METHOD ARTICLE

Fast effect size shrinkage software for beta-binomial models of allelic imbalance [version 1; peer review: 3 approved with reservations]

Joshua P. Zitovsky ¹, Michael I. Love ^{1,2}

¹Department of Biostatistics, University of North Carolina at Chapel Hill, Chapel Hill, NC, 27516, USA

²Department of Genetics, University of North Carolina at Chapel Hill, Chapel Hill, NC, 27514, USA

V1 First published: 28 Nov 2019, 8:2024
<https://doi.org/10.12688/f1000research.20916.1>

Latest published: 14 Dec 2020, 8:2024
<https://doi.org/10.12688/f1000research.20916.2>

Abstract

Allelic imbalance occurs when the two alleles of a gene are differentially expressed within a diploid organism, and can indicate important differences in cis-regulation and epigenetic state across the two chromosomes. Because of this, the ability to accurately quantify the proportion at which each allele of a gene is expressed is of great interest to researchers. This becomes challenging in the presence of small read counts and/or sample sizes, which can cause estimates for allelic expression proportions to have high variance. Investigators have traditionally dealt with this problem by filtering out genes with small counts and samples. However, this may inadvertently remove important genes that have truly large allelic imbalances. Another option is to use Bayesian estimators to reduce the variance. To this end, we evaluated the accuracy of three different estimators, the latter two of which are Bayesian shrinkage estimators: maximum likelihood, approximate posterior estimation of GLM coefficients (*apeglm*) and adaptive shrinkage (*ash*). We also wrote C++ code to quickly calculate ML and *apeglm* estimates, and integrated it into the *apeglm* package. The three methods were evaluated on both simulated and real data. *Apeglm* consistently performed better than ML according to a variety of criteria, including mean absolute error and concordance at the top. While *ash* had lower error and greater concordance than ML on the simulations, it also had a tendency to over-shrink large effects, and performed worse on the real data according to error and concordance. Furthermore, when compared to five other packages that also fit beta-binomial models, the *apeglm* package was substantially faster, making our package useful for quick and reliable analyses of allelic imbalance. *Apeglm* is available as an R/Bioconductor package at <http://bioconductor.org/packages/apeglm>.

Open Peer Review

Approval Status

	1	2	3
version 2			
(revision)
14 Dec 2020
version 1
28 Nov 2019

1. **Matthew Stephens**, University of Chicago, Chicago, USA

University of Chicago, Chicago, USA

2. **Jarad Niemi**, Iowa State University, Ames, USA

Ignacio Alvarez-Castro , University of the Republic, Montevideo, Uruguay

3. **Ernest Turro**, University of Cambridge, Cambridge, UK

University of Cambridge, Cambridge, UK

Any reports and responses or comments on the article can be found at the end of the article.

Keywords

RNA-seq, Allelic imbalance, Allele-specific expression (ASE), Beta-binomial, Shrinkage estimation, Empirical Bayes, Bioconductor, Statistical software

This article is included in the **Bioconductor** gateway.

Corresponding author: Michael I. Love (michaelisaiahlove@gmail.com)

Author roles: **Zitovsky JP:** Data Curation, Formal Analysis, Investigation, Software, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Love MI:** Conceptualization, Data Curation, Funding Acquisition, Methodology, Project Administration, Resources, Software, Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: JPZ was supported the National Institutes of Health [R01 HG009125]. MIL was supported by the National Institutes of Health grants [HG009937, R01 MH118349, P01 CA142538 and P30 ES010126].

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2019 Zitovsky JP and Love MI. This is an open access article distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Zitovsky JP and Love MI. **Fast effect size shrinkage software for beta-binomial models of allelic imbalance [version 1; peer review: 3 approved with reservations]** F1000Research 2019, 8:2024 <https://doi.org/10.12688/f1000research.20916.1>

First published: 28 Nov 2019, 8:2024 <https://doi.org/10.12688/f1000research.20916.1>

Introduction

Allelic imbalance (AI) occurs when the two alleles of a gene are expressed at different levels in a diploid organism, and the measurement of AI is valuable in elucidating the factors that regulate the expression of genes. For example, for a diploid organism, the allele on one chromosome may have higher or lower expression levels compared to the allele on the other chromosome due to genetic variation in nearby non-coding regulatory sites, a process known as cis-regulation. Allelic imbalance in expression may also be associated with differential epigenetic state of the genomic region across the chromosomes. In some cases, differential allelic expression resulting from differential epigenetic state can be linked to the parent-of-origin of the alleles, a phenomenon known as genetic imprinting.

One challenge currently faced in allelic expression studies is that estimates for allelic expression proportions can be highly variable in the presence of low read counts and/or small sample sizes. Large estimates of allelic proportions in these cases often result from estimation error as opposed to true differences in allelic expression. Though small samples and low counts are a problem for RNA-seq data in general, they are especially problematic when dealing with allele-specific counts. When a subject is heterozygous for a gene at a particular SNP(s), RNA-seq reads that overlap the SNP(s) allow for quantification of the levels of expression from either chromosome¹. Thus, allelic expression cannot be measured within a gene for subjects that are homozygous for that gene, and the number of samples with allele-specific counts for a gene can be much less than the number of samples in the study. Furthermore, alleles are often differentiated by a single SNP, and RNA-seq reads that do not overlap the SNP cannot be mapped to either allele. For these reasons, the proportion of RNA-seq reads that are allele-specific can be quite low, depending on both read length and heterozygosity of the subjects. For instance, one study with 2x50 base pair (bp) paired-end reads and 30 million heterozygous SNPs from breast tumors of 550 human subjects found that allele-specific counts made up only 3.4% of RNA-seq reads². Experiments making use of model organism crosses can maximize the number of RNA-seq reads overlapping heterozygous SNPs, for example Raghupathy *et al.*³ found in an RNA-seq dataset of a mouse F1 cross that 22% of uniquely mapping reads were allele-specific.

One traditional remedy investigators have used to deal with the challenges of high-variance estimates is to filter out genes that have low counts or small samples. While this does cause the resulting estimates to be more stable and thus representative of true allelic expression proportions, filtering may also remove genes that have true allelic imbalances. Furthermore, the cutoff used to determine what genes to filter out (i.e. how many counts a gene must have for it to not be removed) must be chosen per dataset by the analyst. Another potential remedy to the problem of variable estimates from low-count and low-sample genes is to use Bayesian shrinkage estimators to moderate estimates.

A large number of Bayes estimators have already been developed for allelic expression studies. For instance, MMSEQ⁴ uses

a Gamma prior on allele-specific transcript abundance to provide isoform and allelic imbalance estimates that are more accurate and stable in the face of low coverage. Other methods that have used Bayesian approaches to test for AI include those by Leòn-Novelo *et al.* 2014⁵ and Skelly *et al.*⁶ However, these methods can only test overall AI, and cannot test the effects of covariates, such as different groups, on AI. More recently, a method was developed that expanded on that by Leòn-Novelo *et al.* 2014 and was able to estimate AI within groups as well as compare AI between groups⁷. It uses Bayesian shrinkage estimates for its parameters to shrink allelic proportions within groups toward 0.5, overdispersion toward a pre-specified prior mean, and the total counts of both alleles toward a pooled estimate. While the method is more flexible than the other methods listed, it still cannot estimate the effects of continuous covariates on allelic imbalance, nor can it estimate differences in AI between groups while controlling for additional confounding variables. Furthermore, while the authors showed their method to be effective in reducing type I and type II error in the face of different sources of bias, the advantage of their method in estimation accuracy itself and in the face of genes with low counts need to be more thoroughly investigated.

Though gene expression read counts are typically larger than allele-specific counts and can be measured for all subjects, the uncertainty of estimates in the presence of low counts and/or low sample sizes is still an issue. Thus, several shrinkage estimators for log fold changes in gene expression have also been developed which shrink estimates that are only large due to the variance of the estimator and leave unchanged estimates that are likely to be large due to true expression changes⁸⁻¹¹. Many of these methods directly involve or can easily be applied to linear models, which provide great flexibility in the kinds of study designs that can be treated and hypotheses that can be investigated. Though these methods were originally developed for improving accuracy and stability of log fold change estimates in gene expression, several can be directly applied or at least easily extended to estimating the effects of covariates on allelic expression proportions.

To this end, we look at three different estimation methods and their performance on data sets with small-to-moderate numbers of samples: maximum likelihood (ML), approximate posterior estimation of GLM coefficients (apeglm)¹¹ and adaptive shrinkage (ash)¹⁰ ML estimates are based on estimating effects by modelling allele-specific counts with a beta-binomial GLM. Apeglm and ash are Bayesian shrinkage estimators which shrink maximum likelihood-based estimates toward zero. Our results show that while apeglm is not always the best method, it always performs better than ML and never performs much worse than ash for most metrics, making it the most robust and reliable when dealing with small sample sizes in our analysis. We also introduced new source code for the apeglm package to improve computational performance for beta-binomial GLMs, and compared our improved package to other R packages that can also fit beta-binomial GLMs. As the apeglm package can calculate both ML and Bayesian shrinkage estimates, our improvements can be used even by those

who wish not to use shrinkage estimators. Compared to other R packages, we show that `apeglm` with our improved code gives better running times and greater scalability with the number of covariates.

Methods

Estimation methods

We evaluated three estimation methods on their ability to estimate allelic expression proportions (or equivalently, the effects of covariates on allelic expression proportions): maximum likelihood estimation (ML estimation or MLE) with the likelihood described below, approximate posterior estimation of GLM coefficients (`apeglm`) and adaptive shrinkage (`ash`). All analyses were done using R version 3.5.1¹². The first two methods mentioned are implemented in the `apeglm` v.1.7.5 package, while the last is implemented in the `ashr` v.2.2.32 package. When using the `ash` function in the latter package, we set the `method` parameter equal to "shrink". While there are many Bayesian estimation methods that can be used to quantify allelic imbalance, these allow for arbitrary design matrices. For instance, these methods can estimate differences in AI between groups while controlling for, or allowing interactions with, multiple additional variables, and can estimate the effects of continuous variables on AI.

