta(p, 1, 1)
(ml3 <- integrate(l3, 0, 1)[[1]])
```

```
## [1] 0.0099
```

Bayes factor

BF is a measure of relative evidence, compares the predictive performance of two models, by means of a ratio of marginal likelihoods:

$$BF_{12} = \frac{P(D \mid \mathcal{M}_1)}{P(D \mid \mathcal{M}_2)} \tag{2}$$

- BF_{12} indicates the extent to which the data are more probable under $\mathcal{M}_1$ over $\mathcal{M}_2$, or
- which of the two models is more likely to have generated the data, or
- the relative evidence that we have for $\mathcal{M}_1$ over $\mathcal{M}_2$.

Bayes factor interpretation

BF_{12}	Interpretation
> 100	Extreme evidence for $\mathcal{M}_1$.
$30 - 100$	Very strong evidence for $\mathcal{M}_1$.
$10 - 30$	Strong evidence for $\mathcal{M}_1$.
$3 - 10$	Moderate evidence for $\mathcal{M}_1$.
$1 - 3$	Anecdotal evidence for $\mathcal{M}_1$.
1	No evidence.
$\frac{1}{1} - \frac{1}{3}$	Anecdotal evidence for $\mathcal{M}_2$.
$\frac{1}{3} - \frac{1}{10}$	Moderate evidence for $\mathcal{M}_2$.
$\frac{1}{10} - \frac{1}{30}$	Strong evidence for $\mathcal{M}_2$.
$\frac{1}{30} - \frac{1}{100}$	Very strong evidence for $\mathcal{M}_2$.
$< \frac{1}{100}$	Extreme evidence for $\mathcal{M}_2$.

In our previous example, we can calculate BF_{12}, BF_{13}, and BF_{23}. (Notice that BF_{21} is simply $\frac{1}{BF_{12}}$).

- $BF_{12} = ml1/ml2 = 2399.666$
- $BF_{13} = ml1/ml3 = 2.018$
- $BF_{23} = ml2/ml3 = 0.001 = \frac{1}{BF_{32}} = \frac{1}{1189.007}$

Probability of a model

If we want to know how much more probable model $\mathcal{M}_1$ than $\mathcal{M}_2$ is given the data, D, we need the prior odds, how much probable $\mathcal{M}_1$ is than $\mathcal{M}_2$ *a priori*.

$$\frac{p(\mathcal{M}_1 \mid D)}{p(\mathcal{M}_2 \mid D)} = \frac{p(\mathcal{M}_1)}{p(\mathcal{M}_2)} \times \frac{P(D \mid \mathcal{M}_1)}{P(D \mid \mathcal{M}_2)} \tag{3}$$

$$\text{Posterior odds}_{12} = \text{Prior odds}_{12} \times BF_{12} \tag{4}$$

The Bayes factor **only** tells us how much we need to update our relative belief between the two models.

Example: Null hypothesis testing the N400 effect

While we have previously estimated the effect of cloze probability on the N400, estimation cannot really answer a very popular question: *How much evidence we have in support for the effect of cloze probability?*

We are going to answer this question with the Bayes factor, by doing model comparison: We'll compare a model that assumes a *certain* effect, with a null model that assumes no effect.

The prior on β will be **crucial** for the calculation of the Bayes factor.

1. I generally want to be agnostic regarding the direction of the effect: I will center the prior of β on zero.
2. I would need to know a bit about the variation on the DV that I'm analyzing. I would say that for N400 averages, the standard deviation of the signal is between 8-15 microvolts.
3. Effects in psycholinguistics are rather small, representing between 5%-30% of the SD of the DV.
4. I know that the effect of noun predictability on the N400 is one the most reliable and strongest effects in neurolinguistics, and β represents the change in average voltage when we move from a cloze probability of zero to one –the strongest prediction effect.

We will start then with $\beta \sim Normal(0, 5)$ (since 5 microV is 30% of 15).

We are going to “smooth” the Cloze probability in this example:

```
eeg_data <- read_tsv("data/public_noun_data.txt") %>%
    filter(lab=="edin") %>%
    mutate(nans = round(cloze/100 *20),
           scloze = (nans + 1) / 22,
           cscloze = scloze - mean(scloze))
```

