General information about GAMs

Pablo E. Gutiérrez-Fonseca

3/18/2022

Model construction

s (Both GAMs)

s = represent smooth function

```
gam(value ~s(date,...
```

Define knots (Both GAMs)

k = knots. 12 month per year or 24 sampling event per year.
Seleccione 12 por Simpson, del siguiente enlace:
https://fromthebottomoftheheap.net/2014/05/09/modelling-seasonal-data-with-gam/

bs= basis spline (Both GAMs)

bs= basis spline.
Smooth classes are invoked directly by s terms.
https://stat.ethz.ch/R-manual/R-devel/library/mgcv/html/smooth.terms.html

```
brms::brm(bf(value ~ s(date, bs="cs"...))
```

Cubic regression splines & A cyclic cubic regression spline bs="cr". These have a cubic spline basis defined by a modest sized set of knots spread evenly through the covariate values.
bs="cs" specifies a shrinkage version of "cr".
bs="cc" specifies a cyclic cubic regression splines. i.e. a penalized cubic regression splines whose ends match, up to second derivative.

P-splines
bs="ps".

bf (Bayesian GAMs) Note that we use the bf() argument to specify this nonlinear model.

Output

Hay que ver el Smooth Terms: -> sds(sdate_1) -> sds(stimes_1) is the variance parameter, which has the effect of controlling the wiggliness of the smooth - the larger this value the more wiggly the smooth. https://fromthebottomoftheheap.net/2018/04/21/fitting-gams-with-brms/

Check models

pp_check()

The pp_check allows for graphical posterior predictive checking. We can generate figures to compare the observed data to simulated data from the posterior predictive distribution. This is a great graphical way to evaluate your model.
https://tem11010.github.io/regression_brms/

Here, nsamples refers to the number of draws from the posterior distribution to use to calculate yrep values.

```
pp_check(model, nsamples=100)
```

bayes_r2

Bayes R2 quantifies the expected fit or variance explained by a model

We can also get an R-squared estimate for our model, thanks to a newly-developed method from Andrew Gelman, Ben Goodrich, Jonah Gabry and Imad Ali, with an explanation here:
http://www.stat.columbia.edu/~gelman/research/unpublished/bayes_R2.pdf https://tem11010.github.io/regression_brms/

r2(cc.qp_A.Bayes_mod) Existe esta otra, pero usare la de Gelman

```
bayes_R2(model)
```

loo_compare()

The loo_compare() output rank orders the models such that the best fitting model appears on top. All models receive a difference score relative to the best model.
https://bookdown.org/ajkurz/DBDA_recoded/model-comparison-and-hierarchical-modeling.html

Note that the best model is always on top, and the comparison is already on the difference score. Following a two standard error heuristic (but see here), since the difference in the elpd scores is more than twice its standard error, we would choose the power law model as the better model. But wait - what do these warnings mean? Let's look at the output of the loo function for the exponential law (suppressing the warning):
https://www.r-bloggers.com/2019/05/bayesian-modeling-using-stan-a-case-study/

brms::model_weights()

I don't know that I'd call these weights probabilities, but they do sum to one.
https://bookdown.org/ajkurz/DBDA_recoded/model-comparison-and-hierarchical-modeling.html

AIC

AIC and DIC are not recommended in particular not with Bayesian models. So it is not necessarily surprising that the results differ (buerkner).

https://discourse.mc-stan.org/t/waic-aicc-and-dic-differences-in-a-bernoulli-glmm-in-brms-vs-glmmtmb/5966

Gelman, A., Hwang, J., & Vehtari, A. (2014). Understanding predictive information criteria for Bayesian models. Statistics and computing, 24 (6), 997-1016.

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. Statistics and Computing , 27 (5), 1413-1432.