For the g -th gene ($1 \leq g \leq G$), a beta-binomial GLM was fit to model allele-specific counts as follows. Let Y_{ig} be the read counts of the first of the two alleles (which allele is designated as the first allele is arbitrary) for the i -th subject, $1 \leq i \leq I$. Investigators may designate the first and second alleles of a gene as the paternal and maternal alleles or as the alternate and reference alleles. It is assumed that $Y_{ig} \sim \text{BetaBin}(n_{ig}, p_{ig}, \phi_g)$, where n_{ig} is the equal to the total counts of both alleles for the i -th subject, p_{ig} is the probability of counts belonging to the first allele of the i -th subject, and ϕ_g is the overdispersion parameter. For the remainder of this paper, we will refer to the total allele-specific counts for both alleles of a particular gene and for a particular sample as the 'total counts' for that gene and sample. Furthermore, we will refer to the probability that counts for a particular gene belong to a particular allele for a particular sample as the 'allelic proportion' for that particular allele and sample. In this case, $\phi \rightarrow \infty$ implies no overdispersion beyond what would be seen in a binomial distribution and $\phi \rightarrow 0$ implies increasing variance. $n_{1g}, \dots, n_{Ig}$ are assumed to be fixed and known. As the beta-binomial probability density function has multiple forms and parameterizations, we specify our parametrization as:

$$f(y; n, p, \phi) = \frac{\binom{n}{y} B(y + \phi p, n - y + \phi(1 - p))}{B(\phi p, \phi(1 - p))}$$

where B specifies the beta function. Furthermore, let $\mathbf{x}_i$ be the i -th row of the design matrix $\mathbf{X}$ (matrix where columns are vectors of covariates of interest). Potential predictors include disease status for association studies, parent of origin for

imprinting studies, and the presence of a SNP for eQTL linkage studies. We also assume that $p_{ig} = [1 + \exp(-\mathbf{x}_i^T \boldsymbol{\beta}_g)]^{-1}$, or equivalently $\text{logit}(p_{ig}) = \mathbf{x}_i^T \boldsymbol{\beta}_g$, where $\boldsymbol{\beta}_g = (\beta_{1g}, \dots, \beta_{K_g})^T$ is a vector of coefficients representing the effect sizes for the predictors in the design matrix. For ML estimation, $\boldsymbol{\beta}_g$ is estimated via maximum likelihood. Constrained optimization is used for the nuisance parameter ϕ_g with a maximum of 500, so that genes with no overdispersion have finite estimated values of ϕ .

`Apeglm` additionally assumes a zero-centered Cauchy prior distribution for the effects of one of the predictors¹¹. For estimating the effect of the j -th predictor in our model, where $1 \leq j \leq K$ is chosen by the user, and for the g -th gene, we have:

$$\begin{aligned} Y_{ig} | \boldsymbol{\beta}_g &\sim \text{BetaBin}(n_{ig}, p_{ig}, \phi_g) \\ p_{ig} &= \frac{1}{1 + \exp(-\mathbf{x}_i^T \boldsymbol{\beta}_g)} \\ \beta_{jg} &\sim \text{Cauchy}(0, \gamma_j) \end{aligned}$$

`Apeglm` shrinks the effect of one chosen predictor at a time, across all genes. The scale parameter of the Cauchy prior, γ_j , is estimated by pooling information across genes. The posterior distribution of $\boldsymbol{\beta}_g$ is the product of the above Cauchy prior and beta-binomial likelihood, and `apeglm` provides Bayesian shrinkage estimates based on the mode of the posterior as well as standard errors. Genes with lower expression, smaller numbers of heterozygous subjects and higher dispersion in allelic proportions will have flatter likelihoods, which will lead to the prior having more influence and shrinkage being greater. Furthermore, if the ML estimates are tightly clustered about zero, the estimated scale parameter of the Cauchy prior will be smaller. This will lead to more peakedness in the prior and also cause shrinkage to be greater.

The original `apeglm` package estimated regression coefficients using C++ for negative binomial GLMs, while GLMs with other likelihoods, such as the beta-binomial, were fit completely in R. To improve scalability for large data sets with beta-binomial GLMs, we wrote fast C++ code for calculating maximum likelihood and `apeglm` shrinkage estimates of beta-binomial regression coefficients. We also changed the source code to speed up computation of the standard errors (though such computations were still done in R) and prevent convergence issues. Details can be found in the Supplementary Methods section¹³.

`Ash` is a general Empirical Bayes shrinkage estimator for hypothesis testing and measuring uncertainty in a vector of effects of interest, such as a set of log fold changes in gene expression between biological conditions¹⁰. Suppose again that one is interested in the effect sizes of the j -th predictor, $\boldsymbol{\beta}_{\cdot j} = (\beta_{j1}, \dots, \beta_{jG})$, where $1 \leq j \leq K$. `Ash` takes as input a vector of ML estimated effects $\hat{\boldsymbol{\beta}}_{\cdot j} = (\hat{\beta}_{j1}, \dots, \hat{\beta}_{jG})$ and corresponding

estimated standard errors $\sigma_{\beta_j} = (\sigma_{\beta_{j1}}, \dots, \sigma_{\beta_{jG}})$. Here we take the estimated standard errors to be the true standard errors as suggested in the original methodology for ash, though the developers of ash have recently proposed an extension to their method that allows for random errors¹⁴. For all $1 \leq g \leq G$, it is assumed that $\hat{\beta}_{jg} | \beta_{jg} \sim N(\beta_{jg}, \sigma_{\beta_{jg}})$ and that $\beta_{jg} \sim h_j$, where h_j is some unimodal, zero-mode prior distribution. h_j is estimated from the ML estimates using mixtures of uniforms and a point-mass at zero, a choice guided by the author's claim that any unimodal distribution can be approximated as a mixture of uniforms with arbitrary accuracy. The posterior is $\beta_{jg} | \hat{\beta}_{jg} \sim N(\beta_{jg}, \sigma_{\beta_{jg}}) \times h_j$, and ash provides Empirical Bayes shrinkage estimates using the mean of the posterior as well as standard errors. Genes with larger standard errors for their ML estimates will have a flatter likelihood that will be less impactful on the estimation. Thus, estimates for these genes will be shrunk more. Like apegglm, ash can only shrink estimates for one covariate at a time.

Datasets and simulations

We compared the three estimation methods using the data set from the allelic expression study by Crowley *et al.*^{15,16} The study took mice from three divergent inbred strains (CAST/EiJ, PWK/PhJ and WSB/EiJ) and performed a diallel cross. The data set contains ASE counts for 72 mice and 23,297 genes in the resulting cross, with 12 mice of each possible parent combination (e.g. CAST/EiJ as mother and PWK/PhJ as father is one parent combination, and PWK/PhJ as mother and CAST/EiJ as father is another), and an equal number of males and females within each parent combination. Sequencing was performed with the Illumina HiSeq 2000 platform to generate 100-bp paired-end reads and following the TruSeq RNA Sample Preparation v2 protocol. To assure that the mice all had the same alleles, we chose one genotype to focus on, namely the genotype resulting from the cross with CAST/EiJ and PWK/PhJ. The resulting data set, which we will refer to for the remainder of this paper as the 'mouse data set', had 24 mice, 12 of each sex and 12 of each parent of origin, and each mouse had nearly the same nuclear genetic composition as a result of the cross.

To evaluate the estimators on estimating effect sizes of predictors when the truth is known, we first fit an intercept-only beta-binomial model on each gene for the mouse data set. $\phi = [\phi_g]$ is the vector of ML estimates of the overdispersion parameter from each model, and $\mu = [\mu_g]$ is the vector of ML estimates of allelic proportions (which ranges between 0 and 1). 8 mice were then selected from the data set. Denote $N_{I \times G} = [n_{ig}]$ as the matrix of total ASE counts for the 8 mice. Finally, a matrix of counts from one of the alleles $Y_{I \times G} = [y_{ig}]$ was simulated for a sample size of 4 vs. 4, where y_{ig} was simulated from $\text{BetaBin}(n_{ig}, p_{ig}, \phi_g)$, $\text{logit}(p_{ig}) = \mu_g + \beta_g x$, β_g was simulated from a standard normal distribution, and x splits the mice into two groups of size four ($x = 1$ if a mouse is in the first group and 0 otherwise). Samples were drawn from the beta-binomial distribution using the `emdbook` v1.3.11 package¹⁷. We refer to this simulation throughout the paper as the 'standard normal simulation', reflecting the distribution of the true effect sizes.

A second simulation was also performed that was similar in setup to the first, but with modifications to the distribution of β_g and ϕ_g . In many studies, the effect sizes of a predictor will be zero for all but a handful of genes. Thus, β_g was simulated from $t_3/10$ (a Student's t-distribution with 3 degrees of freedom scaled by 1/10), which gave effects mostly close to zero, but with moderate and large effects occasionally appearing (Supplementary Figure 1¹³). Furthermore, the distribution of ϕ_g from the mouse data appeared to be a mixture of two distributions: Genes without overdispersion had an obvious point mass at 500 with 70% proportion, and the remaining 30% genes had a distribution somewhat similar to an exponential with a mean of $\beta = 179$ (Supplementary Figure 2¹³). To get more over-dispersed allele-specific counts, ϕ_g was simulated from $0.5\text{Exp}(\beta=89) + 0.5(500)$, a mixture distribution where one component was exponential with a mean of 89 and had 50% proportion, and the other component was a point mass at 500 and had 50% proportion. We refer to this simulation throughout the paper as the 'Student's t simulation', again reflecting the distribution of the true effect sizes. Note that these two simulations assume a data generating process, specifically the same data generating process as our assumed likelihood.

The estimators were then evaluated on real data with the focus on estimating mean, or gene-wide, allelic imbalance. From the mouse data set, random samples of size 6 were drawn, and this process was repeated 100 times. We will refer to these samples throughout the paper as the 'random subsamples'. For each random subsample, the ML, apegglm and ash estimates of intercept-only models were calculated for the genes (where the intercept term was shrunk), and the MLE of the held-out 18 mice was taken to be the truth. Estimating the intercept in an intercept-only model for each gene is equivalent to estimating overall allelic imbalance for each gene.

Additional simulations were conducted for evaluating computational performance of our improvements to apegglm, to see how well they would scale to larger and more complicated data sets. Allele-specific counts were simulated in a similar manner as the apegglm vignette¹⁸. Briefly, we have $Y_{100 \times 5000} = [y_{ig}]$ as our simulated count matrix for one allele with associated total count matrix $N_{100 \times 5000} = [n_{ig}]$ where rows are samples and columns are genes, $y_{ig} \sim \text{BetaBin}(n_{ig}, p_g, \theta_g)$, $\theta_g \sim U(0, 1000)$, $p_g \sim N(.5, 0.5^2)$, $n_{ig} \sim \text{NB}(\mu_g, 1/\phi_g)$, and μ_g, ϕ_g are based on the airway data set by Himes *et al.*¹⁹ To see how well our improvements scaled with increasing numbers of covariates, the data were split multiple times into differing numbers of groups of approximately equal size, where the number of groups ranged from 2 to 10. With K groups, the design matrix was $X_{100 \times K} = [1 \ x_1 \dots x_{K-1}]$, where x_j is an indicator variable for the $(j+1)$ -th group, or a row vector whose i -th element is 1 if the i -th sample is in the $(j+1)$ -th group and 0 otherwise. A simulation was also conducted to see how well apegglm would work with continuous predictors. This time, Y and N was kept the same, but with the design matrix $X_{100 \times 4} = (1, x_1, x_2, x_3) = [x_{ij}]$, where $x_1 = (1, 0, 1, \dots, 1, 0)^T$ separates the samples into two equally sized groups and $x_2, x_3 \sim N(0, 1)$. x_1 is the covariate whose effect size estimates are shrunk.

Data processing

Genes where at least three samples did not have at least 10 counts were removed, which we considered minimal filtering that shouldn't decrease statistical power. Genes without at least one count for both alleles across all individuals were removed. Genes with a marginally significant sex or parent effect were removed, so that all samples could be assumed independent and identically distributed for all genes. Genes were removed from the mouse data set prior to conducting random sampling from the data set or simulations.

