Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *http://www.jstatsoft.org/*

Rgbp: An R Package for Hierarchical Modeling and Method Checking

Joseph Kelly
Harvard University

Carl Morris
Harvard University

Hyungsuk Tak
Harvard University

Abstract

Rgbp is an R package for estimating and validating a two-level model (a random-effects mixed model). The estimation procedure utilizes Bayesian machinery and the validation involves checking frequency properties of the procedure via repeated sampling (which we call "method checking"). It is found that **Rgb** characterized by such Bayesian-frequentist reconciliation has strong points of both sides, working well in small samples and yielding good coverage probability. Following three examples for Gaussian, Poisson and Binomial data validate this point.

Keywords: hierarchical model, multilevel model, random effects mixed model, method checking, coverage probability, Normal, Poisson, Binomial, shrinkage, R.

1. Introduction

Rgbp is an R package for estimating and validating a two-level model (a random-effects mixed model). The estimation procedure utilizes Bayesian machinery and the validation involves checking frequency properties of the procedure via repeated sampling (which we call "method checking"). It is found that even in small samples our procedure yields good frequency properties in comparison to other methods such as Maximum Likelihood Estimation (MLE). This package will be useful for frequentists and Bayesians alike. Bayesians are able to use the package to see a non-informative reference point before and after constructing their full-Bayesian hierarchical model and frequentists, now have a procedure that will provide confidence intervals of a random-effect mixed model with good repeated sampling properties.

2. Multilevel Structure

A two-level or multilevel model, also called a conditionally independent hierarchical model Kass and Steffey (1989), is a very powerful tool for exploring the hierarchical structure in data. For example, we can imagine that there exists a district-level hierarchy (bigger population) for observed school-level data, or a state-level hierarchy for observed hospital-level data in a certain state.

`gbp`, one of the functions in **Rgbp**, fits such a hierarchical model whose first-level hierarchy has a distribution of observed data and second-level (bigger population hierarchy) has a conjugate prior distribution on the first-level parameter. The `gbp` function allows users to choose one of three types of multilevel models, such as Normal-Normal, Poisson-Gamma, and Binomial-Beta, based on their datasets.

2.1. Normal-Normal

The following is a general Normal-Normal hierarchical model. For reference, V_j ($\equiv \sigma^2/n_j$) below is assumed to be known or to be accurately estimated, and subscript j indicates the j-th group among k groups in the dataset.

$$y_j|\mu_j \overset{ind}{\sim} \text{Normal}(\mu_j, V_j), \tag{1}$$

$$\mu_j|\beta, A \overset{ind}{\sim} \text{Normal}(\mu_{0j}, A), \tag{2}$$

where $\mu_{0j} = x_j^T\beta$, $j = 1, \ldots k$, x_j is the j-th group's covariate vector ($m \times 1$), m is the number of regression coefficients and both β and A are unknown. Note that if there is no covariate then $x_j = 1$ for an intercept term ($m = 1$) and so $\mu_{0j} = \mu_0 = \beta_0$ for all j, resulting in an exchangeable prior distribution. For reference, a parameter with a zero subscript, such as μ_{0j}, represents a mean parameter of the prior (second-level) distribution, *i.e.*, a prior mean. Based on this conjugate prior distribution it is easy to derive the corresponding posterior distribution

$$\mu_j|\mathbf{y}, \beta, A \overset{ind}{\sim} \text{Normal}(\ (1 - B_j)y_j + B_j\mu_{0j},\ (1 - B_j)V_j\ ), \tag{3}$$

where $B_j \equiv V_j/(V_j + A),\ j = 1, \ldots, k$, are called shrinkages.

2.2. Poisson-Gamma

`gbp` is also able to build a Poisson-Gamma multilevel model. Note that a constant, $1/r$, multiplied to the Gamma distribution below is a scale and a square bracket below indicates [mean, variance] of distribution. And for notational consistency, let's define $y_j \equiv z_j/n_j$ for all j.

$$z_j|\lambda_j \overset{ind}{\sim} \text{Poisson}(n_j\lambda_j), \tag{4}$$

$$\lambda_j|\beta, r \overset{ind}{\sim} \frac{1}{r}\text{Gamma}(r\lambda_{0j}) \sim \text{Gamma}\left[\lambda_{0j},\ \frac{\lambda_{0j}}{r}\right], \tag{5}$$

where $\log(\lambda_{0j}) = x_j'\beta,\ j = 1, \ldots, k$, with two unknown hyper-parameters, r and β. The posterior distribution of this Poisson-Gamma model is

$$\lambda_j|\mathbf{y}, \beta, r \overset{ind}{\sim} \frac{1}{r + n_j}\text{Gamma}(r\lambda_{0j} + n_jy_j) \sim \text{Gamma}\left[\lambda_j^*,\ \frac{\lambda_j^*}{r + n_j}\right], \tag{6}$$

where $\lambda_j^* \equiv (1 - B_j)y_j + B_j\lambda_{0j}$, $B_j \equiv r/(r + n_j)$, and $y_j \equiv z_j/n_j$, $j = 1, \ldots, k$.

2.3. Binomial-Beta

Binomial-Beta hierarchical model is the last model that `gbp` can fit. Again, a square bracket below indicates [mean, variance] of distribution.

$$z_j|p_j \overset{ind}{\sim} \text{Binomial}(n_j,\ p_j), \tag{7}$$

$$p_j|\beta, r \overset{ind}{\sim} \text{Beta}(rp_{0j},\ r(1 - p_{0j})) \sim \text{Beta}\left[p_{0j},\ \frac{p_{0j}(1 - p_{0j})}{r + 1}\right], \tag{8}$$

where $\log(\frac{p_{0j}}{1-p_{0j}}) = x_j'\beta,\ j = 1, \ldots, k$ and r and β are two unknown hyper-parameters. Then, corresponding posterior distribution is

$$p_j|\mathbf{y}, \beta, r \overset{ind}{\sim} \text{Beta}(rp_{0j} + n_j y_j,\ r(1 - p_{0j}) + n_j(1 - y_j)) \sim \text{Beta}\left[p_j^*,\ \frac{p_j^*(1 - p_j^*)}{r + n_j + 1}\right], \tag{9}$$

where $p_j^* \equiv (1 - B_j)y_j + B_j p_{0j}$, $B_j \equiv r/(r + n_j)$, and $y_j \equiv z_j/n_j$, $j = 1, \ldots, k$.

