Bayesian methods for inference with large datasets

Luis Baroja[1] Joachim Vandekerckhove [1]

[1]University of California, Irvine

Abstract

Divide-and-conquer methods approximate posterior distributions using datasets that are too big to be handled by a single computer. In this work we present a review of the key ideas behind them along with an application using a model of choice behavior.

Our results indicate these methods offer a reasonable approximation to the target posterior in most cases and document some instances in which they fail to do so.

The inferred parameters of the choice model suggest interesting trends in NFL decisions regarding passing or rushing at each play during the last 10 seasons.

Divide-and-Conquer Methods

Divide-and-Conquer methods split the full data set $\mathcal{D}$ into K disjoint subsets $\mathcal{D}_1, \ldots, \mathcal{D}_K$ and factorize the target posterior over the parameter space θ assuming exchangeability between data subsets[1]:

$$p(\theta \mid \mathcal{D}) \propto p(\theta) \cdot \prod_{k=1}^{K} p(\mathcal{D}_k \mid \theta) \qquad (1)$$

$$\propto \prod_{k=1}^{K} p(\theta)^{1/K} \cdot p(\mathcal{D}_k \mid \theta) \qquad (2)$$

Each shard $\mathcal{D}_k$ is then assigned to a different computer, which generates MCMC samples from the corresponding subposterior:

$$s(\theta|\mathcal{D}_k) = p(\theta)^{1/K} \cdot p(\mathcal{D}_k \mid \theta)$$

The MCMC samples from all subposteriors are then mixed back together according to some recombination strategy in order to inform the full target posterior $p(\theta \mid \mathcal{D})$.

Recombination Strategies

There are several ways to recombine the MCMC samples from all data shards. In this work we focus on one of the most simple and computationally straightforward to implement.

Consensus Monte-Carlo (CMC)

Having obtained H samples $\{\theta_{k,h}\}_{h=1}^{H}$ in each k subposterior, the CMC algorithm generates H samples from the full posterior $\{\hat{\theta}_h\}_{h=1}^{H}$ according to

$$\hat{\theta}_h = \sum_{k=1}^{K} W_k \theta_{k,h}. \qquad (3)$$

The weight W_k of each subposterior is given by

$$W_k = \left(\Sigma_0^{-1} + \sum_{k=1}^{K} \Sigma_k^{-1}\right)^{-1} \left(\Sigma_0^{-1}/K + \Sigma_k^{-1}\right), \qquad (4)$$

where Σ_0 is the prior variance over θ, and Σ_k is the posterior variance in subposterior k, estimable from the corresponding MCMC samples[2].

A Beta-Binomial Toy Example

To illustrate the workings of divide-and-conquer methods we generated a sequence of 1000 zeros and ones from a simple Bernoulli model. Using an arbitrary prior $p(\theta) \sim \text{Beta}(2, 20)$ we calculated the closed-form posterior distribution.
Furthermore, we divided the sequence into 10 shards and obtained MCMC samples from each of the corresponding subposteriors. This step was repeated twice:

- Ensuring the shards were homogeneous by splitting the full sequence of 0s and 1s randomly.
- Intentionally generating heterogeneous shards with large differences in their distribution of 0s and 1s, thus violating exchangeability between shards.

Figure 1 presents the results of the CMC recombination in each data partition. The upper panel demonstrates that CMC closely approaches the target posterior when the shards are homogeneous. The lower panel, however, illustrates the drastic consequences of working with non-exchangeable shards even if they represent *exactly* the same data when added back together: the CMC result in the lower panel does not match the theoretical solution.

