空间广义线性模型

代码实现

黄湘云

2018 年 3 月

目录

1 贝叶斯框架

1.1 简单分层模型

考虑分层模型，以 8schools 数据集为例[1]

$$\mu \sim \mathcal{N}(0,5)$$

$$\tau \sim \text{Half-Cauchy}(0,5)$$

$$\theta_n \sim \mathcal{N}(\mu,\tau)$$

$$y_n \sim \mathcal{N}(\theta_n,\sigma_n)$$

其中 $n \in \{1,\ldots,8\}$，$\{y_n,\sigma_n\}$ 是已知数据

```
# stan 表示的模型
writeLines(readLines("code/stan/8schools.stan"))
#> // saved as 8schools.stan
#> data {
#>   int<lower=0> J; // number of schools
#>   real y[J]; // estimated treatment effects
#>   real<lower=0> sigma[J]; // s.e. of effect estimates
#> }
#> parameters {
#>   real mu;
#>   real<lower=0> tau;
#>   real eta[J];
#> }
#> transformed parameters {
```

[1] http://mc-stan.org/users/documentation/case-studies/divergences_and_bias.html

```
#>   real theta[J];
#>   for (j in 1:J)
#>     theta[j] = mu + tau * eta[j];
#> }
#> model {
#>   target += normal_lpdf(eta | 0, 1);
#>   target += normal_lpdf(y | theta, sigma);
#> }
```

```r
library(rstan)
#> Loading required package: ggplot2
#> Loading required package: StanHeaders
#> rstan (Version 2.17.3, GitRev: 2e1f913d3ca3)
#> For execution on a local, multicore CPU with excess RAM we recommend calling
#> options(mc.cores = parallel::detectCores()).
#> To avoid recompilation of unchanged Stan programs, we recommend calling
#> rstan_options(auto_write = TRUE)
options(mc.cores = 2) # 两个线程
rstan_options(auto_write = TRUE)
```

加载数据

```r
schools_dat <- list(J = 8,
                    y = c(28,  8, -3,  7, -1,  1, 18, 12),
                    sigma = c(15, 10, 16, 11,  9, 11, 10, 18))
```

加载模型

```r
fit <- stan(file = 'code/stan/8schools.stan', data = schools_dat,
            iter = 1000, chains = 4, seed = 483892929, refresh = 1200)
#> Warning: There were 1 divergent transitions after warmup. Increasing adapt_del
#> http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
#> Warning: Examine the pairs() plot to diagnose sampling problems
```

模型输出

```r
print(fit)
#> Inference for Stan model: 8schools.
#> 4 chains, each with iter=1000; warmup=500; thin=1;
#> post-warmup draws per chain=500, total post-warmup draws=2000.
#>
#>            mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
#> mu         7.97    0.16 5.04  -2.00   4.68   8.14  11.17  17.62   953    1
#> tau        6.91    0.25 5.86   0.29   2.61   5.45   9.84  22.07   528    1
#> eta[1]     0.38    0.02 0.92  -1.44  -0.23   0.41   1.05   2.11  2000    1
#> eta[2]     0.01    0.02 0.85  -1.64  -0.53   0.00   0.54   1.78  2000    1
#> eta[3]    -0.21    0.02 0.95  -2.04  -0.81  -0.22   0.41   1.67  2000    1
#> eta[4]     0.00    0.02 0.89  -1.73  -0.60  -0.01   0.60   1.79  2000    1
#> eta[5]    -0.35    0.02 0.87  -2.06  -0.95  -0.37   0.22   1.38  2000    1
#> eta[6]    -0.20    0.02 0.88  -1.91  -0.81  -0.22   0.36   1.55  2000    1
#> eta[7]     0.35    0.02 0.89  -1.47  -0.21   0.35   0.92   2.13  2000    1
#> eta[8]     0.04    0.02 0.89  -1.70  -0.53   0.04   0.69   1.72  2000    1
#> theta[1]  11.74    0.28 9.03  -1.89   6.11  10.35  15.88  33.48  1007    1
#> theta[2]   7.84    0.14 6.13  -4.58   3.88   7.78  11.70  20.46  2000    1
#> theta[3]   5.99    0.19 7.90 -12.61   1.82   6.56  10.75  20.51  1753    1
#> theta[4]   7.75    0.14 6.46  -5.44   3.88   7.61  11.57  21.15  2000    1
#> theta[5]   5.15    0.14 6.39  -8.92   1.25   5.76   9.51  16.94  2000    1
#> theta[6]   6.08    0.14 6.43  -8.27   2.33   6.49  10.31  18.09  2000    1
#> theta[7]  10.86    0.16 7.00  -1.68   6.22  10.22  14.77  27.13  2000    1
#> theta[8]   8.39    0.17 7.42  -6.75   3.80   8.43  12.69  23.84  2000    1
#> lp__     -39.39    0.12 2.66 -45.39 -41.01 -39.23 -37.48 -34.76   508    1
#>
#> Samples were drawn using NUTS(diag_e) at Mon Mar  5 23:56:24 2018.
#> For each parameter, n_eff is a crude measure of effective sample size,
#> and Rhat is the potential scale reduction factor on split chains (at
```

```
#> convergence, Rhat=1).
```