2.4. Hyper-prior Distribution

In our examples the hyper-prior distribution is an assumed distribution on the second-level parameters and with the goal of objectivity in mind `gbp` assumes non-informative hyper-prior distributions.

$$\beta \sim \text{Uniform on } \mathbf{R}^m \quad \text{and} \quad A \sim \text{Uniform}(0, \infty) \quad (\text{or } \frac{1}{r} \sim \text{Uniform}(0, \infty)), \tag{10}$$

where m is the number of regression coefficients. As for β, it is a reasonable choice to assume a flat (non-informative) distribution because information about the location gets plentiful as the number of groups (k) increases. The next flat prior distribution of the second-level variance component A (or $1/r$) was chosen for its good repeated sampling properties and for posterior propriety under moderate conditions.

3. Estimation

3.1. Shrinkage Estimation

Estimating the shrinkage factors $(B_1, \ldots B_k)$ is the key estimation problem with the hierarchical models `gbp` assumes. As we can see in (3), (6), and (9), the posterior means are a linear function of the shrinkage factors and the posterior variances are also linear (Gaussian), quadratic (Poisson), or cubic (Binomial) functions. A natural method then to estimate $E(\mu_j|\mathbf{y})$ and $Var(\mu_j|\mathbf{y})$ is to first estimate the shrinkage factors.

3.2. Adjustment for Density Maximization

It is noted that the shrinkage factors $(B_1, \ldots B_k)$ are a function of the second-level variance component, *i.e.*, $B_j \equiv V_j/(V_j + A) = B_j(A)$ for Gaussian and $B_j \equiv r/(r + n_j) = B_j(r)$

for Poisson and Binomial models. Current popular methods of estimation via maximization (MLE or MAP) of these unknown shrinkage factors can result in estimates lying on the boundary of the parameter space. In the Normal-Normal model this situation typically arises when the within-group variation is greater than the between-group variation resulting in a parameter estimate of $\hat{A} = 0$ Morris and Lysy (2012).

To continue with a maximization-based estimation procedure but to steer clear of aforementioned issues we make use of adjustment for density maximization (ADM) Morris and Tang (2011). In general ADM is a procedure to obtain estimates of posterior moments via maximization if the underlying posterior can be approximated by a Pearson family. For our purposes we can approximate the posterior distribution of a shrinkage factor with the Beta distribution allowing us to finally obtain estimates of the posterior moments, *i.e.*, $E(B_j|\text{data})$ and $Var(B_j|\text{data})$, without any trouble that MLE can cause. Please refer to Morris and Tang (2011) for additional advantages of ADM.

Once we estimate these two moments of shrinkage, we can also estimate the posterior moments given only the data ($\mathbf{y}$). Taking the Normal-Normal model as an example we note the following identities

$$E(\mu_j|\mathbf{y}) = E(E(\mu_j|\mathbf{y},\beta,A)|\mathbf{y}) = E((1-B_j)y_j + B_j\mu_{0j}|\mathbf{y}). \quad (11)$$

$$Var(\mu_j|\mathbf{y}) = E(Var(\mu_j|\mathbf{y},\beta,A)|\mathbf{y}) + Var(E(\mu_j|\mathbf{y},\beta,A)|\mathbf{y}). \quad (12)$$

Note that both $E(\mu_j|\mathbf{y},\beta,A)$ and $Var(\mu_j|\mathbf{y},\beta,A)$ are linear functions of the shrinkage factors as specified in (3) and since we can estimate $E(B_j|\mathbf{y})$ and $Var(B_j|\mathbf{y})$ via ADM, we can finally estimate $E(\mu_j|\mathbf{y})$ and $Var(\mu_j|\mathbf{y})$.

3.3. Approximation to Posterior Distribution via Matching Moments

After estimating the two posterior moments, for example $E(\mu_j|\mathbf{y})$ and $Var(\mu_j|\mathbf{y})$, `gbp` reasonably approximates a posterior distribution of the mean effects given the data by assuming a reasonable distribution and matching moments. For the Binomial-Beta model we approximate $p_j|\mathbf{z}$ with a Beta distribution and for the Poisson-Gamma model we approximate $\lambda_j|\mathbf{z}$ with a Gamma distribution. Finally for the Normal-Normal model we actually estimate the first three moments and approximate $\mu_j|\mathbf{y}$ with a skewed-t distribution.

4. Method Checking

Like the two sides of the same coin, checking a statistical model always comes with fitting a model. If a fitted model cannot pass a validation or checking process, we usually go back and forth from estimation and checking steps iteratively. In this sense, checking a fitted model is an interactive procedure for the model justification.

There are two kinds of model justification processes; one is a model checking and the other is a method checking. The model checking is for the justification of the multilevel model on a specific dataset. One possible question is, "Can this dataset benefit from such a multilevel modeling?" Christiansen and Morris (1997) answered this question by using the Negative-Binomial mixture model on Poisson data to justify the second-level hierarchy. They found that their data had more variation than expected of the first-level Poisson distribution and Poisson hierarchical model could successfully account for such additional variation.

Once we make sure that the hierarchical modeling can be appropriate for our data, the following question will be about the validity of interval estimates, the final product of this multilevel modeling. "Does the 95% (can be specified differently) confidence interval obtained via this Bayesian model-fitting process achieve 95% confidence level for any true parameter values?" **Rgbp** has a function called `coverage` to answer this question and it comprises of two parts, generating peudo-dataset and estimating coverage probability.