To determine whether sex or parent effects were significant, beta-binomial GLMs were estimated for each gene by maximum likelihood, with a design matrix that included a sex effect (an indicator that was 1 if male and 0 if female), a parent-of-origin effect (an indicator that was 1 if the mother was the CAST/EiJ strain and 0 if the father was the CAST/EiJ strain) and an interaction term. For each gene, if the p-value for the sex, parent-of-origin or interaction effect was less than 0.1, the effect was deemed marginally significant for that gene.

Technical details of evaluations

For each gene, we define the shrinkage score as movement from MLE to zero. We define a gene as (noticeably) shrunk if shrinkage exceeds 0.1, and substantially or most shrunk if shrinkage is greater than $\max(1, |\hat{\beta}_{MLE}|/4)$. For instance, if an `apeglm` estimate for a gene is 0.15 closer to zero than the MLE, then the shrinkage score is 0.15 and the gene is noticeably shrunk but not substantially shrunk by `apeglm`.

Concordance at the top (CAT) plots²⁰ were used to determine which estimation method could best find the most important genes (the genes with the greatest allelic imbalance or largest effect size). For an estimation method, concordance at the top takes the top genes according to the true ranking and compares it to the top genes according to the estimates, where the top genes are the genes with the largest true or estimated effect sizes in absolute value. For instance, a concordance at the top 10 of 90% means that the top 10 genes according to the estimation method and the top 10 genes according to the truth agree for 9 out of 10 genes.

For evaluating the performance of the three methods in estimating intervals, we calculated normality-based 95% confidence and credible intervals (both of which we will abbreviate as CIs) of the ML and `apeglm` estimators using their standard errors, or intervals based on the Laplace approximation of the likelihood and posterior. Such normality-based intervals are the default and suggested method for the `apeglm` package. Credible intervals in the `ashr` package were calculated from directly estimating tail probabilities of the posterior.

For each of the design matrices posited in our computation simulation, computational performance of `apeglm` estimation was compared between the old and new `apeglm` code. From `apeglm` v1.7.5, we set the `method` parameter equal to "betabinCR" to run the new C++ code, and set the

`log.lik` parameter equal to a beta-binomial log-likelihood function to run the old code from before our improvements were introduced (version 1.6.0 of the package). Details can be found in the vignette¹⁸. Computational performance of ML estimation was also compared between our improved `apeglm` package and the following packages: `aod` v1.3.1²¹, `VGAM` v1.1²², `aods3` v0.4²³, `gamlss` v5.1²⁴ and `HRQoL` v1.0²⁵. Computational performance was evaluated using the `microbenchmark` v1.4.6 package²⁶ for estimation of a single gene and elapsed time for estimation of all 5000 genes, on a 2012 15-inch MacBook Pro with an Intel Core i7-3720QM processor.

Determining the optimal filtering rule

In addition to comparing the three estimation methods described above, maximum likelihood estimation paired with optimal filtering criteria was also assessed via concordance at the top. CAT was chosen over other benchmark metrics, such as mean absolute error, as the different number of genes after filtering would make comparisons between filtered MLE and the three unfiltered methods biased. Furthermore, as we were primarily interested in whether a good filtering rule even existed, the true ranking of genes was used to determine the filtering rule. We looked at three rules: 1) removing genes where less than half the samples had a minimum total count threshold, 2) removing genes where less than all the samples had a minimum total count threshold, and 3) removing genes where the sum of total counts across samples was less than a certain threshold. For the remainder of the paper, we will refer to the sum of total counts across samples as the 'summed counts' of a gene. For each rule, various different thresholds were looked at: {0, 10, ..., 200} were potential thresholds for rule 1, {0, 10, ..., 100} were potential thresholds for rule 2, and {0, 50, ..., 1000} were potential thresholds for rule 3. For each rule and threshold, the MLE was calculated and concordance among the top 50, 100, 200, 300, 400 and 500 genes were averaged. We will refer to the rule and threshold that had the best concordance as the 'optimal filtering rule'.

Results

Standard normal simulation

We began by looking at a simulation where allelic counts came from known beta-binomial distributions and effect sizes came from a standard normal distribution. In this simulation, `apeglm` and `ash` successfully shrunk erroneously large estimates and reduced estimation error, particularly for genes that were noticeably shrunk (see Table 1 and Figure 1).

All three estimation methods gave similar mean absolute error (MAE), as many genes did not differ much between the methods (Table 1). In exploring the behavior of shrinkage estimators, we were most interested in genes where shrinkage was high, and thus where estimates would be much closer to or much farther from the truth for one estimation method than for another. Thus, in addition to overall MAE, we also calculated MAE among genes that were noticeably shrunk by `apeglm` and genes that were noticeably shrunk by `ash`, to determine whether there was substantial improvement on average when `apeglm` or `ash` *did* noticeably shrink a gene. Among genes that

Table 1. Performance Metrics for Normal Simulation. MLE: Maximum Likelihood Estimation, apegglm: Approximate Posterior Estimation of Generalized Linear Model Coefficients, ash: Adaptive Shrinkage.

Performance Metric	MLE	Apegglm	Ash
Mean Absolute Error	0.152	0.142	0.141
Mean Absolute Error (apeglm-shrunk genes)	0.498	0.408	0.397
Mean Absolute Error (ash-shrunk genes)	0.469	0.38	0.37
Mean Absolute Error (effect size >2)	0.221	0.203	0.269
Coverage Probability for 95% CI	0.951	0.951	0.949
Average Interval Width for 95% CI	0.761	0.697	0.685

Figure 1. Truth vs. estimate and CAT Plots for normal simulation. **a)** truth vs. estimate plot for MLE. Blue points represent genes substantially shrunk by apegglm only, orange points represent genes substantially shrunk by ash only and green points represent genes substantially shrunk by both ash and apegglm. **b)** truth vs. estimate plots for apegglm. **c)** truth vs. estimate plots for ash. **d)** CAT plot for the three methods as well as for MLE after filtering. CAT: Concordance at the Top, MLE: Maximum Likelihood Estimation, apegglm: Approximate Posterior Estimation of Generalized Linear Model Coefficients, ash: Adaptive Shrinkage.

were shrunk by apegglm, apegglm decreased the mean absolute error by 18.1%, and among genes that were shrunk by ash, ash decreased the mean absolute error by 21.1%. Moreover, from Figure 1a–c, it can be seen that apegglm shrunk most ML estimates that were inflated, bringing them closer to the truth, and mostly left truly large effects alone. Ash also shrunk ML estimates that were inflated, including some inflated estimates missed by apegglm. However, ash also had a tendency to incorrectly and excessively shrink: some genes with estimates close to the truth were severely shrunk, and several genes with truly large effects were shrunk to zero. Because of this tendency to over-shrink, ash performed worse among genes with large effects than among genes with small effects. For instance, among genes with effect sizes greater than two

in absolute value, ash estimates had a higher mean absolute error than the MLE.

Ash and apegglm also performed better than the MLE in determining the most important genes, where concordance at the top was higher regardless of the number of genes being considered (Figure 1d). Apegglm performed slightly better than ash in concordance at the top 100 genes, but otherwise they performed about the same. Concordance at the top for the MLE was optimized when filtering out genes with summed counts less than 350. Using this filtering, we were able to get CAT results better than that of the shrinkage estimates, even if only by a very small amount. Thus, for this simulation, it was possible to outperform both apegglm and ash with filtering alone

(provided that the true ranking of genes was known, and used to determine the optimal filtering rule).

With regard to the extent of shrinkage, both apegglm and ash mainly exhibited shrinkage for genes that had very low counts (Supplementary Figure 3¹³). This is not too surprising for this particular simulation, as after filtering out lowly-expressed genes, the remaining ML estimates were much closer to the truth (Supplementary Figure 4¹³). When comparing shrinkage scores between apegglm and ash, we found that there was a clear upward shift of shrinkage scores for ash (Supplementary Table 1¹³), further showing that ash had more extreme shrinkage than apegglm for this dataset. Though all three methods gave intervals that were similar in coverage probability, average interval width was smaller for apegglm and ash compared to the MLE (Table 1).

Student's t Simulation

We also investigated the performance of the estimators when most of the effect sizes were close to zero and overdispersion was large. Here the shrinkage estimates had even more marked improvement over the ML estimates (see Table 2 and Figure 2).

Apegglm improved mean absolute error by 49.5% among all genes, and by 65.4% among noticeably shrunk genes specifically (Table 2). Ash improved mean absolute error by 52.2% among all genes and by 65.8% among noticeably shrunk genes specifically. These improvements were greater than that seen from the standard normal simulation. Figure 2a–c show that ash successfully shrunk inflated ML estimates closer to the truth while leaving truly large effects mostly unchanged.

Table 2. Performance metrics for Student's t Simulation.

Performance Metric	MLE	Apegglm	Ash
Mean Absolute Error	0.186	0.094	0.089
Mean Absolute Error (apeglm-shrunk genes)	0.353	0.122	0.113
Mean Absolute Error (ash-shrunk genes)	0.336	0.126	0.115
Coverage Probability for 95% CI	0.924	0.937	0.941
Average Interval Width for 95% CI	0.873	0.45	0.456

Figure 2. Truth vs. estimate and CAT Plots for Student's t Simulation. a) truth vs. estimate plot for MLE. Orange points represent genes substantially shrunk by ash only and green points represent genes substantially shrunk by both ash and apegglm. All genes substantially shrunk by apegglm were shrunk by practically the same amount or more by ash. b) truth vs. estimate plots for apegglm. c) truth vs. estimate plots for ash. d) CAT plot for the three methods as well as for ML after filtering. MLE: Maximum Likelihood Estimation, apegglm: Approximate Posterior Estimation of Generalized Linear Model Coefficients, ash: Adaptive Shrinkage.

Apeglm brought many inflated ML estimates closer to the truth as well, but not as many as ash.

Concordance at the top was better for the shrinkage estimates than for the ML estimates, regardless of the number of top genes in question. Furthermore, similar to mean absolute error, the improvements seen from the shrinkage estimates over the MLE was larger than those seen from the standard normal simulation. Ash performed better than apegm in concordance at the top 50 and 100 genes, though performance was similar when looking at larger number of genes. Concordance at the top for the MLE was optimized when filtering out genes where less than half the samples had at least 110 counts. Though this improved CAT by quite a lot, performance was still much lower than apegm and ash. Thus, unlike in the standard normal simulation, the performance in CAT obtained by shrinkage could not be matched with filtering, even when using the true gene ranking to determine the optimal filtering rule.

In this simulation, due to the increased overdispersion, there were many effects that were overestimated or underestimated by ML, even among genes with large counts. Because of this, both ash and apegm exhibited shrinkage for effects across the dynamic range of summed counts, as opposed to only shrinking effects with small counts (Supplementary Figure 5¹³). Together with the true vs. estimate plots, this shows that both apegm and ash can correctly shrink falsely large effects even when the summed counts are large. Ash had larger shrinkage scores than apegm on average, indicating that ash tended to shrink estimates more than apegm (Supplementary Table 2¹³). All methods had coverage slightly less than nominal (95%), ranging from 92 to 94%. However, both apegm and ash had half the average interval width compared to maximum likelihood, despite both having slightly higher coverage rates.