1.2 泊松

1.3 二项

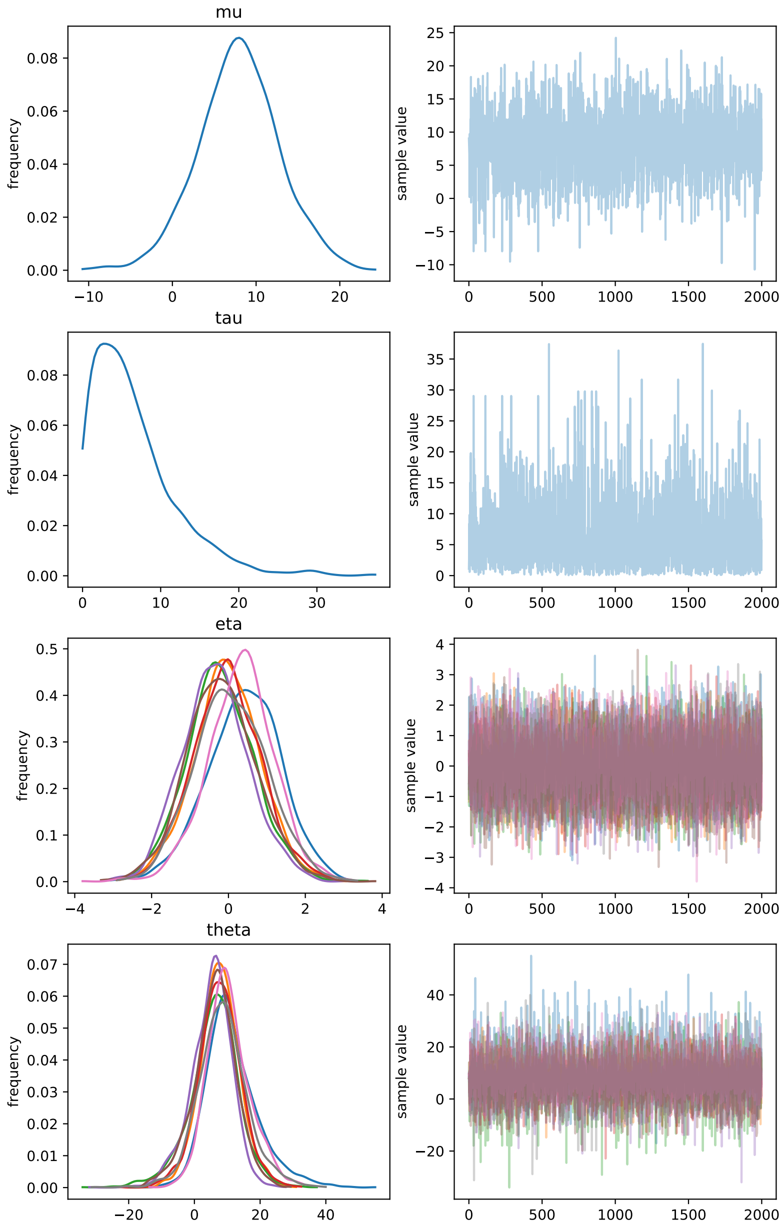

图 1.1: 迭代过程/参数分布/诊断图

2 程序实现

2.1 *PrevMap* 包

(Giorgi and Diggle, 2017) 将 MCML 和 MCMC 方法应用于空间广义线性混合效应模型的参数估计和预测，

2.2 *geoR* 与 *geoRglm* 包

2.3 Stan 框架

Stan[1] 是一种概率编程语言（Carpenter et al., 2017），可以替代 BUGS（**B**ayesian inference **U**sing **G**ibbs **S**ampling）(Lunn et al., 2009) 作为 MCMC 的高效实现，可用于贝叶斯框架下，标准地统计模型的参数估计，Stan 提供多种语言的接口实现，方便起见，本文采用它提供的 R 语言接口 – rstan 包 (Stan Development Team, 2018)。 基于 GPU 加速是一个不错的选择，Stan 开发者也把 GPU 加速列入开发日程。scikit-cuda (Givon et al., 2015) ArrayFire (Yalamanchili et al., 2015) 等基于 CUDA 开发的通用加速框架获得越来越多的关注。类似 Stan 的编程框架还有 PyMC 框架

2.4 R 进程信息

```
sessionInfo()
#> R version 3.4.3 (2017-11-30)
#> Platform: x86_64-pc-linux-gnu (64-bit)
#> Running under: CentOS Linux 7 (Core)
#>
```

[1] http://mc-stan.org/

```
#> Matrix products: default
#> BLAS: /usr/local/lib64/R/lib/libRblas.so
#> LAPACK: /usr/local/lib64/R/lib/libRlapack.so
#>
#> locale:
#>  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
#>  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
#>  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
#>  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
#>  [9] LC_ADDRESS=C               LC_TELEPHONE=C
#> [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
#>
#> attached base packages:
#> [1] stats     graphics  grDevices utils     datasets  methods   base
#>
#> other attached packages:
#> [1] rstan_2.17.3       StanHeaders_2.17.2 ggplot2_2.2.1
#>
#> loaded via a namespace (and not attached):
#>  [1] Rcpp_0.12.15     knitr_1.20       magrittr_1.5     munsell_0.4.3
#>  [5] colorspace_1.3-2 rlang_0.2.0      stringr_1.3.0    plyr_1.8.4