For reference, the explanation will be based on Normal-Normal model because the idea can be easily applied to the other two models.

4.1. Pseudo-data Generation Process

Figure 1 will be helpful to understand this process. As we can see in (3), the distribution of each true parameter ($\mu_j,\ j = 1, \ldots, k$) is completely determined by two hyper-parameters, A and $\beta_{(m\times 1)}$. So, once we fix these hyper-parameters at specific values, we can generate true parameters. Suppose we sampled 500 $\mu_{(k\times 1)}$'s, *i.e.*, $\{\mu^{(i)}_{(k\times 1)},\ i = 1, \ldots, 500\}$ from the prior distribution in (2), where A and β are given. Then, we can also generate $\{\mathbf{y}^{(i)}_{(k\times 1)},\ i = 1, \ldots, 500\}$ given each μ_{ij}, where i indicates the i-th pseudo-dataset and j does the j-th group.

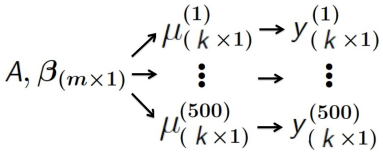

Figure 1: Pseudo-data generating process

4.2. Coverage Estimation Process

Next, `coverage` fits the Normal-Normal model 500 times using the 500 peudo-datasets in order to obtain $500\times k$ interval estimates.

Simple Unbiased Estimates of Coverage Probabilities

Let's define an indicator variable, I_{ij}, which is 1 if the j-th group's interval estimate from the i-th pseudo-dataset includes μ_{ij} and 0 otherwise. One way to estimate the coverage probability is taking average over these indicator variables. We call it a simple unbiased coverage probability estimate. For example, $\bar{I}_1 = \sum_{i=1}^{500} I_{i1}/500$ is the estimated coverage probability for the first group.

Rao-Blackwellized Unbiased estimates of Coverage Probabilities

Rao-Blackwellization improves accuracy of an unbiased estimator by taking conditional expectation given a sufficient statistic. Based on this idea, we define $E(I_{ij}|y_{ij}, A, \beta)$, where A and β were given at first (see 4.1) and y_{ij} is a sufficient statistic. This expectation is the same as $\Pr(\hat{\mu}_{ij,low} \leq \mu_{ij} \leq \hat{\mu}_{ij,upp}|y_{ij}, A, \beta)$, where $(\hat{\mu}_{ij,low},\ \hat{\mu}_{ij,upp})$ is the j-th group's interval estimate on the i-th dataset. We can calculate this probability because we know the distribution of $\mu_{ij}|y_{ij}, A, \beta$ in (3). Note that conditioning on y_{ij} is equivalent to conditioning on $\mathbf{y}$ as

long as A and β are known. Then, we can estimate the first group's coverage probability by $\sum_{i=1}^{500} E(I_{i1}|y_{i1}, A, \beta)/500$, which is also unbiased but more accurate than the previous simple estimator.

5. Examples

5.1. 31 Hospitals: Known Second-level Mean

Suppose a person living in the New York state (NY) has been suffering from severe coronary heart disease. If this person is supposed to receive the coronary artery bypass graft (CABG) surgery soon, he or she might want to find the most reliable hospital for dealing with such a surgery. On top of that, if this person can figure out each hospital's ability to handle this surgery, it will be useful for choosing a hospital.

For this purpose, this person gathered data of 31 hospitals in NY composed of the number of deaths (z) within a month of CABG surgeries and total number of patients (n) receiving CABG surgeries in each hospital. The following code is an example of input based on the last ten hospital data.

```
R> z <- c( 14,   9,  15,  13,  35,  26,  25,  20,   35,   27)
R> n <- c(593, 602, 629, 636, 729, 849, 914, 940, 1193, 1340)
```

In addition, while one looks for such information, suppose one knows that the state-level death rate per exposure of this surgery is 0.020.

The multilevel modeling that assumes a bigger population-level hierarchy will be insightful in this problem. Here, we presume a state-level (NY) hierarchy governing the true death rates (λ) of CABG surgery of all the hospitals in NY. This perspective is to view the true death rates of those 31 hospitals as sampled from the state-level population distribution whose mean is 0.020.

Assuming an additional hierarchy is reasonable, a model-fitting process begins. We know the true death rate per exposure after CABG surgery might be small and the caseload (n_j) is much bigger than the number of deaths (z_j) in each hospital. Then the independent Poisson distribution, *i.e.*, $z_j|\lambda_j \overset{ind}{\sim} \text{Poisson}(n_j\lambda_j)$, $j = 1, \ldots, 31$, would be the first choice to describe the uncertainty in our data. Note that caseload (n_j) can be interpreted as an exposure, which is not necessarily an integer.