We also conducted simulations similar to that of the standard normal and Student's t, but with 5 vs. 5 samples. Like the 4 vs. 4 case, both apegm and ash had lower average estimation error and higher concordance at the top than the MLE (results not shown).

Sampling from the mouse dataset

To evaluate performance on real data, we took 100 random subsamples of 6 mice from the mouse data set and averaged various performance metrics across the random subsamples. Similar to the simulations, apegm appeared to improve estimation accuracy and shrink erroneously large genes. Ash,

on the other hand, appeared to perform worse than the MLE according to mean absolute error, concordance at the top and interval coverage (see Table 3 and Figure 3).

Among genes that were shrunk by apegm, mean absolute error was 14.2% lower on average for apegm (Table 1). On the other hand, average MAE rose by 34.1% for ash among genes that were shrunk. Moreover, the minimum MAE obtained by ash across all 100 random subsamples was larger than the maximum MAE obtained by apegm (results not shown). From Figure 3a–c, ash appears to be over-shrinking, and some of the genes with the largest held-out effect estimates were shrunk to zero. Though some genes also appeared to have been incorrectly or overly shrunk by apegm, apegm mainly was observed to shrink genes with inflated estimates and over-shrinkage was normally less severe when it occurred.

Both apegm and MLE had universally higher concordance at the top than ash (Figure 1d). While apegm performed slightly better than the MLE in concordance at the top 50 genes, performance was identical when looking at larger numbers of genes. We found that any filtering only decreased concordance at the top, as many top genes had low counts (i.e. the optimal filtering rule was no filtering). The most likely reason for this is that for each random subsample, we are treating the MLE of the held-out set as the truth. Thus, estimation error in the face of low-count genes would affect the held-out effect estimates and bias CAT results to some degree, even though the held-out sets have larger numbers of samples and performance metrics are averaged over many random subsamples. However, because genes with very large held-out effect estimates are more likely to have low counts, metrics that average across all genes, such as mean absolute error, would not be biased as much by estimation error.

A large amount of variability in the ML estimates was discernible for genes with low counts (Supplementary Figure 6¹³). Like in the standard normal simulation, the low-count genes were mainly the ones shrunk by apegm and ash. As the truth vs. estimate plots suggest, ash had larger shrinkage scores than apegm (indicating more extreme shrinkage), and with the difference in shrinkage between the two methods being larger than in the simulations (Supplementary Table 4¹³). Though apegm intervals appeared to have smaller coverage than the ML intervals, the difference in coverage was very small, and average interval width was also 26.8% smaller for apegm than that of maximum likelihood. Ash intervals were slightly more

Table 3. Performance metrics averaged across random subsamples.

Performance Metric	MLE	Apegm	Ash
Mean Absolute Error	0.113	0.102	0.132
Mean Absolute Error (apeglm-shrunk genes)	0.556	0.477	0.772
Mean Absolute Error (ash-shrunk genes)	0.437	0.377	0.586
Coverage Probability for 95% CI	0.944	0.937	0.914
Average Interval Width for 95% CI	0.597	0.437	0.399

Figure 3. Truth vs. estimate and CAT plots for random subsamples. a) through c) is based on one of the 100 random subsamples used in the mouse data benchmarking, and plots the ML estimates of the associated held-out set against the MLE, apegglm and ash estimates from the random subsample, respectively. Orange points represent genes substantially shrunk by ash only and green points represent genes substantially shrunk by both ash and apegglm. All genes substantially shrunk by apegglm were shrunk by practically the same amount or more by ash. d) plots concordance at the top averaged across the 100 random subsamples for each method. MLE: Maximum Likelihood Estimation, apegglm: Approximate Posterior Estimation of Generalized Linear Model Coefficients, ash: Adaptive Shrinkage.

narrow than apegglm, with average interval width 32.5% smaller than that of maximum likelihood, but coverage was also lower.

As the mouse data set only had 24 samples, we determined that we didn't have the sufficient sample size to evaluate our methods on estimating effect sizes of predictors, or models with more than just an intercept term. For instance, even if we wanted to look at the performance of our methods on estimating a group effect with only four samples in each group, each held-out set would only have eight samples in each group. Thus, the ML estimates of the held-out sets would have a lot of variance and could be far from the truth.

Computational performance of Apegglm

To evaluate the computational performance of our package on larger datasets, we simulated allelic counts for 5000 genes and 100 samples, and randomly divided the samples into differing numbers of groups. apegglm with our improvements had very fast running times for both ML and apegglm estimation and scaled well with the number of covariates (see Figure 4 and Figure 5).

Estimation times per gene for ML estimation was substantially faster for apegglm than all other packages (Figure 4). The next best package, aods3, took 5 to 11 times longer than apegglm and did not scale as well with the number of groups. Furthermore, the aods3, gamlss and HRQoL packages occasionally produced errors and could not fit beta-binomial models for all the simulated genes.

For estimating all genes in the simulation via maximum likelihood, apegglm took 24 seconds for two groups and added only 1–2 seconds of computational time for every group added (Figure 5a). The next fastest package that could fit beta-binomial models for all the genes, aod, took seven times longer for two groups and grew 80 times as much for every group added. Comparisons in apegglm estimation between our improved apegglm package and the original package gave similar conclusions. Furthermore, unlike the new apegglm package, which grew roughly linearly with the number of groups in the range we assessed, the order of growth from the original package was not linear: the greater the number of groups already in the model, the greater the computational time increased for adding additional groups. At 10 groups, our improvements made apegglm 27 times faster than aod for ML estimation and 33 times faster than the old package for apegglm estimation. Our improvements also performed quite favorably when fitting beta-binomial models with two groups and two numerical controls. Elapsed time was 31 seconds for ML estimation and 43 seconds for apegglm estimation with the new apegglm package. In contrast, ML estimation took over nine minutes for aod and apegglm estimation took over seven minutes for the old apegglm package. Introducing multicollinearity into the design matrix did not substantially change computational performance for any package (results not shown).

Discussion

Here the performance of three estimators was compared across two simulations and one real dataset of allele specific expression

Figure 4. Comparisons in estimation time for one gene (in milliseconds).

Figure 5. Comparisons in estimation time for all genes. **a)** computational time of ML estimation (in seconds) for the `apeglm` and `aod` packages by the number of groups (covariates). **b)** computational time of `apeglm` estimation for the new and old `apeglm` packages by number of groups (covariates).

in mice. Though `apeglm` was not the best estimator in all cases, it was the most robust and with consistent performance. `Apegglm` had smaller mean absolute error and greater concordance at the top than the MLE, and was never much worse than `ash` in these respects. `Ash` also performed better than the MLE for the simulated data for most metrics, including mean absolute error and concordance at the top. Moreover,

`ash` had higher concordance at the top than `apeglm` in the Student's *t* simulation. However, `ash` also had a tendency to over-shrink some genes, shrinking some truly large effects close to zero. Furthermore, for the real data set, `ash` performed worse than the MLE for most metrics, including mean absolute error and concordance at the top, most likely due to over-shrinking of many genes. As performance on the real data set was based on

taking random subsamples of mice and using the MLE of the held-out set as the truth, estimation error of the held-out effect estimates may have biased results. For future research, using larger data sets to analyze apegglm performance than that of Crowley *et al.* would allow for held-out sets with more samples and thus reduce estimation error of held-out effect size estimates.

The shrinkage estimators compared here typically shrunk only low-count genes, as low-count genes tend to be those with the most uncertain and variable estimates. However, during a simulation where extreme overdispersion and heavy tails of the distribution of true effects were introduced, there were some large-count highly-variable genes that were shrunk as well, showing that ash and apegglm will shrink large-count genes if there is high uncertainty in the estimates. Ash consistently had more extreme shrinkage than apegglm and greater estimation error among genes with truly large effects. Thus, ash would most likely perform best in a situation where most effects were small, such as in the Student's *t* simulation.

No method gave confidence or credible intervals with the highest coverage rates for all scenarios. However, across both simulations and analysis of the mouse data, differences in coverage rates between the three methods were small, and coverage rates for apegglm credible intervals in particular were always very close to the interval that had the largest coverage. Furthermore, interval width for apegglm and ash were always smaller than that of maximum likelihood. This suggests that interval estimates from apegglm could be advantageous over those by maximum likelihood. For future research, it would be beneficial to evaluate the accuracy of hypotheses tests based on the estimates or posterior distribution of apegglm using metrics such as type I and type II error. The method of Leòn-Novelo *et al.* 2018⁷ rejected hypotheses based on credible intervals of its posterior distribution, and if a similar step was taken for apegglm, its narrower intervals and robust coverage could potentially give more powerful hypothesis tests without suffering from inflated type I error.

Our changes to the apegglm package greatly improved computational performance for both ML and apegglm estimation of beta-binomial GLMs, particularly when larger numbers of covariates were involved. Among the R packages that we looked at which could fit beta-binomial models, the new apegglm package was always the fastest for fitting many GLMs in sequence, e.g. across many genes or variant locations. Thus, the new apegglm package is useful for quick and reliable analyses of allelic imbalance even for researchers who wish to only use likelihood-based estimators. Moreover, only coefficient estimates are currently calculated in C++, and even better computational performance would be achieved if overdispersion and standard error calculations were integrated into C++ as well. We are not aware of any other R packages that utilize faster programming languages such as C or C++ to estimate numerous beta-binomial regression models based on large matrices of observed allelic counts. The most similar package we noted was fastglm²⁷, which fits individual

quasi-binomial models in C++. While quasi-binomial models also estimate proportions and control for overdispersion, they do so in a different manner and with different assumptions.

Based on previous work, there are several ways in which the apegglm methodology could potentially be improved for allelic expression studies. For instance, while our extension of apegglm estimated overdispersion by MLE, the original methodology for apegglm as applied to negative binomial GLMs utilized Bayesian estimates for overdispersion as well as for regression coefficients. Introducing a prior for beta-binomial overdispersion that pools information across genes may lead to better estimation and inference of regression coefficients. We also assumed that the total allele-specific counts were fixed and known. Allowing such quantities to be random, as in the method by Leòn-Novelo *et al.* 2018, may lead to better inference as well. Adjusting for read mapping biases and ambiguities (Leòn-Novelo *et al.* 2014⁵; Leòn-Novelo *et al.* 2018⁷; Raghupathy *et al.* 2018³) could also lead to better estimates when such biases and quantification uncertainty are present. Lastly, though here we focused on beta-binomial GLMs, a wide variety of statistical models can be used for ASE, from quasi-binomial²⁸ to Poisson-lognormal models²⁹.

Data availability

Underlying data

Zenodo: RNA-seq Dataset from Crowley *et al.* 2015. <http://doi.org/10.5281/zenodo.3404689>¹⁶.

This project contains the following underlying data:

- fullGeccoRnaDump.csv

This file contains the Crowley *et al.* mouse dataset which was obtained from <http://csbio.unc.edu/gecco/data/fullGeccoRnaDump.csv.gz>^{15,30}. We uploaded the dataset to Zenodo on the authors' behalf with their permission, due to the fact that the original dataset is not currently hosted in a stable repository.

The dataset from this repository is available under the terms of the [Creative Commons Attribution 4.0 International license](#) (CC-BY 4.0).