#>  [9] tools_3.4.3      parallel_3.4.3   grid_3.4.3       gtable_0.2.0
#> [13] xfun_0.1         htmltools_0.3.6  yaml_2.1.17      lazyeval_0.2.1
#> [17] rprojroot_1.3-2  digest_0.6.15    tibble_1.4.2     bookdown_0.7.1
#> [21] gridExtra_2.3    codetools_0.2-15 inline_0.3.14    evaluate_0.10.1
#> [25] rmarkdown_1.9.2  stringi_1.1.6    compiler_3.4.3   pillar_1.2.1
#> [29] rticles_0.4.1    scales_0.5.0     backports_1.1.2  stats4_3.4.3
```

斜体用于扩展包和框架，如 *knitr*、*PrevMap*、*CUDA*、*Stan* 等，粗体用于软件，如 **R**、**Python** 等，等宽体用于代码和代码块。

参考文献

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32.

Giorgi, E. and Diggle, P. J. (2017). PrevMap: An R package for prevalence mapping. *Journal of Statistical Software*, 78(8):1–29.

Givon, L. E., Unterthiner, T., Erichson, N. B., Chiang, D. W., Larson, E., Pfister, L., Dieleman, S., Lee, G. R., van der Walt, S., Menn, B., Moldovan, T. M., Bastien, F., Shi, X., Schlüter, J., Thomas, B., Capdevila, C., Rubinsteyn, A., Forbes, M. M., Frelinger, J., Klein, T., Merry, B., Merill, N., Pastewka, L., Clarkson, S., Rader, M., Taylor, S., Bergeron, A., Ukani, N. H., Wang, F., and Zhou, Y. (2015). scikit-cuda 0.5.1: a Python interface to GPU-powered libraries.

Lunn, D., Spiegelhalter, D., Thomas, A., and Best, N. (2009). The bugs project: Evolution, critique and future directions. *Statistics in Medicine*, 28(25):3049–3067.

Stan Development Team (2018). RStan: the R interface to Stan. R package version 2.17.3.

Yalamanchili, P., Arshad, U., Mohammed, Z., Garigipati, P., Entschev, P., Kloppenborg, B., Malcolm, J., and Melonakos, J. (2015). ArrayFire - A high performance software library for parallel computing with an easy-to-use API.