Next, `gbp` will help us fit the Poisson multilevel model with the Gamma conjugate prior distribution on the true death rate λ_j whose mean is 0.020 ($\lambda_0 = 0.020$) as described in the section 2.2. For reference, the number of regression coefficients (m) is 0 because we do not need to estimate the prior mean via log-linear regression (see section 3.2).

```
R> data(hospital)
R> z <- hospital$z
R> n <- hospital$n
R> p <- gbp(z, n, mean.PriorDist = 0.02, model = "pr")
R> p
```

```
Summary for each unit (sorted by n):

          obs.mean    n prior.mean shrinkage low.intv post.mean upp.intv post.sd
1            0.045   67       0.02     0.834    0.011     0.024    0.042   0.008
2            0.029   68       0.02     0.832    0.010     0.022    0.038   0.007
3            0.024  210       0.02     0.616    0.011     0.021    0.035   0.006
4            0.043  256       0.02     0.568    0.017     0.030    0.046   0.008
5            0.033  269       0.02     0.556    0.015     0.026    0.041   0.007
6            0.044  274       0.02     0.551    0.018     0.031    0.047   0.008
7            0.043  278       0.02     0.548    0.018     0.030    0.047   0.007
8            0.014  295       0.02     0.533    0.008     0.017    0.029   0.005
9            0.029  347       0.02     0.492    0.014     0.024    0.038   0.006
10           0.037  349       0.02     0.491    0.017     0.029    0.043   0.007
11           0.039  358       0.02     0.484    0.018     0.030    0.045   0.007
12           0.018  396       0.02     0.459    0.010     0.019    0.030   0.005
13           0.028  431       0.02     0.438    0.015     0.024    0.037   0.006
14           0.025  441       0.02     0.433    0.013     0.023    0.035   0.005
15           0.027  477       0.02     0.414    0.015     0.024    0.036   0.006
16           0.045  484       0.02     0.410    0.023     0.035    0.050   0.007
17           0.030  494       0.02     0.405    0.016     0.026    0.039   0.006
18           0.022  501       0.02     0.402    0.012     0.021    0.032   0.005
19           0.028  505       0.02     0.400    0.015     0.025    0.036   0.005
20           0.020  540       0.02     0.384    0.012     0.020    0.031   0.005
21           0.028  563       0.02     0.374    0.016     0.025    0.037   0.005
22           0.024  593       0.02     0.362    0.014     0.022    0.033   0.005
23           0.015  602       0.02     0.358    0.010     0.017    0.026   0.004
24           0.024  629       0.02     0.348    0.014     0.023    0.033   0.005
25           0.020  636       0.02     0.346    0.012     0.020    0.030   0.005
26           0.048  729       0.02     0.316    0.027     0.039    0.053   0.007
27           0.031  849       0.02     0.284    0.019     0.028    0.038   0.005
28           0.027  914       0.02     0.269    0.017     0.025    0.035   0.005
29           0.021  940       0.02     0.264    0.014     0.021    0.030   0.004
30           0.029 1193       0.02     0.220    0.020     0.027    0.036   0.004
31           0.020 1340       0.02     0.201    0.014     0.020    0.027   0.003
colMeans     0.029  517       0.02     0.438    0.015     0.025    0.037   0.006
```

For reference, we need to type '`R> print(p, sort = FALSE)`' instead of '`R> p`' in order to list hospitals by the order of data input in the above output. '`R> p`' automatically sorts the output by the increasing order of caseload (n_j).

The output contains information about sample mean (y_j), caseload (n_j), known prior mean (λ_0), shrinkage estimate ($\hat{B}_j$), lower bound of interval estimate ($\hat{\lambda}_{low}$), posterior mean ($\hat{\lambda} = E(\lambda_j|\mathbf{z})$), upper bound of interval estimate ($\hat{\lambda}_{upp}$), and standard deviation of posterior distribution ($sd(\lambda_j|\mathbf{z})$).

As we can see in (6), the posterior mean, $(1 - B_j)y_j + B_j\lambda_0$, is a linear function of shrinkage, $B_j \equiv r/(r + n_j)$, locating between the sample mean and prior mean ($\lambda_0 = 0.02$). It makes sense because r can be interpreted as the amount of prior information and n_j as the amount

of observed information. If the second level has more information than the first level, then the sample mean shrinks towards the prior mean more than 50%. This point is clear in the above output because, as caseload increases, shrinkage decreases, depending less on the second level information.

A function "`summary`" shows selective information on hospitals and more detailed estimation result as below. To be specific, it displays some hospitals (not all as above) with minimum, median, and maximum caseloads (n_j). On top of that, more specific estimation results, such as the estimation result of $\alpha \equiv \log(1/r)$, follow. Note that when we do not know the prior mean in advance unlike this hospital problem, `gbp` fits a linear regression for the Normal model, a log-linear regression for the Poisson model, or a logistic regression for the Binomial model and a summary of regression fit will appear.

```
R> summary(p)

Main summary:

                  obs.mean    n prior.mean shrinkage low.intv post.mean
Unit w/ min(n)       0.045   67       0.02     0.834    0.011     0.024
Unit w/ median(n)    0.045  484       0.02     0.410    0.023     0.035
Unit w/ max(n)       0.020 1340       0.02     0.201    0.014     0.020
Overall Mean         0.029  517       0.02     0.438    0.015     0.025

                  upp.intv post.sd
                     0.042   0.008
                     0.050   0.007
                     0.027   0.003
                     0.037   0.006

Second-level Variance Component Estimation Summary:
alpha = log(A) for Gaussian and log(1/r) for Binomial and Poisson data:

  post.mode.alpha post.sd.alpha
1          -5.818         0.411
```

Since estimated α is -5.818, we can easily calculate $\hat{r} = \exp(5.818) = 336$, which is a good indicator of how valuable and informative the hypothetical second-level hierarchy is. It means that observed sample means of hospitals whose caseloads are less than 336 will shrink toward the prior mean (0.020) more than 50%. For example, the shrinkage estimate of the first hospital ($\hat{B}_1 = 0.834$) was calculated by 336 / (336 + 67), where 67 is its caseload (n_1), and its posterior mean is $(1 - 0.834) * 0.045 + 0.834 * 0.020 = 0.024$. As for this hospital, using more information from the prior distribution is an appropriate choice because its observed amount of information (67) is far less than the amount of state-level information (336).