Extended data

Zenodo: Supplementary Material for Zitovsky and Love 2019. <http://doi.org/10.5281/zenodo.3404697>¹³.

This project contains the following extended data:

- Supplementary Methods.pdf (Contains the mathematical and algorithmic details of how the apegglm package estimates beta-binomial coefficient effect sizes by maximum likelihood and apegglm, including the steps taken to improve computational performance, increase numerical stability and prevent convergence issues)
- Supplementary Figures and Tables.pdf (Contains supplementary figures 1–6 and supplementary tables 1–3. These figures and tables were referenced and described in the main body of the article)

Data are available under the terms of the [CC-BY 4.0](#) license.

Software availability

Zenodo: Apeglm v1.7.5 Source Code. <http://doi.org/10.5281/zenodo.3404504>³¹. This repository contains the source code for the version of the apegglm package used in this paper.

The software from this repository is available under the terms of the [GNU General Public License v3.0](#) (GPL-3).

Zenodo: Source Code for Zitovsky and Love 2019. <http://doi.org/10.5281/zenodo.3404669>³². This repository contains the R scripts used to run the analyses described in this article and generate all of its figures. All figures associated with this paper, including figures present in the main article and supplementary figures, were generated as separate png and .eps files and can also be found in this repository. The R scripts can

be found under the 'Code' folder while the figures can be found under the 'Figures' folder.

Material from this repository are available under the terms of the [GPL-3](#) license.

apeglm is available as part of the Bioconductor project³³ at <http://bioconductor.org/packages/apeglm>. The vignette¹⁸ and manual provide detailed information on how to use the package.

Acknowledgements

We thank Anqi Zhu and Joseph G. Ibrahim of the Department of Biostatistics at UNC Chapel Hill for their contributions to the conceptualization and development of the original apegglm methodology, and Rob Patro for useful discussions.

References

- Castel SE, Levy-Moonshine A, Mohammadi P, *et al.*: **Tools and best practices for data processing in allelic expression analysis.** *Genome Biol.* 2015; **16**(1): 195. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Sun W, Hu Y: **Mapping of Expression Quantitative Trait Loci Using RNA-seq Data.** In: Somnath Datta and Dan Nettleton, editors, *Statistical Analysis of Next Generation Sequencing Data*. Springer International Publishing, Switzerland. 2014; 145–168. [Publisher Full Text](#)
- Raghupathy N, Choi K, Vincent MJ, *et al.*: **Hierarchical analysis of RNA-seq reads improves the accuracy of allele-specific expression.** *Bioinformatics.* 2018; **34**(13): 2177–84. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Turro E, Su SY, Gonçalves A, *et al.*: **Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads.** *Genome Biol.* 2011; **12**(2): R13. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- León-Novelo LG, McIntyre LM, Fear JM, *et al.*: **A flexible Bayesian method for detecting allelic imbalance in RNA-seq data.** *BMC Genomics.* 2014; **15**(1): 920. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Skelly DA, Johansson M, Madeoy J, *et al.*: **A powerful and flexible statistical framework for testing hypotheses of allele-specific gene expression from RNA-seq data.** *Genome Res.* 2011; **21**(10): 1728–37. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- León-Novelo LG, Gerken AR, Graze RM, *et al.*: **Direct Testing for Allele-Specific Expression Differences Between Conditions.** *G3 (Bethesda).* 2018; **8**(2): 447–460. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Love MI, Huber W, Anders S: **Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2.** *Genome Biol.* 2014; **15**(12): 550. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Landau W, Niemi J, Nettleton D: **Fully Bayesian analysis of RNA-seq counts for the detection of gene expression heterosis.** *J Am Stat Assoc.* 2018; **114**(526): 610–621. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Stephens M: **False discovery rates: a new deal.** *Biostatistics.* 2017; **18**(2): 275–94. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Zhu A, Ibrahim JG, Love MI: **Heavy-tailed prior distributions for sequence count data: removing the noise and preserving large differences.** *Bioinformatics.* 2018; **35**(12): 2084–2092. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- R Core Team: **R: A Language and Environment for Statistical Computing.** R Foundation for Statistical Computing, Vienna, Austria. 2018. [Reference Source](#)
- Zitovsky JP, Love MI: **Supplementary Material for Zitovsky and Love 2019.** (Version v1.0). Zenodo. 2019. <http://www.doi.org/10.5281/zenodo.3404697>
- Lu M, Stephens M: **Empirical Bayes Estimation of Normal Means, Accounting for Uncertainty in Estimated Standard Errors.** 2019; arXiv:1901.10679. [Reference Source](#)
- Crowley JJ, Zhabotynsky V, Sun W, *et al.*: **Analyses of allele-specific gene expression in highly divergent mouse crosses identifies pervasive allelic imbalance.** *Nat Genet.* 2015; **47**(4): 353–360. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Crowley JJ, Zitovsky JP, Love MI: **RNA-seq Dataset from Crowley *et al.* 2015.** (Version v1.0). Zenodo. 2019. <http://www.doi.org/10.5281/zenodo.3404699>
- Bolker B: **emdbook: Ecological Models and Data in R.** In: R package version 1.3.11. 2019. [Reference Source](#)
- Zhu A, Ibrahim JG, Love MI: **Effect Size Estimation with Apeglm.** Bioconductor. 2019. [Reference Source](#)
- Himes BE, Jiang X, Wagner P, *et al.*: **RNA-Seq transcriptome profiling identifies CRISPLD2 as a glucocorticoid responsive gene that modulates cytokine function in airway smooth muscle cells.** *PLoS One.* 2014; **9**(6): e99625. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Irizarry RA, Warren D, Spencer F, *et al.*: **Multiple-laboratory comparison of microarray platforms.** *Nat Methods.* 2005; **2**(5): 345–350. [PubMed Abstract](#) | [Publisher Full Text](#)
- Lesnoff M, Lancelot R: **aod: Analysis of Overdispersed Data.** R package version 1.3.3. 2012.
- Yee TW: **Vector Generalized Linear and Additive Models: With an Implementation in R.** R package version 1.1. 2019. [Publisher Full Text](#)
- Lesnoff M, Lancelot R: **aods3: Analysis of Overdispersed Data Using S3 Methods.** R package version 0.4-1.1. 2018. [Reference Source](#)
- Rigby RA, Stasinopoulos DM: **Generalized Additive Models for Location, Scale and Shape.** *J R Stat Soc C-Appl.* 2005; **54**(3): 507–54. [Publisher Full Text](#)
- Dae-Jin L, Najera-Zuloaga J, Arostegui I: **HRQoL: Health Related Quality of Life Analysis.** R package version 1.0. 2017. [Reference Source](#)
- Mersmann O: **microbenchmark: Accurate Timing Functions.** R package version 1.4-6. 2018.
- Huling J: **fastglm: Fast and Stable Fitting of Generalized Linear Models using RcppEigen.** R package version 0.0.1. 2019. [Reference Source](#)
- McVicker G, van de Geijn B, Degner JF, *et al.*: **Identification of genetic variants that affect histone modifications in human cells.** *Science.* 2013; **342**(6159): 747–749. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Alvarez-Castro I: **Bayesian Analysis of High-Dimensional Count Data.** PhD dissertation, Iowa State University. 2017. [Publisher Full Text](#)
- Crowley JJ, *et al.*: **Gene Expression in the Collaborative Cross.** (and Others).

2015. [Data set].
31. Zhu A, Zitovsky J, Ibrahim J, *et al.*: **Apeglm v1.7.5 Source Code (Version v1.0)**. *Zenodo*. 2019. <http://www.doi.org/10.5281/zenodo.3404504>
32. Zitovsky JP, Love MI: **Source Code for Zitovsky and Love 2019 (Version v1.3)**. *Zenodo*. 2019. <http://www.doi.org/10.5281/zenodo.3404669>
33. Huber W, Carey VJ, Gentleman R, *et al.*: **Orchestrating high-throughput genomic analysis with Bioconductor**. *Nat Methods*. 2015; **12**(2): 115–121. [PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)

Open Peer Review

Current Peer Review Status: ? ? ?

Version 1

Reviewer Report 10 February 2020

<https://doi.org/10.5256/f1000research.23018.r57280>

© 2020 Turro E. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Ernest Turro

¹ Department of Hematology, University of Cambridge, Cambridge, UK

² MRC Biostatistics Unit, University of Cambridge, Cambridge, UK

This paper has two components:

1) An advance in computational efficiency for estimating beta-binomial regression coefficients with shrinkage. The authors have produced a C++ implementation of the inference code previously written in R. Both versions of the code are implemented in the `apeglm` R package.

2) An application of this new implementation of their method to the task of inferring allele-specific expression (ASE) and an assessment of its statistical performance in relation to two alternative approaches (ash and MLE).

As the authors start the paper by discussing ASE, rather than computational inference for shrinkage models, it is not immediately apparent that the innovation presented in this paper is computational rather than statistical. Distinguishing these two components clearly would make it more readily apparent that the paper does not present a novel statistical method.

The modelling of ASE has important facets that the authors do not discuss in the introduction (page 3) but which other (uncited) methods have addressed. For example, in a given sample, a gene may contain multiple heterozygous variants (potentially with uncertain phasing of alleles). Each heterozygous variant could overlap different sets of isoforms, each of which may have different levels of ASE. This phenomenon is modelled by the MMDIFF method (Turro *et al*, 2014, *Bioinformatics*¹), for example. The authors should acknowledge this (unmodelled) complication in ASE and explain how they summarise allele-specific count data across multiple variants (e.g., SNPs or indels, which are possibly unphased) within genes to obtain the count pairs modelled by the beta-binomial shrinkage estimators.

The authors have performed several simulation studies and an analysis of a real ASE dataset. Both shrinkage estimators outperform MLE in the simulation studies. However, `apeglm` and MLE do approximately equally well in the real data set and both outperform ash by a significant margin. In

addition, filtering of genes with low allele-specific read counts improves the MLE in the simulation studies but it does not do so in the real data analysis. This discordance demonstrates that the real data are very dissimilar from the simulated data. Although I don't think a major rewrite is warranted, if the authors could demarcate the computational advance (which can be demonstrated by simulation studies that are not representative of ASE, as the authors have done) from the specific application to ASE (using a real data set and perhaps a more faithful simulation study), the striking difference in performance shown in Figures 1-3 would be less incongruous.

In the introduction, the inability of other methods to model the effects of continuous covariates or estimate differences in allelic imbalance between groups (this is not the case though, see MMDIFF) is highlighted and contrasted with the proposed method. However, the authors' own analysis of real data only uses an intercept model. It would be desirable to demonstrate the flexibility afforded by the proposed approach.

In the assessment of statistical performance using the real data set, the MLEs obtained from the held-out data are treated as truth, even though earlier in the paper the authors demonstrate that MLEs have a particularly high mean absolute error. Presumably, this is the case (for genes with relatively low counts) even when the sample size is 18. The authors should consider alternative measures of performance that do not have this drawback.