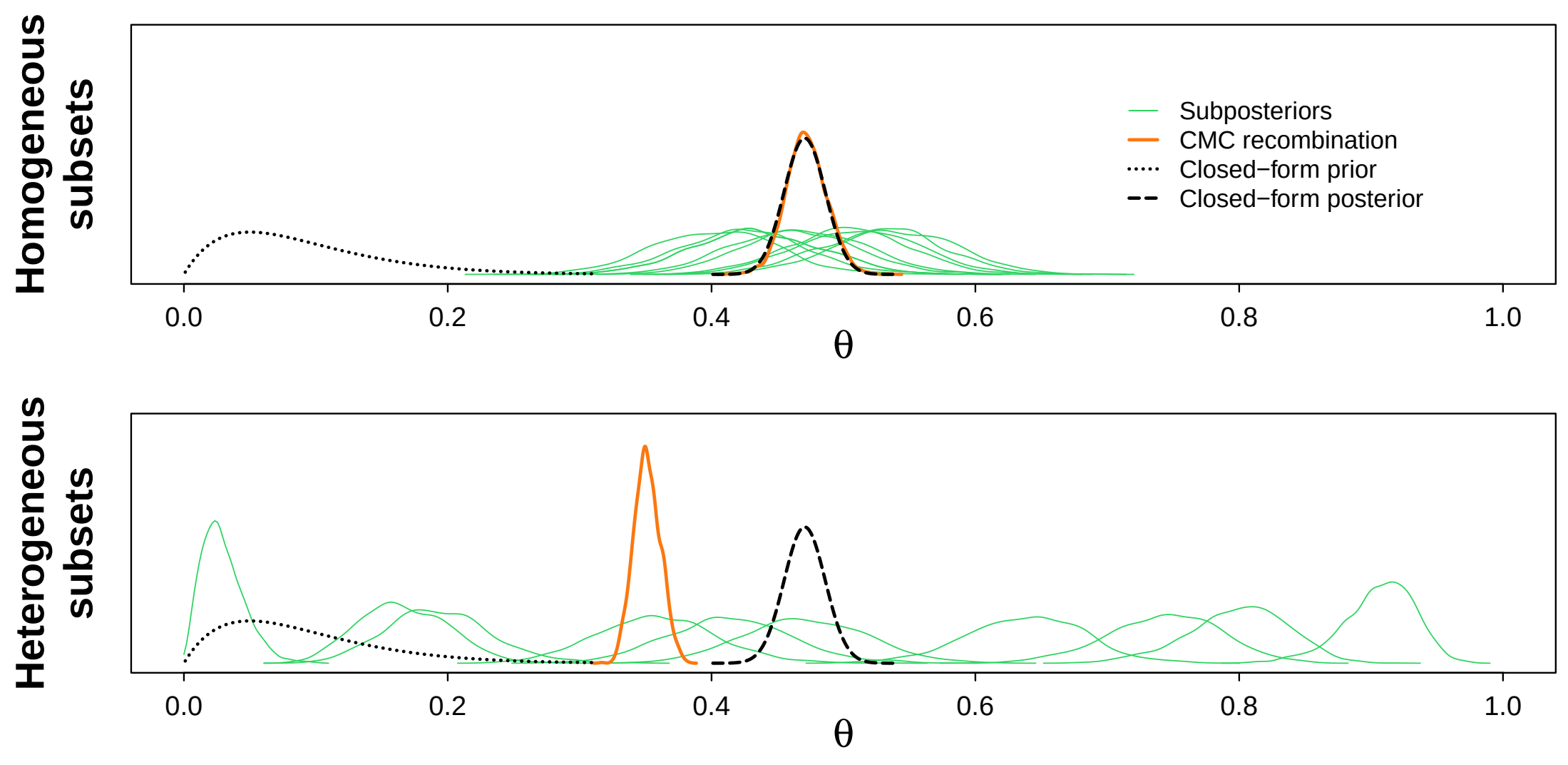

Figure 1. Posterior recombination in a simple Beta-Binomial model. Homogenous subsamples are required in order for the CMC recombination to properly track the target posterior.