We also need a graphical summary that can give us valuable insight buried in pile of numbers and a function '`plot`' is exactly for this purpose.

```
R> plot(p)
```

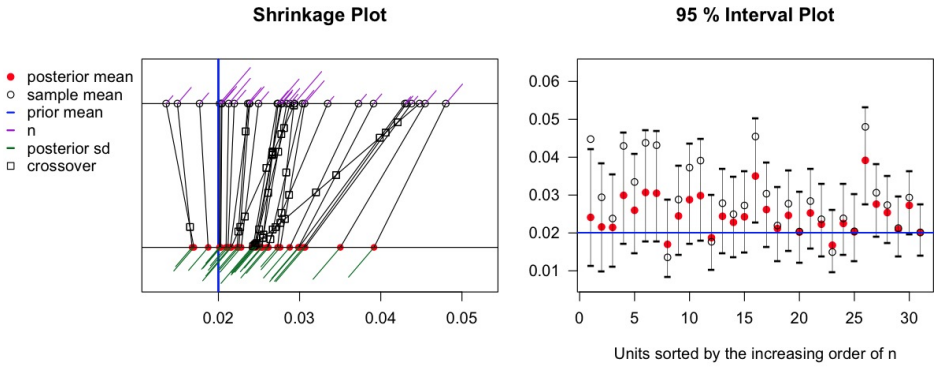

Figure 2: Shrinkage plot and 95% interval plot

The regression towards the mean (RTTM) is obvious in the left-side graph; the observed sample means, empty dots on the upper line, are shrinking towards the known second-level mean (a blue vertical line at 0.02) to the different extents. Note that some hospitals' ranks have changed by shrinking much harder towards 0.02 than others. For example, the empty square symbol at the crossing point of the two left-most lines (8th and 23rd hospitals on the list above) indicates that seemingly safest hospital among 31 hospitals in terms of the observed sample mean was not actually safer than the second safest hospital.

Intuitively, the result of multilevel modeling makes more sense than that of naive sample mean. For example, suppose there are two hospitals, whose sample means (z_j/n_j) are 0 and 0.001 and caseloads (n_j) are 1 and 1000 each. Do you believe that the former hospital, whose observed death rate is 0, is better than the latter and are you going to choose the former hospital? Borrowing information from state-level hierarchy seems reasonable for the former hospital because it is hard to judge its true death rate per exposure with just one caseload. Though somewhat extreme, this is what happened to the two left-most hospitals in the first plot and this is why hierarchical modeling is a reasonable choice for this dataset.

The estimated 95% intervals are displayed on the right-side plot. We can clearly see that all the posterior means (red dots) are between sample mean (empty dots) and second-level mean (a blue horizontal line). For reference, if we want to draw this plot by the order of data input, `plot(p, sort = FALSE)` is the right adjustment.

This interval plot can give us one more valuable insight, which we could not have noticed. Let's look at the 8th and the 31st hospitals on the graph (or on the outcome for numeric reference). The point estimate of the true death rate per exposure of the 31st hospital is higher than the one of the 8th. But, the upper bound of interval estimate of this 31st hospital is lower than that of the 8th. This interval plot makes the 31st hospital emerge as one of your candidates.

Also it reveals that the 23rd hospital, whose estimated true rate was the smallest, has also the smallest upper bound. If you are a risk-adverse, this hospital will attract you most strongly. And if you already chose this 23rd hospital compared to the 8th from the shrinkage plot, your decision might become stronger at this point, excluding the 8th hospital with more certainty.

Then, how reliable are these intervals? Does our procedure to generate interval estimates have good repeated sampling property? The following method checking will answer this

question. The `coverage` function below generates 10,000 pseudo-datasets regarding the estimated r (= 336.37) as a given true value. For reference, we can try any other value of r, for example $r = 200$, by replacing below code with '`R> pcv <- coverage(p, A.or.r = 200, mean.PriorDist = 0.02, nsim = 10000)`.

Also, `gbp` can give us interval estimates with different confidence level, for example 90%, and the appropriate code adjustment is `R> p <- gbp(z, n, Alpha = 0.9, mean.PriorDist = 0.02, model = "pr")`. Then, the function `coverage` will evaluate whether interval estimates achieve 90% confidence level.

```
R> pcv <- coverage(p, nsim = 10000)
```

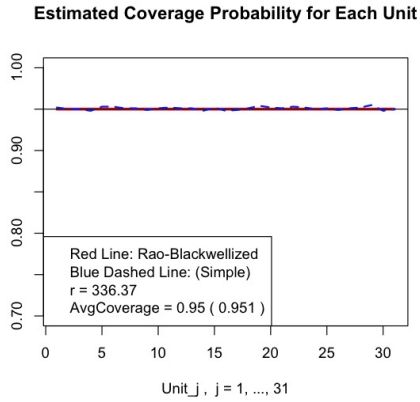

Figure 3: Coverage plot via method checking

It estimated coverage probabilities by simple unbiased estimates (a blue dashed line) and Rao-Blackwellized unbiased estimates (a red line). Both lines are indistinguishable from the horizontal line at 0.95 and the estimated overall average coverage rate is 0.950 from the Rao-Blackwellized estimate. This result shows that the interval estimates from the suggested multilevel modeling on this particular dataset have nice repeated sampling property.

Morris and Christiansen (1995) also looked into a similar ranking problem in hierarchical modeling, taking shrinkage into account.

5.2. 8 Schools: Unknown Second-level Mean and No Covariate

As stated in **??** the Education Testing Service (ETS) conducted randomized experiments in eight separate schools (group) to test whether students (unit) SAT scores are effected by coaching. The dataset contains the estimated coaching effects on SAT scores ($y_j, j = 1, \ldots, 8$) and standard errors ($se_j, j = 1, \ldots, 8$) of the eight schools Rubin (1981).

```
R> y  <- c(12, -3, 28,  7,  1,  8, 18, -1)
R> se <- c(18, 16, 15, 11, 11, 10, 10,  9)
```

Due to the nature of the test each school's coaching effect has an approximately Normal sampling distribution with known sampling variance, *i.e.*, standard error of each school is

assumed to be known or to be accurately estimated. Hence, we can use gbp to fit a Normal-Normal hierarchical model:

```
R> data(schools)
R> attach(schools)
R> g <- gbp(y, se, model = "gr")
R> g

Summary for each unit (sorted by n):

          obs.mean   se prior.mean shrinkage low.intv post.mean upp.intv post.sd
5            -1.00  9.0      8.168     0.408  -13.297     2.737   16.692   7.634
2             8.00 10.0      8.168     0.459   -7.255     8.077   23.361   7.810
7            18.00 10.0      8.168     0.459   -1.289    13.484   30.821   8.176
4             7.00 11.0      8.168     0.507   -8.780     7.592   23.602   8.257
6             1.00 11.0      8.168     0.507  -13.027     4.633   20.131   8.441
1            28.00 15.0      8.168     0.657   -2.315    14.979   38.763  10.560
3            -3.00 16.0      8.168     0.685  -17.130     4.650   22.477  10.096
8            12.00 18.0      8.168     0.734  -10.208     9.189   29.939  10.227
colMeans      8.75 12.5      8.168     0.552   -9.163     8.168   25.723   8.900
```