Minor comments:

- p3: "estimates for allelic expression proportions can be highly variable" - estimates are fixed, the authors should write "estimators".
- p3: a cancer dataset may not be the best choice of example to refer to the proportion of genes with allele-specific reads, due to the prevalence of somatic mutations.
- p3: when discussing filtering as a "remedy" perhaps explain that this achieves a boost in specificity at the cost of power.
- p3: "the most robust and reliable when dealing with small sample sizes" - this part of the sentence does not follow from the previous part, as there is no mention of ash's inadequacy.
- p3: "also introduced new source code" - it is not clear what the "also" refers to.
- p4: "the probability that counts for a particular gene belong to a particular allele" should be changed to "the probability that a read for a particular gene belongs to a particular allele" as the total "counts" will not be assigned to an allele as a block (the total counts derive from a heterogeneous mixture of reads from the two different alleles).
- p4: more information should be given about how the scale parameter of the Cauchy prior is "estimated by pooling information across genes".
- p4: the placement of the $\cdot$ indexing the bold face β is unusual, as the j subscript corresponds to the first rather than the second index.
- p9: rerunning the simulation study with 4 v 4 samples having run it with 5 v 5 samples

seems unnecessary, as such a small change in sample size is unlikely to alter the conclusions.

- p9: "Figure 1d" should read "Figure 3d".

References

1. Turro E, Astle WJ, Tavaré S: Flexible analysis of RNA-seq data using mixed effects models. *Bioinformatics*. 2014; **30** (2): 180-8 [PubMed Abstract](#) | [Publisher Full Text](#)

Is the rationale for developing the new method (or application) clearly explained?

Partly

Is the description of the method technically sound?

Yes

Are sufficient details provided to allow replication of the method development and its use by others?

Partly

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Biostatistics, genomics.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 01 Nov 2020

Josh Zitovsky, University of North Carolina at Chapel Hill, Chapel Hill, USA

This paper has two components:

1) An advance in computational efficiency for estimating beta-binomial regression coefficients with shrinkage. The authors have produced a C++ implementation of the inference code previously written in R. Both versions of the code are implemented in the *apeglm* R package.

2) An application of this new implementation of their method to the task of inferring allele-specific expression (ASE) and an assessment of its statistical performance in relation to two alternative approaches (ash and MLE).

As the authors start the paper by discussing ASE, rather than computational inference for shrinkage models, it is not immediately apparent that the innovation presented in this paper is computational rather than statistical. Distinguishing these two components clearly would make it more readily apparent that the paper does not present a novel statistical method.

We feel that the manuscript title referencing “software”, the abstract mentioning “we evaluated the accuracy of three different estimators” and “We also wrote C++ code to quickly calculate ... apegglm estimates”, the citation of the apegglm publication in the Introduction (“To this end, we look at three different estimation methods... approximate posterior estimation of GLM coefficients (apeglm)¹¹”, and the note about the software in the Introduction (“We also introduced new source code for the apegglm package”) make it clear that the apegglm shrinkage method is not proposed as novel in this manuscript.

The modelling of ASE has important facets that the authors do not discuss in the introduction (page 3) but which other (uncited) methods have addressed. For example, in a given sample, a gene may contain multiple heterozygous variants (potentially with uncertain phasing of alleles). Each heterozygous variant could overlap different sets of isoforms, each of which may have different levels of ASE. This phenomenon is modelled by the MMDIFF method (Turro *et al*, 2014, Bioinformatics¹), for example. The authors should acknowledge this (unmodelled) complication in ASE and explain how they summarise allele-specific count data across multiple variants (e.g., SNPs or indels, which are possibly unphased) within genes to obtain the count pairs modelled by the beta-binomial shrinkage estimators.

We thank the reviewer for bringing up this concern. Here we have focused exclusively on observed allelic counts, ignoring uncertainty of reads that align to both alleles and aggregation of read information across SNPs within a gene. Such data could feasibly be acquired with longer reads that are approaching the transcript length, but in general we agree this as a limitation of our manuscript. We have now added the following to our manuscript to address this unmodelled complication:

*“The methods and performance benchmarks we focus on here address issues stemming from low-count genes and small sample sizes. There are other important concerns in allele-specific analysis of short read RNA-seq datasets, such as reference allele bias, but we do not address such problems here and the methods discussed cannot directly account for them. Our simulation does not involve reference allele bias, and the RNA-seq study we examine took specific measures to avoid reference allele bias. For methods and analysis concerns involving reference allele bias, see Turro *et. al.*⁴ and Castel *et. al.*¹.”*

The authors have performed several simulation studies and an analysis of a real ASE dataset. Both shrinkage estimators outperform MLE in the simulation studies. However, apegglm and MLE do approximately equally well in the real data set and both outperform

ash by a significant margin. In addition, filtering of genes with low allele-specific read counts improves the MLE in the simulation studies but it does not do so in the real data analysis. This discordance demonstrates that the real data are very dissimilar from the simulated data. Although I don't think a major rewrite is warranted, if the authors could demarcate the computational advance (which can be demonstrated by simulation studies that are not representative of ASE, as the authors have done) from the specific application to ASE (using a real data set and perhaps a more faithful simulation study), the striking difference in performance shown in Figures 1-3 would be less incongruous.

We thank the reviewer for pointing out that the simulation and real data results may have been seen as contradicting each other. Based on concerns voiced by other reviewers and our own investigations, we have determined that the issue is not in the simulations, but rather in the real data analyses. Specifically, when benchmarking our methods on the real data set, we had treated the ML estimates from a held-out set as the truth, but as the held-out set only contains 18 samples, the inherent instability and estimation variance present in ML estimators could still present an issue in the accuracy of these estimates. In other words, it may not be reasonable to expect that these ML estimates are close to the true effect sizes, and treating them as such could bias results in favor of ML estimates and against ash (as ash estimates are further from the MLE than apeglm on average). The real data analyses now have been changed to focus more on qualitative comparisons where the truth need not be known (e.g. extent of shrinkage, estimation variance, etc.), and we have largely left estimation accuracy assessments to the simulations. With these changes in place, the simulation and real data results are no longer incongruous.

In the introduction, the inability of other methods to model the effects of continuous covariates or estimate differences in allelic imbalance between groups (this is not the case though, see MMDIFF) is highlighted and contrasted with the proposed method. However, the authors' own analysis of real data only uses an intercept model. It would be desirable to demonstrate the flexibility afforded by the proposed approach.

Thank you for bringing the Turro, Astle and Tavaré (2014) paper to our attention. We have added a mention of this paper in the Introduction as an example of a Bayesian method that can deal with allelic counts and arbitrary design matrices, and have removed the sentence that mentioned that methods do not exist to perform Bayesian analysis with arbitrary designs.

Moreover, we agree that it would have been useful to showcase our method on more complicated design matrices to demonstrate the flexibility of our method. To this extent, we have extended our analysis of real data to include an application of apeglm and ash to a model with two binary covariates and an interaction. The results are discussed in the last paragraph of the "Sampling from the mouse dataset" subsection of the "Results" section.

In the assessment of statistical performance using the real data set, the MLEs obtained from the held-out data are treated as truth, even though earlier in the paper the authors demonstrate that MLEs have a particularly high mean absolute error. Presumably, this is the case (for genes with relatively low counts) even when the sample size is 18. The authors should consider alternative measures of performance that do not have this drawback.

We agree that treating the held-out MLEs as the truth is problematic and have changed the

analyses of our real data set so that results do not depend on knowledge of the truth. See our previous response detailing this issue.

Minor comments:

- p3: "estimates for allelic expression proportions can be highly variable" - estimates are fixed, the authors should write "estimators".
This typo has been corrected.
- p3: a cancer dataset may not be the best choice of example to refer to the proportion of genes with allele-specific reads, due to the prevalence of somatic mutations.

We now clarify that the TCGA dataset referenced here only used the normal breast tissue samples, not the tumor samples.

- p3: when discussing filtering as a "remedy" perhaps explain that this achieves a boost in specificity at the cost of power.
We have added this explanation as suggested.
- p3: "the most robust and reliable when dealing with small sample sizes" - this part of the sentence does not follow from the previous part, as there is no mention of ash's inadequacy.

We have changed this part of the sentence from "the most robust and reliable" to just "robust and reliable".

- p3: "also introduced new source code" - it is not clear what the "also" refers to.
We have changed this sentence to make it more clear.
- p4: "the probability that counts for a particular gene belong to a particular allele" should be changed to "the probability that a read for a particular gene belongs to a particular allele" as the total "counts" will not be assigned to an allele as a block (the total counts derive from a heterogeneous mixture of reads from the two different alleles).
We have made the suggested change.
- p4: more information should be given about how the scale parameter of the Cauchy prior is "estimated by pooling information across genes".

We have added the mathematical details regarding how the scale parameter is estimated in the Supplementary Methods section.

- p4: the placement of the $\cdot$ indexing the bold face β is unusual, as the j subscript corresponds to the first rather than the second index.

We have made notational changes so that the $\cdot$ appears after the j subscript and not before

- p9: rerunning the simulation study with 4 v 4 samples having run it with 5 v 5 samples seems unnecessary, as such a small change in sample size is unlikely to alter the conclusions.

Another reviewer made a similar comment, and so this result has been removed.

- p9: "Figure 1d" should read "Figure 3d".

The typo has been corrected.

Competing Interests: No competing interests were disclosed.

Reviewer Report 04 February 2020

<https://doi.org/10.5256/f1000research.23018.r58251>

© 2020 Niemi J et al. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Jarad Niemi

Department of Statistics, Iowa State University, Ames, IA, USA

Ignacio Alvarez-Castro

University of the Republic, Montevideo, Uruguay

Is the rationale for developing the new method (or application) clearly explained?

Yes.

- In our work a key issue is bias of allele reads toward a reference genome as explained in Sun and Hu (2014).¹ The authors should mention if this bias is relevant for the applications in this manuscript and, if yes, how the methods deal with the bias.
- The introduction argues against eliminating low count genes, yet the manuscript says "Genes where at least three samples did not have at least 10 counts were removed...Genes without at least one count for both alleles across all individuals were removed...Genes with a marginally significant sex or parent effect were removed." Why the contradiction?

Is the description of the method technically sound?

No.

- While the writing is clear, we generally found the order of content confusing. For example, normal-based CI construction should be explained immediately after point estimation and before competing methods, simulation details, and method comparison metrics. We also found there was a lack of details, some of which was in the Supplementary Material but seemed like it should be included in the main manuscript.

In addition, we have outlined concerns below:

Major concerns:

- It isn't clear how MAE or CI coverage are calculated for the real data. For real data the truth is not known and therefore MAE and coverage cannot be calculated the way they can for the simulated data. Are you calculating MAE and coverage relative to the data? You comment "we are treating the MLE of the held-out set as the truth". Why? The simulation studies

seemed to show this is a relatively poor estimate of the truth.