Sampling from the Root of a Prior

In order to generate MCMC samples from each subposterior k it becomes necessary to sample from the K^{th} root of the overall prior $p(\theta)$:

$$p(\theta \mid \mathcal{D}) \propto \prod_{k=1}^{K} p(\theta)^{1/K} \cdot p(\mathcal{D}_k \mid \theta)$$

In the case of the Beta-Binomial model above, for example, each subposterior requires sampling from the K^{th} root of a Beta distribution:

$$p(\theta)^{1/K} = \left(\frac{\theta^{a-1}(1-\theta)^{b-1}}{\mathcal{B}(a, b)}\right)^{1/K}$$

It is possible to sample from this distribution in Stan[3] by defining a custom function on top of the log-transformation of the predefined `beta()`:

```
functions{
    real beta_root_lpdf(real theta, real a, real b,
                        real K) {
        return beta_lpdf(theta | a, b) / K;
    }
}
model{
    theta ~ beta_root(a_prior, b_prior, K);
    x ~ binomial(n, theta);
}
```

where **x** and **n** refer to observations in one of K shards. Using this model separately for each shard returns MCMC samples from the corresponding subposterior.

In the following sections we apply the CMC recombination over an important model of choice behavior and a real-world dataset.

The Matching Law

The Matching Equilibrium is a model of choice that explains the distribution of behavioral investment among different sources of reward as a function of the distribution of returns obtained from them[4].
In scenarios with two sources of reward, strict matching predicts:

$$\frac{B_1}{B_2} = \frac{R_1}{R_2},$$

where B_i is the amount of investment in alternative i (e.g., the number of responses to that alternative) and R_i is the amount of reward delivered by the same alternative.
Two systematic deviations from strict matching are commonly observed in experimental and natural settings: bias and under- (or over-) sensitivity to the reward ratio. These deviations can be quantified by the generalized matching equilibrium[5]:

$$\log\left(\frac{B_1}{B_2}\right) = \alpha + \beta \cdot \log\left(\frac{R_1}{R_2}\right)$$

Where the deviations previously mentioned are special cases of the linear parameters:

- Bias occurs when $\alpha \neq 0$ and represents a preference towards one alternative that is constant across all possible reward ratios:
  - $\alpha > 0$ bias towards alternative 1
  - $\alpha < 0$ bias towards alternative 2
- Anomalous sensitivity occurs if $\beta \neq 1$. Such scenarios can be divided into under- and over-matching:
  - $\beta > 1$ is known as over-matching since the resulting preference towards alternative 1 is more extreme than the distribution of rewards with respect to the same alternative.
  - $\beta < 1$ is under-matching since the preference towards the first alternative is less extreme than predicted by strict matching. The limiting case $\beta = 0$ reflects an scenario in which the distribution of behavior among alternatives is the same regardless of how well those alternatives pay off.

The α and β parameters are unobservable but can be inferred after observing B_i and R_i. The core of the model is the simple linear relationship presented above, while the appropriate link functions depend on the nature of the measurements of behavior B_i and returns R_i, along with their log transformations.

NFL Dataset

In the following sections we present a Bayesian implementation of the generalized matching model to account for attacking decisions in American Football. Specifically, we model the choices between passing and rushing in all NFL games played over the last decade (2014-2023).

For the following analyses we extracted the information from:

- 297,583 plays (passes and rushes only).
  - **178,609** passes, with **1,317,505** total yards gained by passing.
  - **118,974** rushes, with **513,003** total yards gained by rushing.
- 2,436 games.
- 32 teams.

The moderate size of this dataset allowed us to approximate the target posteriors using both divide-and-conquer and CMC recombination, and also using all the observations at once in a single computer to evaluate the performance of the recombination strategy.