The output from gbp provides an easy way to read summary of the results of the estimation. From this there seems to be little evidence that the training provided much of an added benefit with every school's 95% posterior interval containing 0. **Rgbp** also provides functionality to plot the results of the analysis as seen in 5.2.

```
R> plot(g)
```

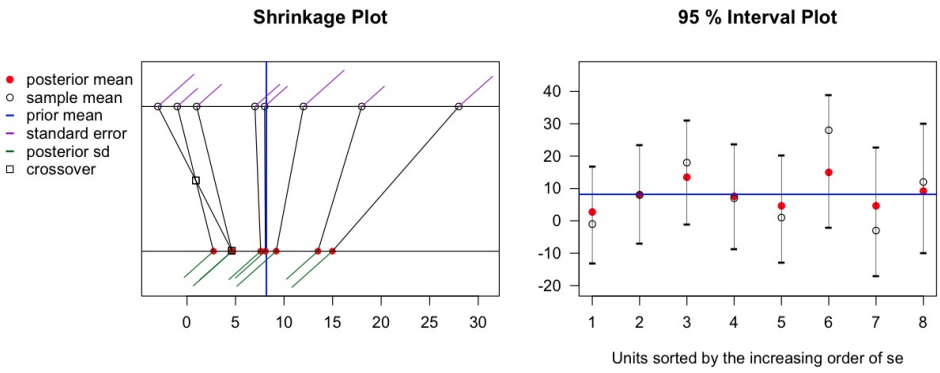

Figure 4: Shrinkage plot and 95% interval plot for 8 schools

Plotting the results provides a visual aid to understanding but is only largely beneficial when the number of groups (k) is small. In the case where the number of groups is large **Rgbp** provides a summary feature:

```
R> summary(g)

Main summary:

                    obs.mean   se prior.mean shrinkage low.intv post.mean
Unit w/ min(se)        -1.00  9.0      8.168     0.408  -13.297     2.737
Unit w/ median(se)1     1.00 11.0      8.168     0.507  -13.027     4.633
Unit w/ median(se)2     7.00 11.0      8.168     0.507   -8.780     7.592
Unit w/ max(se)        12.00 18.0      8.168     0.734  -10.208     9.189
Overall Mean            8.75 12.5      8.168     0.552   -9.163     8.168

                     upp.intv post.sd
                       16.692   7.634
                       20.131   8.441
                       23.602   8.257
                       29.939  10.227
                       25.723   8.900

Second-level Variance Component Estimation Summary:
alpha = log(A) for Gaussian and log(1/r) for Binomial and Poisson data:

  post.mode.alpha post.sd.alpha
1           4.768         1.139


Regression Summary:

      estimate   se z.val p.val
beta0    8.168 5.73 1.425 0.154
```

An integral part of fitting any model is to check the method of estimation. Namely we can generate new pseudo-data from our assumed model by fixing the hyper-parameter values (A and μ_0 in this example) at their estimates. It is then possible to simulate "true" θ_i for each group i and re-fit the model many times to estimate the coverage probabilities of our procedure.

```
R> gcv <- coverage(g, nsim = 10000)
```

As seen in 5.2 the desired 95% coverage probability was achieved for each group in this example.

5.3. 18 Baseball Players: Unknown Prior Mean and One Covariate

The following dataset from the New York Times published on 26 April 1970 contains information on the batting averages of 18 major league baseball players through their first 45 official at-bats of the 1970 season Efron and Morris (1975). In addition one covariate relating to the position of each player was observed and for illustrative purposes, we transform this variable into a binary indicator (1 if a player was an outfielder and 0 otherwise).

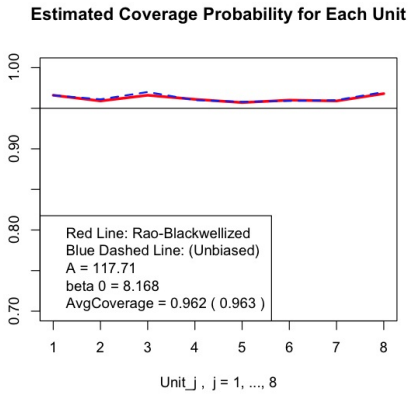

Figure 5: Coverage plot via method checking for 8 schools

```
R> z <- c(18, 17, 16, 15, 14, 14, 13, 12, 11, 11, 10, 10, 10, 10, 10,  9,  8,  7)
R> n <- c(45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45, 45)
R> x <- c( 1,  1,  1,  1,  1,  0,  0,  0,  0,  1,  0,  0,  0,  1,  1,  0,  0,  0)
```

The data indicate that independent Binomial distribution is appropriate probability distribution for describing each player's number of hits (z_j) among 45 at-bats conditioning on the unknown true batting average, *i.e.*, $z_j|p_j \overset{ind}{\sim} \text{Binomial}(45, p_j)$, $j = 1, \ldots, 18$. For reference, we can also assume the Normal distribution using $y_j = z_j/n_j$ and $V_j = \bar{y}(1-\bar{y})/n$, where $\bar{y} = \sum_j z_j / \sum_j n_j$, $j = 1, \ldots, 18$.