Minor concerns:

- Please provide some statements for why a beta-binomial model is assumed as opposed to alternative model assumptions, e.g. binomial, normal, Poisson.
- We assume you are assume conditional independence in your beta-binomial likelihood and in your Cauchy distribution for the regression coefficients. If so, this should be stated explicitly, e.g. using "ind" above the tilde.
- How often is ϕ_g estimated to be 500? How important is the value 500? Is this user specifiable in the package?
- It is unclear what is meant by "standard error" in the statement "apeglm provides Bayesian shrinkage estimates based on the mode of the posterior as well as standard errors." Is this the posterior standard deviation? Is it the (asymptotic) standard deviation of the estimator?
- The manuscript states "The scale parameter of the Cauchy prior, γ_j , is estimated by pooling information across genes". How exactly is this computed?
- It seems odd to have the Supplementary Material on a site other than F1000. We're disappointed that the Estimation Procedure in the Supplementary Material is not included in the main body of the manuscript as this seems to be key to the methodology. If not included in the main manuscript, perhaps more specific references, say to equation numbers, could be included in the main manuscript.
- We don't understand the statement "Like apeglm, ash can only shrink estimates for one covariate at a time." Isn't the assumed hierarchical distribution a joint hierarchical distribution, albeit assuming independence, for all regression coefficients? If so, then isn't it jointly shrinking all the estimates? Or is the procedure a step-wise procedure where MLEs are shrunk one-at-a-time?
- It is unclear why a Cauchy distribution is chosen. While a Cauchy distribution has the appealing property that it does not shrink large signals (very much), it generally does little shrinkage to small signals compared to alternative estimators, e.g. Bayesian LASSO (10.1198/016214508000000337, 10.1093/biomet/asp047)^{2,3}, horseshoe (10.1093/biomet/asq017)⁴, point-mass priors (10.1080/01621459.1993.10476353)⁵. In our applications, the true distribution of these regression coefficients often has a large spike around 0 which would suggest using a distribution with more mass than a Cauchy near 0.
- The statement "where $1 \leq j \leq K$ is chosen by the user" is confusing. Does the user specify which predictors have a Cauchy distributions? What exactly is the user choosing?

Are sufficient details provided to allow replication of the method development and its use by others?

Partly.

- One reason to provide code and data are to ensure ability to replicate even if the text is

insufficient. So, ensuring the code is able to be run will provide sufficient details.

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes.

- We also applaud the authors for making their code and data available.

Reviewer 1 addressed this and we did not attempt to evaluate this further.

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Partly.

In the abstract, the article claims:

1. "Apeglm consistently performed better than ML[E] according to a variety of criteria, including mean absolute error (MAE) and concordance at the top (CAT)."

Table 1 and 2 provide supporting evidence for the claim that apegglm has lower MAE than MLE for a variety of simulation scenarios.

Figures 1d and 2d shows apegglm and ash having similar CAT and ahead of the non-filtered MLE approach.

It might be helpful to point out that ash, another shrinkage estimator, also consistently performs better than the MLE.

2. "While ash had lower error and greater concordance than ML on the simulations, it also had a tendency to over-shrink large effects, and performed worse on the real data according to error and concordance."

We guess Figures 1a-c and 2a-c as well as line 4 in Table 1 were the evidence for this comment, but we find these figures extremely hard to interpret. The comment in the text is that "some genes with estimates close to the truth were severely shrunk, and several genes with truly large effects were shrunk to zero.", but it isn't clear that this is undesirable. Just because the truth is non-zero doesn't mean that the data randomly generated from this truth should suggest a non-zero result.

With this being said, we would not be surprised about ash shrinking large signals more than apegglm since the Cauchy distribution (used in apegglm) will shrink large signals less than a normal distribution (used in ash) will, but, as Reviewer 1 points out, there are differences in likelihood and estimation procedure between these two methods which make understanding why differences occur more difficult.

3. "When compared to five other packages that also fit beta-binomial models, the apegglm package was substantially faster, making our package useful for quick and reliable analyses of allelic imbalance."

Figure 4 provides the computational cost comparison and seems to show that `apegm` is faster than `aod`, `aods3`, `gamlss`, `HRQoL`, and `VGAM` under the tested scenario. An alternative version of this figure would provide the ratio of runtimes for these other methods compared to `apegm`. While the current version allows for an understanding of the computation time involved, the main purpose of the figure is in comparison of times.

It does seem a bit odd that the authors compared these packages for computation but not for accuracy. In addition, why is `ash` not included in this comparison?

Other:

Minor issues:

- Once you've defined an acronym, just use it, e.g. CAT.
- Be consistent with acronyms: choose ML or MLE and stick with it.
- Figure 5 seems unnecessary since an argument in this manuscript is to use "shrinkage" estimators rather than un-shrunk MLEs.
- An updated reference for 29. Alvarez-Castro is [10.3934/mbe.2019389](https://doi.org/10.3934/mbe.2019389)⁶
- The beta-binomial is a discrete random variable and thus it has a probability mass function rather than a probability density function.

References

1. Sun W, Hu Y: Mapping of Expression Quantitative Trait Loci Using RNA-seq Data. 2014. 145-168 [Publisher Full Text](#)
2. Park T, Casella G: The Bayesian Lasso. *Journal of the American Statistical Association*. 2008; **103** (482): 681-686 [Publisher Full Text](#)
3. Hans C: Bayesian lasso regression. *Biometrika*. 2009; **96** (4): 835-845 [Publisher Full Text](#)
4. Carvalho C, Polson N, Scott J: The horseshoe estimator for sparse signals. *Biometrika*. 2010; **97** (2): 465-480 [Publisher Full Text](#)
5. George E, McCulloch R: Variable Selection via Gibbs Sampling. *Journal of the American Statistical Association*. 1993; **88** (423): 881-889 [Publisher Full Text](#)
6. Alvarez-Castro I, Niemi J: Fully Bayesian analysis of allele-specific RNA-seq data. *Math Biosci Eng*. 2019; **16** (6): 7751-7770 [PubMed Abstract](#) | [Publisher Full Text](#)

Is the rationale for developing the new method (or application) clearly explained?

Yes

Is the description of the method technically sound?

No

Are sufficient details provided to allow replication of the method development and its use by others?

Partly

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bayesian statistics

We confirm that we have read this submission and believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however we have significant reservations, as outlined above.

Author Response 01 Nov 2020

Josh Zitovsky, University of North Carolina at Chapel Hill, Chapel Hill, USA

Is the rationale for developing the new method (or application) clearly explained?

Yes.

- In our work a key issue is bias of allele reads toward a reference genome as explained in Sun and Hu (2014).¹ The authors should mention if this bias is relevant for the applications in this manuscript and, if yes, how the methods deal with the bias.

Reference allele bias is indeed a potential problem when dealing with allelic counts from RNA-seq. However, the methods we benchmark in the manuscript cannot directly deal with such bias. Our simulation does not involve reference allele bias, and the RNA-seq study we examine took specific measures to avoid reference allele bias. We apologize for not clarifying this before and have added a paragraph at the end of the Introduction explaining these points.

- The introduction argues against eliminating low count genes, yet the manuscript says "Genes where at least three samples did not have at least 10 counts were removed...Genes without at least one count for both alleles across all individuals were removed...Genes with a marginally significant sex or parent effect were removed."

Why the contradiction?

When filtering is done to remove genes with a high variance estimated allelic ratio, it is usually done with a threshold greater than e.g. 10 total counts per gene / one count per allele. Increased filtering may result in a loss of statistical power, when the optimal filtering rule is not known. Our minimal filtering was performed such that the metrics (e.g. error and ranking concordance) represent features for which there is some minimally detectable signal across alleles.

Removing genes with a significant sex or parent effect was done for the purposes of performance analysis, as our analysis involved fitting intercept-only models. We did not want the extra variability induced from sex and/or parent effects in the set of genes used for evaluation.

Is the description of the method technically sound?

No.

- While the writing is clear, we generally found the order of content confusing. For example, normal-based CI construction should be explained immediately after point estimation and before competing methods, simulation details, and method comparison metrics. We also found there was a lack of details, some of which was in the Supplementary Material but seemed like it should be included in the main manuscript.

We have moved the description of how the methods compute CIs as suggested. Moreover, we have added additional details about the estimation methods in both the main manuscript (under the "Estimation Methods" subsection of the "Methods" section) and the Supplementary Material. For example, in the main manuscript, we added more details regarding apeglm's likelihood and prior, estimation of the overdispersion and qualitative differences between apeglm's and ash's methodologies. In the Supplemental material, we added more details regarding estimation of the overdispersion, estimation of the scale of the Cauchy prior and the numerical accuracy of our package.

In addition, we have outlined concerns below:

Major concerns:

- It isn't clear how MAE or CI coverage are calculated for the real data. For real data the truth is not known and therefore MAE and coverage cannot be calculated the way they can for the simulated data. Are you calculating MAE and coverage relative to the data? You comment "we are treating the MLE of the held-out set as the truth". Why? The simulation studies seemed to show this is a relatively poor estimate of the truth.

We thank the reviewer for noting this drawback in our initial submission. Initially, our choice to use the MLE of the held-out set as the truth came from the fact that the ML estimators are consistent and asymptotically efficiency estimators of the regression parameters, and thus if the held-out sets are sufficiently large, the ML estimates will be very close to the truth. However, the held-out set only consists of 18 samples, which in practice may be too small to be useful. We agree with your concerns that many of the same problems of ML estimators that we address in our manuscript, such as instability in the presence of low information, would still be present in the held-out sets. After thinking about this more and conducting additional analysis, we came to the conclusion that even when using as many as 24 samples, the ML estimates are not close enough to the truth for some genes and using them as the truth may bias results.

As a result, we have rewritten the real data analysis section to focus on qualitative assessments that do not require knowledge of the truth, such as differences in nature and extent of shrinkage between apeglm and ash and on estimation variance. Accuracy assessments have been largely left to simulations, where the true parameter values are known. Relatedly, we have changed the simulations so that the intercept is simulated from a standard normal distribution, as opposed to being drawn from ML estimates of intercept-only models fit to the genes of the real data set. The reason for this is similar: we have no reason to believe that the intercept ML estimates are close to the true intercepts, and upon investigation, we found that the distribution of ML estimates had several properties that would not realistically be demonstrated by a distribution of true effect sizes.

Minor concerns:

- Please provide some statements for why a beta-binomial model is assumed as opposed to alternative model assumptions, e.g. binomial, normal, Poisson.

We have added a justification for choosing a beta-binomial distribution to model allelic counts in the second paragraph of the "Estimation Methods" subsection of the "Methods" section.

- We assume you are assume conditional independence in your beta-binomial likelihood and in your Cauchy distribution for the regression coefficients. If so, this should be stated explicitly, e.g. using "ind" above the tilde.

We have made the suggested changes to the notation so that the assumed conditional independence is clearer

- How often is ϕ_g estimated to be 500? How important is the value 500? Is this user specifiable in the package?

It is difficult to give an exact frequency, as how often ϕ is estimated at 500 varies from dataset to dataset. The number of genes in a dataset where no or very little overdispersion is exhibited by the allelic proportions (conditional on the covariates) is roughly the number of times at which ϕ will be estimated at 500 for the dataset. As ϕ approaches infinity, the resulting regression parameter MLEs converge to the MLEs from a binomial distribution. We found that with $\phi=500$, the ML estimates are already quite close to the ML estimates from a model with assumption of a binomial distribution, and setting the maximum above 500 led to only very small differences in the coefficients. However, the user can specify a different maximum (and minimum) than that used in this package as desired. Details have been added to the main manuscript and Supplemental Methods regarding our chosen minimum and maximum.

