Bayesian Compromise Modeling

Bayesian Statistics and Data Analysis, 2ST128

Valentin Zulj

Department of Statistics

1. Introduction
2. Bayesian Model Averaging
3. Bayesian Stacking
4. Examples – Code and Research

1 Introduction

2 Bayesian Model Averaging

3 Bayesian Stacking

4 Examples – Code and Research

Introduction

Assume access to a vector of outcomes, $\boldsymbol{y} = (y_1, \dots, y_n)^T$, and a matrix of covariates, $\boldsymbol{X}$, with which to build a prediction model. Traditionally, statisticians proceed by

1. Selecting (selection criteria, scientific theory, etc) what they believe to be the best model,
2. Applying the model as if it was the truth or, the generally the best.

However, it is quite easy to criticise the model selection step. For example,

- Model selection can be unstable – Can we trust it?
- Very different models can have similar performance in one sample – How do we discriminate?

Of course, these issues have consequences, including

- Poor generalizability and out-of-sample performance,
- Sub-optimal selection due to ad-hoc decision making,
- Overly optimistic inference, due to misrepresentation of model selection uncertainty.

Compromise modeling tries to deal with these issues, epsecially poor generalizability.

Compromise Modeling – General Idea

Suppose that φ is some quantity of interest. Moreover, assume that

- There are K candidate models under consideration, and that
- Each candidate model produces φ_k as an approximation of φ.

Then, compromise modeling relies on a weighted average of the candidates,

$$\bar{\varphi} = \sum_{k=1}^{K} w_k \varphi_k,$$

as the final approximation of φ. Here w_k are model specific weights that can be estimated to suit the purpose of the analysis.

Example: Variable Selection in Linear Regression

Problem: Which subset $\boldsymbol{X}_k \subset \boldsymbol{X}$ of variables gives the best regression model?

Common solution: Use some off-the-shelf procedure to chose "optimal" subset.

Alternative solution: Consider several different subsets, and combine outputs using a weighted average. For example, if

- φ is the posterior predictive distribution, then $\bar{\varphi} = \sum_k w_k \cdot p(\tilde{\boldsymbol{y}}|\boldsymbol{y}, \boldsymbol{X}_k)$, and if
- φ is the posterior predictive mean, then $\bar{\varphi} = \sum_k w_k \cdot E[\tilde{\boldsymbol{y}}|\boldsymbol{y}, \boldsymbol{X}_k]$.

Note: both $p(\tilde{\boldsymbol{y}}|\boldsymbol{y}, \boldsymbol{X}_k)$ and $E[\tilde{\boldsymbol{y}}|\boldsymbol{y}, \boldsymbol{X}_k]$ are derived as usual, and require no new theory.

However, the weights $w_1, \ldots, w_K$, need to be estimated. In Bayesian theory, this usually means employing either Bayesian model averaging (BMA) or Bayesian stacking (BS).

1 Introduction

2 Bayesian Model Averaging

3 Bayesian Stacking

4 Examples – Code and Research

The Bayesian Model Average

BMA is concerned with $\varphi = p(\Delta|D)$ (i.e. the posterior distribution of Δ), where

- Δ denotes a particular quantity of interest (e.g. $\tilde{\boldsymbol{y}}$ or regression coefficients or whatever), and
- D denotes the data.

The BMA posterior probability for Δ is

$$p(\Delta|D) = \sum_{k=1}^{K} p(\Delta|M_k, D) \cdot p(M_k|D),$$

where M_k denotes the k:th candidate model (i.e. regression fitted using the subset $\boldsymbol{X}_k$). Here,

- $p(\Delta|M_k, D)$ is the posterior probability for Δ given model k, adn
- $p(M_k|D)$ is the posterior probability of model M_k itself.

It is clear that $p(\Delta|D)$ is a compromise posterior that assigns $w_k = p(M_k|D)$ to model M_k.

Problems in Putting BMA to Use

- Posteriors can be difficult to derive in advanced models. For BMA, this amounts to
  - The usual difficulty of finding $p(\Delta|M_k, D)$, and
  - The additional step of finding $p(M_k|D)$ (priors and computations).
- Some simple problems have analytical solutions (e.g. Raftery et al., 1997 on linear regression).
- MCMC Model Composition (MCMCMC) simplifies things by generating a Markov chain that moves through the model space (I think it can be applied using the `BMA` package).
- Another couple of issue have to do with the set of candidate models. To discuss them, some further notation and context have to be introduced.