- *The dataset we used was curated and is maintained by Daren Willman, and is available at `https://nflsavant.com/`.*
- *Our code, figures, and this poster can be found in `https://github.com/JLBaroja/NFL_matching`*

Hiearchical Model with Team Effects

In this model we assume each team t has its own bias (α_t) and sensitivity (β_t) parameters, each of which depends on its own hiearchical distribution. The observed variables are presented in `monospace` and correspond to the number of passes, rushes, and the yards obtained under each of those decisions in the plays of team t that took place in quarter q in each year y.

$$\mu_\alpha \sim \text{Normal}(0, 1) \qquad \mu_\beta \sim \text{Normal}(0, 1)$$
$$\sigma_\alpha \sim \text{Uniform}(0, 2) \qquad \sigma_\beta \sim \text{Uniform}(0, 2)$$
$$\alpha_t \sim \text{Normal}(\mu_\alpha, \sigma_\alpha) \qquad \beta_t \sim \text{Normal}(\mu_\beta, \sigma_\beta)$$

$$\text{logit}(\theta_{tyq}) \leftarrow \alpha_t + \beta_t \cdot \log\left(\frac{\texttt{yards\_pass}_{tyq}}{\texttt{yards\_rush}_{tyq}}\right)$$

$$\texttt{n\_pass}_{tyq} \sim \text{Binomial}\left(\texttt{n\_pass}_{tyq} + \texttt{n\_rush}_{tyq}, \theta_{tyq}\right)$$

We implemented this model a) using all the observations at once in a single computer, and b) dividing the dataset into 10 shards and computing each subposterior sequentially. Crucially, we ensured each subsample included approximately the same number of observations from each team in order to generate exchangeable data shards.

In both cases a) and b) we generated 2000 MCMC samples from the posterior (or subposterior) distribution with each of 3 chains. We used Stan to easily implement the K^{th} root of the Normal and Uniform prior distributions over the hieararchical parameters required to approximate the subposterior of each data shard.

In all cases the MCMC samples we report below met standard criteria of appropriate convergence.

Results

Figure 2 presents the results of the CMC recombination for this model. The upper panels correspond to the hiearchical means of biases (μ_α, left panel) and sensitivities (μ_β). For these variables the CMC result closely aligns with the posteriors obtained by running the model with all observations at once. Moreover, the hiearchical means indicate that, on average, teams are biased towards rushing and their preference towards passing is less extreme than predicted by matching. Such results could suggest that the decision of passing or rushing is also influenced by other factors other than the raw gain in yards, such as the relative risk of each alternative or the cost they require from key players in the team.

The CMC performance is poor when tracking the hiearchical dispersion parameters σ_α and σ_β. This result could reflect a deficient sharding with respect to these variables, as suggested by the large differences between some subposteriors, and may also reflect inherent limitations of CMC when dealing with truncated, non-symmetric posterior distributions (e.g.[6]).