```r
m_N400_h_linear <- brm(n400 ~ cscloze +
                         (cscloze | subject) +
                         (cscloze | item),
                       prior = c(prior(normal(2, 5), class = Intercept),
                                 prior(normal(0, 5), class = b),
                                 prior(normal(10, 5), class = sigma),
                                 # taus in our model
                                 prior(normal(0, 2), class = sd),
                                 prior(lkj(4), class =cor)),
                       warmup = 2000,
                       iter = 20000,
                       control = list(adapt_delta = 0.9),
                       save_all_pars = TRUE,
                       data = eeg_data)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: n400 ~ cscloze + (cscloze | subject) + (cscloze | item)
##    Data: eeg_data (Number of observations: 2827)
## Samples: 4 chains, each with iter = 20000; warmup = 2000; thin = 1;
##          total post-warmup samples = 72000
##
## Group-Level Effects:
## ~item (Number of levels: 80)
##                         Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)               1.51      0.34     0.82     2.16 1.00    27153    35328
## sd(cscloze)                 1.91      1.02     0.12     3.88 1.00    21533    28485
## cor(Intercept,cscloze)     -0.26      0.29    -0.74     0.38 1.00    62181    53276
##
## ~subject (Number of levels: 37)
##                         Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)               2.16      0.35     1.54     2.91 1.00    30933    46745
## sd(cscloze)                 1.26      0.81     0.06     3.00 1.00    28600    40319
## cor(Intercept,cscloze)      0.08      0.30    -0.53     0.64 1.00   106208    53749
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept     3.64      0.45     2.75     4.53 1.00    47343    49162
## cscloze       2.53      0.70     1.14     3.88 1.00    89936    54142
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma    11.51      0.16    11.21    11.83 1.00   101098    52495
##
## Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
```

And we'll run our model without the parameter of interest, the null model:

```
m_N400_h_null <- brm(n400 ~ 1 +
                       (cscloze | subject) +
                       (cscloze | item),
                     prior = c(prior(normal(2, 5), class = Intercept),
                               prior(normal(10, 5), class = sigma),
                               ## taus in our model
                               prior(normal(0, 2), class = sd),
                               prior(lkj(4), class =cor)),
                     warmup = 2000,
                     iter = 20000,
                     control = list(adapt_delta = 0.9),
                     save_all_pars = TRUE,
                     data = eeg_data)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: n400 ~ 1 + (cscloze | subject) + (cscloze | item)
##    Data: eeg_data (Number of observations: 2827)
## Samples: 4 chains, each with iter = 20000; warmup = 2000; thin = 1;
##          total post-warmup samples = 72000
##
## Group-Level Effects:
## ~item (Number of levels: 80)
##                         Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)               1.42      0.35     0.71     2.08 1.00    23430    25594
## sd(cscloze)                 2.92      1.02     0.60     4.76 1.00    18182    18089
## cor(Intercept,cscloze)     -0.34      0.25    -0.76     0.22 1.00    45534    47787
##
## ~subject (Number of levels: 37)
##                         Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)               2.15      0.35     1.54     2.91 1.00    33189    49814
## sd(cscloze)                 1.80      0.97     0.12     3.70 1.00    20586    29783
## cor(Intercept,cscloze)      0.09      0.28    -0.48     0.62 1.00    87320    56711
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept     3.69      0.47     2.76     4.61 1.00    45035    49964
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma    11.51      0.16    11.20    11.83 1.00   114631    52069
##
## Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
```

Now we are ready to compute log marginal likelihood via bridge sampling for both models:

```
lml_linear <- bridge_sampler(m_N400_h_linear, silent = TRUE)
lml_null <- bridge_sampler(m_N400_h_null, silent = TRUE)
```

The `bayes_factor` is a convenient function to calculate the Bayes factor.

```
(BF_ln <- bayes_factor(lml_linear, lml_null))
```

```
## Estimated Bayes factor in favor of x1 over x2: 54.15370
```

But it can be done like this as well:

```
BF_ln <- exp(lml_linear$logml- lml_null$logml).
```

About choosing good priors

But what happens if we are have no clue about a good prior for β?

- We might be comparing the null model with a very “bad” alternative model. See Uri Simonsohn’s criticism of Bayes factors https://datacolada.org/78a).

About choosing good priors

How to overcome this?

- learn about the effect size that we are investigating by first running an exploratory analysis without Bayes factor, and use the information of the first experiment to calibrate the priors for the next confirmatory experiment. See Verhagen and Wagenmakers (2014) for a Bayes Factor test calibrated to investigate replication success.
- Examine all (or a lot of) the possible alternative models, using a sensitivity analysis; recall that the model is the likelihood *and* the priors.

Bayes factor for several models

(This will take a very long time)