$\mathcal{M}$-open and $\mathcal{M}$-closed

Let $\mathcal{M} = \{M_1, \ldots, M_K\}$ denote the set of candidate models. Then, in very simplified terms,

- $\mathcal{M}$-closed represents the case where one $M_k \in \mathcal{M}$ is believed to correctly specify the DGP, although there is no knowledge of which candidate it is.
- $\mathcal{M}$-complete represents the case where the DGP can be conceptualized, but cannot be included in $\mathcal{M}$ to due to, for example, complexity or practical feasibility.
- $\mathcal{M}$-open represents the case where the data generating mechanism cannot be conceptualized, and because of that cannot be included in $\mathcal{M}$.

$\mathcal{M}$-complete/open feel like the most realistic options (agree?). What does this mean for BMA?

Problems in Putting BMA to Use, ctd.

1. As $n \to \infty$, BMA assigns weight 1 to the M_k closest to the DGP in terms of KL divergence. So,
   - For $\mathcal{M}$-closed, this is great since BMA will chose the "true model".
   - For $\mathcal{M}$-complete/open, will select a single model that may perform worse than an average.
2. The prior probability of M_k represent our belief that M_k represents the data generating distribution. How do we specify priors if we believe no M_k is correct?

Bayesian stacking tries to address these issues.

1 Introduction

2 Bayesian Model Averaging

3 Bayesian Stacking

4 Examples – Code and Research

Bayesian Stacking – Definition of Problem

Let

- $\boldsymbol{w} = (w_1, \dots, w_K)^T \in \mathcal{W}$, be a weight vector, and
- $S(P, Q)$ be a scoring rule, measuring the similarity of two probabilistic forecasts P and Q.

Then, the Bayesian stacking weights are given by

$$\boldsymbol{w}^* = \underset{\boldsymbol{w} \in \mathcal{W}}{\operatorname{argmax}} \; S\left(\sum_{k=1}^{K} w_k p(\tilde{\boldsymbol{y}} | \boldsymbol{y}, M_k), p_{\text{true}}(\tilde{\boldsymbol{y}} | \boldsymbol{y}) \right).$$

That is, $\boldsymbol{w}^*$ maximizes the similarity of the stacked posterior predictive to the true distribution. Here, there is no need for priors on M_k, but the solution $\boldsymbol{w}^*$ needs to be found.

Bayesian Stacking - Cross-validation Approach

Problem: $p_{\text{true}}(\tilde{\boldsymbol{y}}|\boldsymbol{y})$ is unknown.

Solution (suggestion): use leave-one-out CV to approximate optimization problem empirically, and estimate the weights as

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w} \in \mathcal{W}}{\operatorname{argmax}} \; S\left(\sum_{k=1}^{K} w_k p(y_i | \boldsymbol{y}_{-i}, M_k), y_i \right).$$

This approach has been discussed and evaluated in the literature. For example,

- Clyde & Iversen (2013) formulate it in the Bayesian setting,
- Le & Clarke (2017) theoretically motivate use of CV for weight estimation,
- Yao et al. (2018) implement and evaluate the use of general scoring rules, and
- I write about it in my PhD thesis.

Scoring Rules

The choice of S determines the type of objects that the compromise can consider. For example, it is common to use one of

- The *energy* score, $ES(P, y) = \frac{1}{2}E_P\|Y - Y'\|^\beta - \mathbb{E}_P\|Y - y\|^\beta$. Here, $Y, Y' \sim P$ are independent,
- The log score, $LS(P, y) = \log[p(y)]$.

To evaluate the stacked prediction, let $P = \sum_k w_k p(\tilde{\boldsymbol{y}}|\boldsymbol{y}, M_k) := P_{\boldsymbol{w}}$. Then,

- For ES, setting $\beta = 2$ gives $ES(P_{\boldsymbol{w}}, y) = -\|\mathbb{E}_{P_{\boldsymbol{w}}}[Y] - y\|^2$. Thus, the resulting $\boldsymbol{w}^*$ is good for combining posterior predictive *means*, i.e. point predictions.
- For the log scoring rule, $\boldsymbol{w}^*$ maximizes $\log(P_{\boldsymbol{w}})$, the log of the stacked posterior predictive density. Thus, the resulting $\boldsymbol{w}^*$ is good for stacking posterior predictive *distributions*.