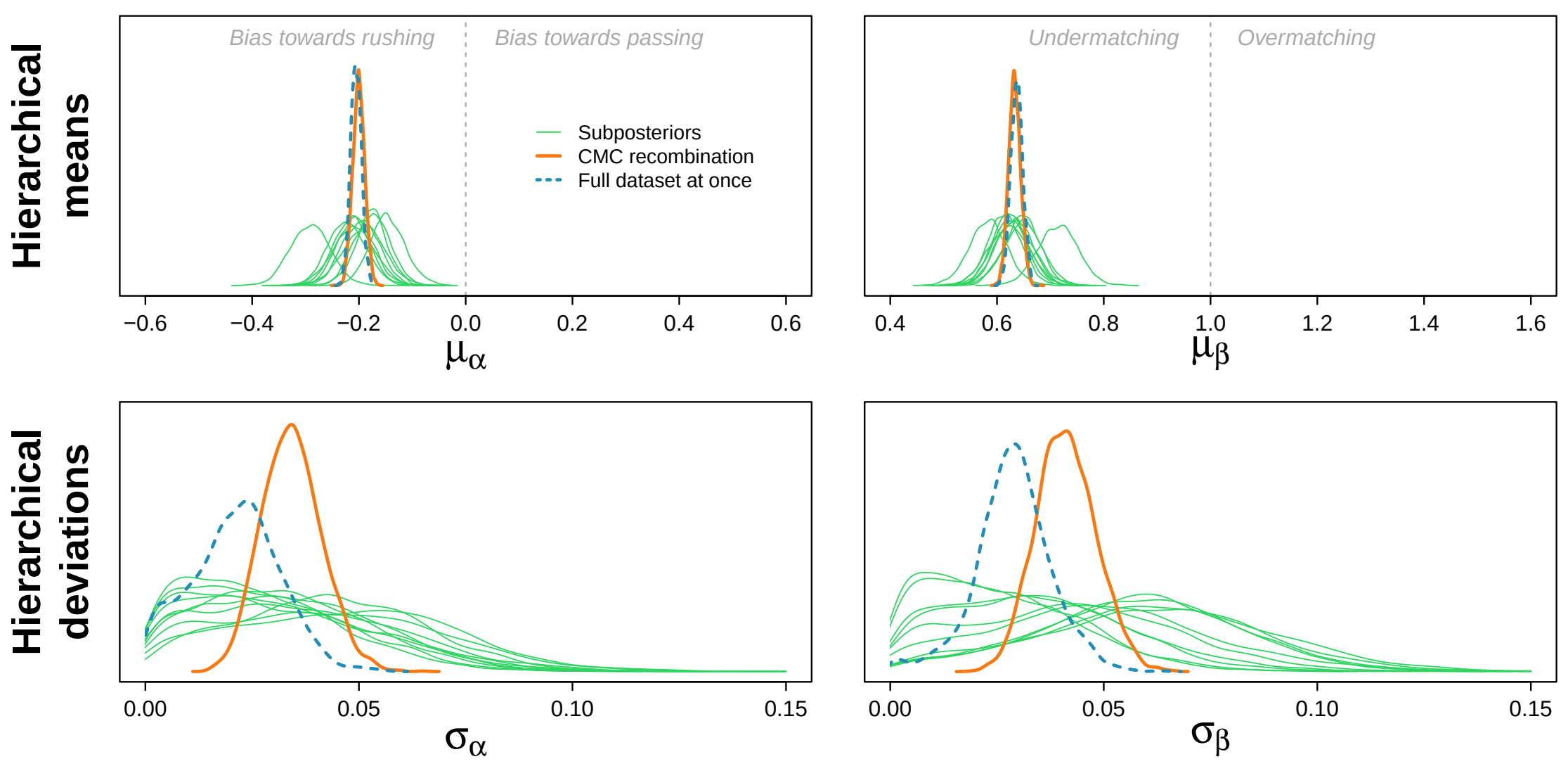

Figure 2. Posterior distributions over hiearchical nodes. Divide-and-conquer methods reasonably track the means of team effects, but they overestimate the dispersion between teams.

Conclusions

- Divide-and-conquer methods offer an efficient way to perform Bayesian inference with large data sets.
- By splitting the full set of observations into smaller shards it becomes possible to closely evaluate and correct each subposterior separately. The final recombination step is computationally cheap and integrates the information from all shards into a single final posterior distribution.
- Cognitive models can benefit from these tools when dealing with data sets too large to be handled efficiently by a single computer, opening the possibility of informing such models with large-scale data sets occurring in and outside the lab, such as online consumer behavior, standardized testing, or human brain activity data under a Bayesian framework.

References

[1] Angelino, E. et al. (2016). *Patterns of Scalable Bayesian Inference*. doi:10.1561/2200000052.

[2] Scott, S. L. et al. (2016). Bayes and big data: the consensus Monte Carlo algorithm. *International Journal of Management Science and Engineering Management*, *11(2)*, 78-88.

[3] Stan Development Team (). RStan: the R interface to Stan. R package version 2.26.24 `https://mc-stan.org/`.

[4] Houston, A. et al. (2021). Matching behaviors and rewards. *Trends in Cognitive Sciences*, *25(5)*, 403-415.

[5] Baum, W. M. (1974). On two types of deviation from the matching law: bias and undermatching. *Journal of the Experimental Analysis of Behavior*, *22*, 231-242.

[6] Mesquita, D. et al. (2020). Embarrasingly parallel MCMC using deep invertible transformations. *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, 1244-1252.

This work was supported by NSF grants #1850849 and #2051186.

jbaroja@uci.edu, joachim@uci.edu

cidlab.com