Suppose we are interested in their true batting averages and believe that the outfielder's batting averages are different from other positions' ones. So, it is desirable to assume two different second-level hierarchies for outfielders and for other positions, within each of which sharing information and regressing toward the mean (RTTM) occur. Our multilevel modeling provides a way to incorporate such information (position) seamlessly into the second-level hierarchy.

```
R> b <- gbp(z, n, x, model = "br")
R> b

Summary for each unit (sorted by n):

         obs.mean  n   X1 prior.mean shrinkage low.intv post.mean upp.intv post.sd
1           0.400 45 1.00      0.310     0.715    0.248     0.335    0.429   0.046
2           0.378 45 1.00      0.310     0.715    0.244     0.329    0.420   0.045
3           0.356 45 1.00      0.310     0.715    0.240     0.323    0.411   0.044
4           0.333 45 1.00      0.310     0.715    0.236     0.316    0.403   0.043
5           0.311 45 1.00      0.310     0.715    0.230     0.310    0.396   0.042
6           0.311 45 0.00      0.233     0.715    0.179     0.256    0.341   0.041
7           0.289 45 0.00      0.233     0.715    0.175     0.249    0.331   0.040
8           0.267 45 0.00      0.233     0.715    0.171     0.243    0.323   0.039
9           0.244 45 0.00      0.233     0.715    0.166     0.237    0.315   0.038
```

```
10          0.244 45 1.00      0.310     0.715     0.210     0.291     0.379    0.043
11          0.222 45 0.00      0.233     0.715     0.161     0.230     0.308    0.038
12          0.222 45 0.00      0.233     0.715     0.161     0.230     0.308    0.038
13          0.222 45 0.00      0.233     0.715     0.161     0.230     0.308    0.038
14          0.222 45 1.00      0.310     0.715     0.202     0.285     0.375    0.044
15          0.222 45 1.00      0.310     0.715     0.202     0.285     0.375    0.044
16          0.200 45 0.00      0.233     0.715     0.155     0.224     0.302    0.038
17          0.178 45 0.00      0.233     0.715     0.148     0.218     0.297    0.038
18          0.156 45 0.00      0.233     0.715     0.140     0.211     0.292    0.039
colMeans    0.265 45 0.44      0.267     0.715     0.191     0.267     0.351    0.041
```

Our model reflects on the additional information, *i.e.*, indicator covariate (1 for outfielder and 0 for other positions), estimating two different prior means, 0.310 and 0.233. Also note that shrinkage estimates are the same for all players. It makes sense because shrinkage ($B_j \equiv r/(r+n_j)$) is determined by the relative amount of information between the first-level and the second-level hierarchies. The fact that all players have the same amount of observed information ($n_j = 45$) and $\hat{r} = \exp(4.727) = 113$ from below summary tell why the estimated shrinkage is 0.715 (=113 / (113 + 45)).

Note that the posterior standard deviation tends to be bigger among outfielders than among others. Let's see its formula in (9). We can notice that the estimated posterior mean is a critical factor that makes posterior variances different from each player because every player has the same at-bats (n_j) and r. As the posterior mean gets closer to 0.5, the posterior variance gets bigger and has the biggest value when the posterior mean is 0.5. Intuitively, it makes sense. If a true batting average is 0.1, we can easily expect less hits. But if it is 0.5, we are less certain whether an at-bat would be more likely a hit or not, like a coin tossing. Our model is automatically reflecting on such uncertainty.

```
R> summary(b)

Main summary:

                           obs.mean  n   X1 prior.mean shrinkage low.intv
Unit w/ min(obs.mean)         0.156 45 0.00      0.233     0.715    0.140
Unit w/ median(obs.mean)1     0.244 45 0.00      0.233     0.715    0.166
Unit w/ median(obs.mean)2     0.244 45 1.00      0.310     0.715    0.210
Unit w/ max(obs.mean)         0.400 45 1.00      0.310     0.715    0.248
Overall Mean                  0.265 45 0.44      0.267     0.715    0.191

                          post.mean upp.intv post.sd
                              0.211    0.292   0.039
                              0.237    0.315   0.038
                              0.291    0.379   0.043
                              0.335    0.429   0.046
                              0.267    0.351   0.041

Second-level Variance Component Estimation Summary:
alpha = log(A) for Gaussian and log(1/r) for Binomial and Poisson data:
```

```
  post.mode.alpha post.sd.alpha
1          -4.727         0.957


Regression Summary:

      estimate    se  z.val p.val
beta0   -1.194 0.131 -9.129 0.000
beta1    0.389 0.187  2.074 0.038
```

This summary also includes the result of logistic regression fit because we did not know prior means in advance. We had to estimate them via this logistic regression model. Let's look at the p-value of the indicator covariate. It shows that the two prior means distinguishing outfielders from other positions were significantly different (p-value = 0.038), supporting our initial belief. The positive sign of $\hat{\beta}_1$ indicates that the mean batting average for all the outfielders tends to be higher than that for those in the other positions.

```
R> plot(b)
```

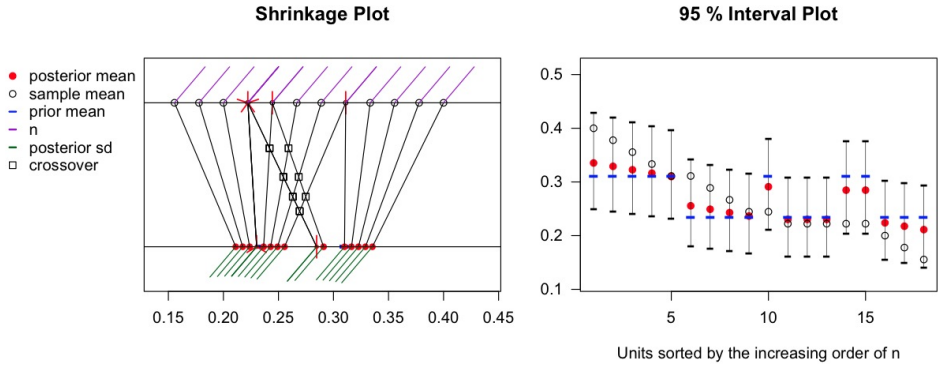

Figure 6: Shrinkage plot and 95% interval plot

Let's see shrinkage and 95% interval plots for more intuition. It is obvious that sample means (empty dots) on the upper line shrink towards two different means, 0.310 and 0.233. For reference, the red line symbols on dots are for when two or more points have the same mean and are plotted over each other. For example, five players (from the 11th player to the 15th) have the same sample mean (0.222) and at this point on the upper line, there are short red lines toward five directions.