1 Introduction

2 Bayesian Model Averaging

3 Bayesian Stacking

4 Examples – Code and Research

Example: Using `rstanarm` and `loo`

`loo::stacking_weights` gives log-score weights. See stylized example below.

```
library("rstanarm"); library("loo")

# Fitting the candidate models
cand1 <- stan_glm(y ~ X1, data = df)
cand2 <- stan_glm(y ~ X1 + X2, data = df)
cand2 <- stan_glm(y ~ X1 + X2 + X3, data = df)

# LOO-CV approximation
loo1 <- loo(cand1); loo2 <- loo(cand2); loo3 <- loo(cand3)

# Pointwise LOO ELPD
lpd_point <- cbind(loo1$pointwise[,"elpd_loo"],
                   loo2$pointwise[,"elpd_loo"],
                   loo3$pointwise[,"elpd_loo"])

stacking_weights(lpd_point) # Estimates the weights
```

An example using real data is given in the enclosed `R` script.

Example: Frequentist Oracle Properties of Bayesian Stacking

Ongoing work aims to evaluate Bayesian stacking using frequentist asymptotics. In particular, the focus is to establish the *oracle property*

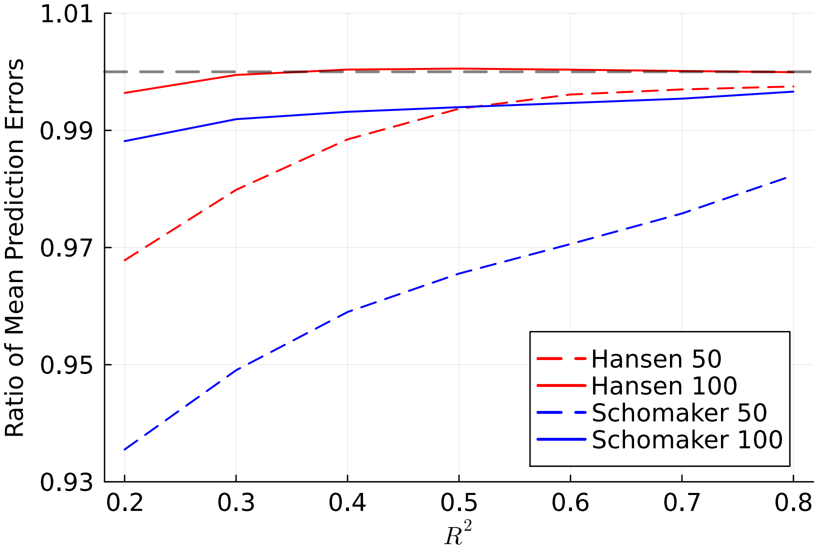
$$\frac{\|\boldsymbol{y} - \sum_k w_k^* E[\boldsymbol{y}|M_k]\|^2}{\inf_{\boldsymbol{w}\in\mathcal{W}} \|\boldsymbol{y} - \sum_k w_k E[\boldsymbol{y}|M_k]\|^2} \xrightarrow{p} 1.$$

This means that no candidate, nor any other average using the same candidates, and weights from the same $\mathcal{W}$, provides smaller asymptotic error than Bayesian stacking.

To date, the oracle has been established for Bayesian stacking of linear regression models using $\mathcal{N}(\mathbf{0}, \boldsymbol{S})$, as the prior for β, where $\boldsymbol{S} > 0$ is symmetric.

The results of a simulation are given on the next slide. The curve gives the ratio of the Bayesian stacking squared error to the squared error of the best single candidate.

Example: Frequentist Oracle Properties of Bayesian Stacking



References

Clyde, M., & Iversen, E. S. (2013). Bayesian Model Averaging in the $\mathcal{M}$-open Framework. In *Bayesian theory and applications*. Oxford University Press.

Le, T., & Clarke, B. (2017). A Bayes Interpretation of Stacking for $\mathcal{M}$-Complete and $\mathcal{M}$-Open Settings. *Bayesian Analysis*, *12*(3), 807–829.

Raftery, A. E., Madigan, D., & Hoeting, J. A. (1997). Bayesian Model Averaging for Linear Regression Models. *Journal of the American Statistical Association*, *92*(437), 179–191.

Yao, Y., Vehtari, A., Simpson, D., & Gelman, A. (2018). Using Stacking to Average Bayesian Predictive Distributions (with Discussion). *Bayesian Analysis*, *13*(3), 917–1003.