```
prior_sd <- c(1, 1.5, 2,2.5, 5, 8, 10, 20, 40, 50)
BFs <- map_dfr(prior_sd, function(psd) {
    gc() # force R "garbage collector" so that we don't run out of memory
    fit <- brm(n400 ~ cscloze +
                   (cscloze | subject) +
                   (cscloze | item),
               prior =
                 c(prior(normal(2, 5), class = Intercept),
                   set_prior(paste0("normal(0,",psd ,")"),
                             class = "b"),
                   prior(normal(10, 5), class = sigma),
                   ## taus in our model
                   prior(normal(0, 2), class = sd),
                   prior(lkj(4), class =cor)),
               warmup = 2000,
               iter = 20000,
               control = list(adapt_delta = 0.9),
               save_all_pars = TRUE,
               data = eeg_data)
    lml_linear_beta <- bridge_sampler(fit, silent = TRUE)
    tibble(beta_sd = psd, BF = bayes_factor(lml_linear_beta, lml_null)$bf)
})
```

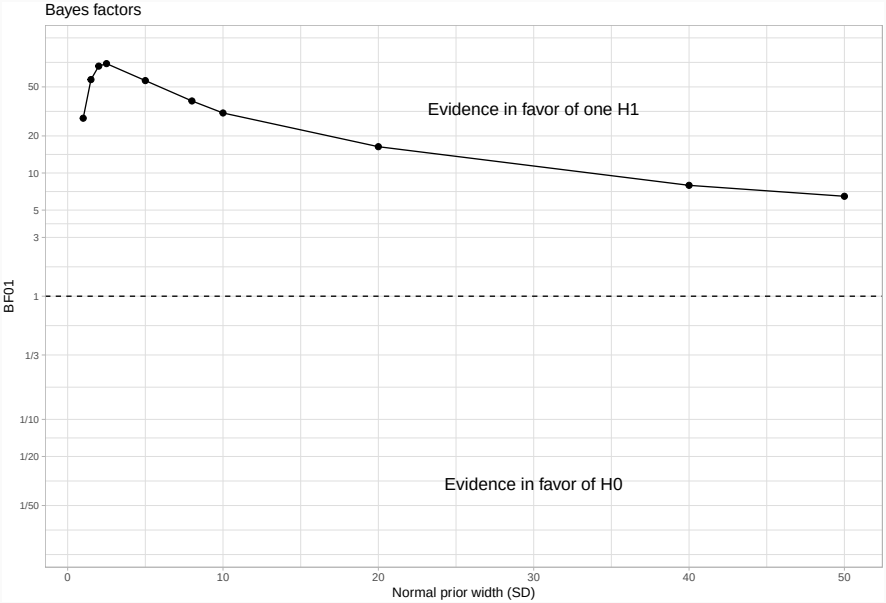

Figure 1: Prior sensitivity analysis for the Bayes factor

Comparison of two different models

Example: Two different models of the N400 effect

It has been argued that the effect of predictability is logarithmic, we might ask ourselves if this is also valid for the N400 effect, and thus how much evidence we have for a logarithmic effect vs a linear effect.

```
eeg_data <- eeg_data %>%
  mutate(clogscloze = log(scloze) - mean(log(scloze)))
```

One new problem that arises is that we need to assign equivalent priors to both β in the models because they are interpreted differently, and we want to put both models on equal footing.

- When there is a linear relationship, β represents the rate of change in the N400 average when we compare words with 0 to 1 Cloze probability,
- When there is logarithmic relationship, β represents a non-linear effect: the rate of change in the average N400 when we compare words with $exp(-1) = .36..$ probability to $exp(0) = 1$, or $exp(-2) = .1353$ probability to $exp(-1) = .36...$

One possible solution is to force them to have the same SD:

```
eeg_data <- eeg_data %>%
  mutate(clogscloze = c(scale(log(scloze)) * sd(cscloze)))
```