For reference, the observed sample mean among outfielders is 0.308 and that among other positions is 0.231. But why are the estimated two prior means from this Binomial model 0.310 and 0.233 each? This can be attributed to the logistic regression which estimates prior mean by a non-linear logit function of $x'\beta$, causing a small bias (see 3.3). The Normal model can avoid this small bias because it uses a linear regression to estimate these prior means.

The 95% interval plot shows us range of true batting average for each player, which clarifies the regression toward the mean (RTTM) within two groups. Let's see the 10th, 14th, and 15th players on the graph for example. They are outfielders but their batting averages (sample mean) are far lower than the first five outfielders. RTTM means that it can happen by their bad luck though in the long run their batting averages will come back to their expected ones as outfielders (blue horizontal line). The fact that their sample means are closer to the lower part of their 95% intervals supports this argument. So, their posterior means (red dots) can also be interpreted as their RTTM.

Then, how much can we trust these interval estimates? The following method checking that regards the estimated values, $\hat{r}$ (=112.95) and $\hat{\beta}$ (= $(-1.194,\ 0.389)^T$), as true values when it generates pseudo-datasets will answer it.

```
R> bcv <- coverage(b, nsim = 10000)
```

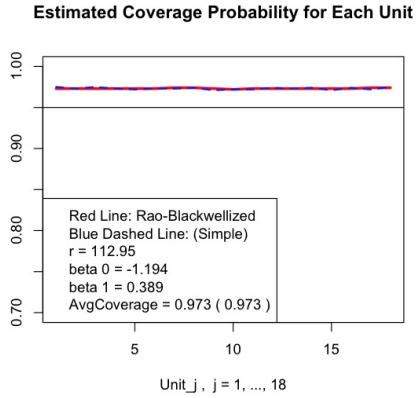

Figure 7: Coverage plot via method checking

Finally, the estimated average coverage probability is 0.973 based on the Rao-Blackwellization, conservatively satisfying the definition of the 95% confidence interval.

6. Discussion

Rgbp is an R package for estimating and validating a two-level hierarchical model. The package aims to provide a procedure that is not only fast and easy to use but has good frequency properties and can be used in many scenarios. The package provides "method checking" functionality to examine repeated sampling properties and test that the method is valid at a particular parameter value.

In addition to good frequency properties Bayesians will be able to use the package to see the results from a non-informative reference point. This allows the user to examine whether it is worth to implement a full Bayesian model (which is often more time consuming).

Due to the fact that the estimation procedures in **Rgbp** rely on differentiation the package is extremely quick in fitting the available models. This makes the package ideal to be used in simulation studies where a hierarchical model needs to be fitted at every iteration and where running a full Bayesian model (via MCMC) would be computationally impractical.

In short the **Rgbp** is an R package that utilizes Bayesian machinery to provide a method of estimating two-level hierarchical models from many families in a fast and computationally efficient manner.

7. Acknowledgments

References

Christiansen CL, Morris CN (1997). "Hierarchical Poisson Regression Modeling." *Journal of the American Statistical Association*, **92**(438), pp. 618–632. ISSN 01621459. URL http://www.jstor.org/stable/2965709.

Efron B, Morris C (1975). "Data Analysis Using Stein's Estimator and its Generalizations." *Journal of the American Statistical Association*, **70**(350), pp. 311–319. ISSN 01621459. URL http://www.jstor.org/stable/2285814.

Kass RE, Steffey D (1989). "Approximate Bayesian Inference in Conditionally Independent Hierarchical Models (Parametric Empirical Bayes Models)." *Journal of the American Statistical Association*, **84**(407), pp. 717–726. ISSN 01621459. URL http://www.jstor.org/stable/2289653.

Morris C, Tang R (2011). "Estimating Random Effects via Adjustment for Density Maximization." *Statistical Science*, **26**(2), pp. 271–287. ISSN 08834237. URL http://www.jstor.org/stable/23059992.

Morris CN (1983). "Parametric Empirical Bayes Inference: Theory and Applications." *Journal of the American Statistical Association*, **78**(381), pp. 47–55. ISSN 01621459. URL http://www.jstor.org/stable/2287098.

Morris CN, Christiansen C (1995). "Hierarchical Models for Ranking and for Identifying Extremes, With Application." In J Bernardo, J Berger, A Dawid, A Smith (eds.), *Bayesian Statistics 5*, pp. 227–296. New York: Oxford University Press.

Morris CN, Lysy M (2012). "Shrinkage Estimation in Multilevel Normal Models." *Statistical Science*, **27**(1), 115–134.

Rubin DB (1981). "Estimation in Parallel Randomized Experiments." *Journal of Educational Statistics*, **6**(4), pp. 377–401. ISSN 03629791. URL http://www.jstor.org/stable/1164617.

Affiliation:

Joseph Kelly
Department of Statistics
Harvard University
1 Oxford Street, Cambridge, MA
E-mail: kelly2@fas.harvard.edu
URL: http://www.people.fas.harvard.edu/~kelly2/

Carl Morris
Department of Statistics
Harvard University
1 Oxford Street, Cambridge, MA
E-mail: morris@fas.harvard.edu

Hyungsuk Tak
Department of Statistics
Harvard University
1 Oxford Street, Cambridge, MA
E-mail: hyungsuk.tak@gmail.com

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume VV, Issue II *Submitted:* yyyy-mm-dd
MMMMMM YYYY *Accepted:* yyyy-mm-dd