```
m_N400_h_log <- brm(n400 ~ clogscloze +
  (clogscloze | subject) +
  (clogscloze | item),
prior =
  c(
    prior(normal(2, 5), class = Intercept),
    prior(normal(0, 5), class = b),
    prior(normal(10, 5), class = sigma),
    # taus in our model
    prior(normal(0, 2), class = sd),
    prior(lkj(4), class = cor)
  ),
warmup = 2000,
iter = 20000,
control = list(adapt_delta = 0.9),
save_all_pars = TRUE,
data = eeg_data
)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: n400 ~ clogscloze + (clogscloze | subject) + (clogscloze | item)
##    Data: eeg_data (Number of observations: 2827)
## Samples: 4 chains, each with iter = 20000; warmup = 2000; thin = 1;
##          total post-warmup samples = 72000
##
## Group-Level Effects:
## ~item (Number of levels: 80)
##                           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)                 1.52      0.34     0.82     2.17 1.00    23552    23901
## sd(clogscloze)                1.40      0.88     0.07     3.25 1.00    26927    35507
## cor(Intercept,clogscloze)    -0.15      0.31    -0.70     0.49 1.00    80968    54969
##
## ~subject (Number of levels: 37)
##                           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)                 2.15      0.35     1.53     2.90 1.00    30355    47870
## sd(clogscloze)                1.27      0.82     0.06     3.03 1.00    26687    34820
## cor(Intercept,clogscloze)     0.04      0.30    -0.55     0.61 1.00   102306    51214
##
## Population-Level Effects:
##             Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept       3.64      0.45     2.75     4.53 1.00    44071    50269
## clogscloze      2.86      0.68     1.50     4.19 1.00    95123    55856
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma    11.52      0.16    11.21    11.83 1.00   101112    54008
##
## Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
```

We calculate the log-marginal likelihood

```
lml_log <- bridge_sampler(m_N400_h_log, silent = TRUE)
```

And we can compare the models now:

```
(BF <- bayes_factor(lml_linear, lml_log))
```

```
## Estimated Bayes factor in favor of x1 over x2: 0.19872
```

We can interpret this more easily as the model with the log Cloze probability being (1/BF) 5 more likely than the model with linear Cloze probability.

Summary

- While in reasonably large samples, the posterior distribution is not overly influenced by weakly informative priors, the Bayes factor *is*.
- When priors are defined to allow a broad range of values, the result will be a lower marginal likelihood (which in turns influences the Bayes factor, as we saw in the examples above).
- The calculation of the Bayes factor depends on answering a question about which there may be disagreement among researchers: “What way of assigning probability distributions of effect sizes as predicted by theories would be accepted by protagonists on all sides of a debate?” (Dienes 2011)
- One of advantage of the Bayes Factor is that once the minimal magnitude of an expected effect is agreed upon, evidence can be gathered in favor of the null hypothesis.

Further readings

- Fabian Dablander's blog post https://fabiandablander.com/r/Law-of-Practice.html for a comparison between Bayes factor and leave-one-out (loo) cross validation
- For a Bayes Factor Test calibrated to investigate replication success, see Verhagen and Wagenmakers (2014).
- Chapter 7 of Gelman et al. (2014)
- For a discussion about the advantages and disadvantages of (leave-one-out) cross-validation, see Gronau and Wagenmakers (2018), Vehtari et al. (2019) and Gronau and Wagenmakers (n.d.).

- Interesting read about when cross-validation can be applied: https://statmodeling.stat.columbia.edu/2018/08/03/loo-cross-validation-approaches-valid/
- Against null hypothesis testing with BF: https://statmodeling.stat.columbia.edu/2019/09/10/i-hate-bayes-factors-when-theyre-used-for-null-hypothesis-significance-testing/
- In favor of null hypothesis testing with BF as an approximation (but assuming realistic effects): https://statmodeling.stat.columbia.edu/2018/03/10/incorporating-bayes-factor-understanding-scientific-information-replication-crisis/

References

Gelman, Andrew, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. 2014. *Bayesian Data Analysis*. Third. Boca Raton, FL: Chapman; Hall/CRC.

Gronau, Quentin F., and Eric-Jan Wagenmakers. 2018. "Limitations of Bayesian Leave-One-Out Cross-Validation for Model Selection." *Computational Brain & Behavior*, September. https://doi.org/10.1007/s42113-018-0011-7.

Gronau, Quentin F, and Eric-Jan Wagenmakers. n.d. "Rejoinder: More Limitations of Bayesian Leave-One-Out Cross-Validation," 25.

Vehtari, Aki, Daniel P. Simpson, Yuling Yao, and Andrew Gelman. 2019. "Limitations of 'Limitations of Bayesian Leave-One-Out Cross-Validation