- It is unclear what is meant by "standard error" in the statement "apeglm provides Bayesian shrinkage estimates based on the mode of the posterior as well as standard errors." Is this the posterior standard deviation? Is it the (asymptotic) standard deviation of the estimator?

It is the posterior standard deviation. We clarified this in the second version.

- The manuscript states "The scale parameter of the Cauchy prior, γ_j , is estimated by pooling information across genes". How exactly is this computed?

We have added this information in the Supplemental Material section

- It seems odd to have the Supplementary Material on a site other than F1000. We're disappointed that the Estimation Procedure in the Supplementary Material is not included in the main body of the manuscript as this seems to be key to the methodology. If not included in the main manuscript, perhaps more specific references, say to equation numbers, could be included in the main manuscript.

All references to the Supplemental Material have been made more specific, and are now references to the specific section of the Supplemental Material that is relevant.

- We don't understand the statement "Like apegm, ash can only shrink estimates for one covariate at a time." Isn't the assumed hierarchical distribution a joint hierarchical distribution, albeit assuming independence, for all regression coefficients? If so, then isn't it jointly shrinking all the estimates? Or is the procedure a step-wise procedure where MLEs are shrunk one-at-a-time?

We apologize if this was not clear in the first version of the manuscript and have added clarifications in the new version of the manuscript and Supplemental Material. In summary, apegm for allelic counts assumes a Beta-binomial likelihood for all regression coefficients, but it only assumes a Cauchy prior for one regression coefficient at a time (more specifically, the

regression coefficients for only one covariate, across all genes). Thus only one covariate is being “shrunk” at a time. If Bayesian shrinkage of two coefficients was desired (for example), you would have to run *apeglm* twice: the first time choosing one coefficient, and the second time choosing the other.

- It is unclear why a Cauchy distribution is chosen. While a Cauchy distribution has the appealing property that it does not shrink large signals (very much), it generally does little shrinkage to small signals compared to alternative estimators, e.g. Bayesian LASSO (10.1198/016214508000000337, 10.1093/biomet/asp047)^{2,3}, horseshoe (10.1093/biomet/asq017)⁴, point-mass priors (10.1080/01621459.1993.10476353)⁵. In our applications, the true distribution of these regression coefficients often has a large spike around 0 which would suggest using a distribution with more mass than a Cauchy near 0.

*Our choice of a Cauchy prior was guided by the fact that a Cauchy prior tends to shrink large effect sizes less than other priors, and in a differential expression context was shown to produce estimates with lower error and better ranking by size than competing estimators (see reference 11). We agree that there are situations where a Cauchy prior would not be ideal, if sparsity of estimated coefficients (setting to exactly zero for certain genes) was desired for selection purposes. However *apeglm* follows and cites the *ashr* publication in providing the false sign rate (FSR) as a criterion for gene selection. A justification of our choice of a Cauchy prior and the flexibility of our software to handle other priors has also been added into the manuscript.*

- The statement “where $1 \leq j \leq K$ is chosen by the user” is confusing. Does the user specify which predictors have a Cauchy distributions? What exactly is the user choosing?

This is exactly right: The user is specifying which predictor (singular) is assumed to follow a Cauchy distribution for the purpose of shrinkage estimation. We have tried to make this clearer in the second version of the manuscript. See two responses above.

Are sufficient details provided to allow replication of the method development and its use by others?

Partly.

- One reason to provide code and data are to ensure ability to replicate even if the text is insufficient. So, ensuring the code is able to be run will provide sufficient details.

We apologize for the reproducibility issues present in the first part of the paper. A detailed explanation of the problems and our fixes was given in our responses to the first reviewer. We believe all previous issues have been fixed and the code should now run without problems (assuming all of the relevant packages are installed and the right package versions are being used).

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes.

- We also applaud the authors for making their code and data available.

Reviewer 1 addressed this and we did not attempt to evaluate this further.
Please see our response to your concern under “Are sufficient details provided to allow replication

of the method development and its use by others?”.

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Partly.

In the abstract, the article claims:

1. "Apegglm consistently performed better than ML[E] according to a variety of criteria, including mean absolute error (MAE) and concordance at the top (CAT)."

Table 1 and 2 provide supporting evidence for the claim that apegglm has lower MAE than MLE for a variety of simulation scenarios.

Figures 1d and 2d shows apegglm and ash having similar CAT and ahead of the non-filtered MLE approach.

It might be helpful to point out that ash, another shrinkage estimator, also consistently performs better than the MLE.

Due to changes in the simulations (see our response to your "Major Concern" under "Is the description of the method technically sound?"), ash no longer performs better than maximum likelihood universally, though in general it still performs better. The abstract has been changed to accommodate the different results. We believe that our new abstract provides a succinct yet comprehensive and accurate summary of the new results.

1. "While ash had lower error and greater concordance than ML on the simulations, it also had a tendency to over-shrink large effects, and performed worse on the real data according to error and concordance."

We guess Figures 1a-c and 2a-c as well as line 4 in Table 1 were the evidence for this comment, but we find these figures extremely hard to interpret. The comment in the text is that "some genes with estimates close to the truth were severely shrunk, and several genes with truly large effects were shrunk to zero.", but it isn't clear that this is undesirable. Just because the truth is non-zero doesn't mean that the data randomly generated from this truth should suggest a non-zero result.

With this being said, we would not be surprised about ash shrinking large signals more than apegglm since the Cauchy distribution (used in apegglm) will shrink large signals less than a normal distribution (used in ash) will, but, as Reviewer 1 points out, there are differences in likelihood and estimation procedure between these two methods which make understanding why differences occur more difficult.

Reviewer 1 voiced similar concerns, and you can see our detailed response to this concern in our responses to the first reviewer. To summarize, we have removed results of mean absolute error stratified by the true effect sizes. We also look more at subsets chosen based only on observed data (e.g. total allele counts and MLE size) to interpret results. We hope our new results are easier to interpret and our conclusions more convincing.

1. "When compared to five other packages that also fit beta-binomial models, the apegglm package was substantially faster, making our package useful for quick and

reliable analyses of allelic imbalance."

Figure 4 provides the computational cost comparison and seems to show that apegglm is faster than aod, aods3, gamlss, HRQoL, and VGAM under the tested scenario. An alternative version of this figure would provide the ratio of runtimes for these other methods compared to apegglm. While the current version allows for an understanding of the computation time involved, the main purpose of the figure is in comparison of times.

It does seem a bit odd that the authors compared these packages for computation but not for accuracy. In addition, why is ash not included in this comparison?

We have changed the Figure as suggested to better illustrate relative performance of the other packages compared to apegglm. Moreover, we have added comparisons of numerical accuracy to the main manuscript (last paragraph of "Computational performance of apegglm" subsection) and Supplemental Material. Our package is more numerically accurate and reliable than other packages compared. As to why ash is not included in the comparison, this is because ash requires a vector of initial parameter estimates and standard error estimates, and thus to use ash as we do in the manuscript, one has to perform ML estimation first, and then use ash to shrink the estimates. Comparing ash to apegglm or the ML-fitting packages would thus not be a same-to-same comparison.

Other:

Minor issues:

- Once you've defined an acronym, just use it, e.g. CAT.

We have made the suggested changes to the manuscript.

- Be consistent with acronyms: choose ML or MLE and stick with it.

We have made the suggested changes to the manuscript.

- Figure 5 seems unnecessary since an argument in this manuscript is to use "shrinkage" estimators rather than un-shrunk MLEs.

Though our previous analysis showed that apegglm has higher accuracy than ML estimators, there are still reasons why one would prefer likelihood-based beta-binomial GLMs, such as if the sample size is large or if simplicity or unbiasedness is desired. Moreover, many shrinkage estimation packages like ash require a vector of initial ML estimates and standard errors. Finally, apegglm estimation is almost as fast as ML estimation when using the new apegglm package, and thus Figure 5 would be practically the same if we were to compare other packages to apegglm estimation speed instead. We have added this clarification in the "Computational performance of Apegglm" subsection of the "Results" section.

- An updated reference for 29. Alvarez-Castro is 10.3934/mbe.20193896

The reference has been updated.

- The beta-binomial is a discrete random variable and thus it has a probability mass function rather than a probability density function.

In the new manuscript, we refer to the probability function of the beta-binomial as its "probability mass function" as opposed to a "density function"

Competing Interests: No competing interests were disclosed.

Reviewer Report 17 December 2019

<https://doi.org/10.5256/f1000research.23018.r57281>

© 2019 Stephens M. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Matthew Stephens

¹ Department of Statistics, University of Chicago, Chicago, IL, USA

² Department of Human Genetics, University of Chicago, Chicago, IL, USA

Summary:

The paper presents new implementations of shrinkage methods for beta binomial models, implemented in the R software package `apeglm`. One potential application of these models is estimating allele-specific biases in various sequencing-based assays (and differences in bias between groups), and the paper focuses on this application.

The performance of the shrinkage methods is assessed via simulation and real data analysis (using performance on hold-out data as a performance metric), and the shrinkage methods implemented here are found to be competitive with another shrinkage approach (adaptive shrinkage, `ash`), and consistently outperform the `mle`. The new implementations are also shown to be computationally faster than existing implementations (eg `aod` or the previous version of `apeglm`).

The paper is generally well written, and carefully done, with some exceptions I note later. The new implementations seem likely to be useful in a range of applications. Certainly the use of shrinkage methods in these types of applications is to be encouraged, and I congratulate the authors for leading the way on this. I hope they will find my report helpful in revising their work.

I was instructed "Please indicate clearly which points must be addressed to make the article scientifically sound." I believe points 2-4 below are most important to address to make the article scientifically sound.

1. A note on differences between the shrinkage methods:

One thing that I felt was missing from the paper was a qualitative summary of how the two shrinkage methods used here differ from one another. Both are a form of Empirical Bayes shrinkage, but they use different prior families, different likelihoods, and different point estimate strategies: `apeglm` uses a Cauchy prior, with beta-binomial likelihood, and posterior mode point estimate; whereas `ash` uses a more flexible unimodal prior (which includes Cauchy as a special case), a normal approximation to the likelihood, and uses a posterior mean point estimate. So the trade-off here is that `ash` is using an approximate likelihood, but a more flexible prior and arguably a more principled point estimate (posterior mean is optimal under mean squared error).

I think many readers might benefit from this "high-level" summary of the differences.

Another important point, which will come up later, is that when using `ash` the user has a choice of

how to make the normal approximation. Specifically ash requires the user to provide point estimates ($\hat{\beta}$) and standard errors ($\hat{s}$), with the goal that $\hat{\beta} \approx \sim N(\beta, \hat{s})$, where β is true value that is being estimated.

So there is not only one way to apply ash to a problem, but many different ways depending on the choice of point estimate $\hat{\beta}$. The mle is one natural choice, but in this application there can be problems with infinite mles; see 2. below.

2. On dealing with infinite mles:

To explain the issue with infinite mles, consider first a simple binomial experiment $X \sim \text{Bin}(n, p)$ in which we observe $X=0$. Then the mle for p is 0, and the mle for $\theta = \log(p/(1-p))$ is $-\infty$. Similarly, if $X=n$ the mle for θ is ∞ . Also, in both cases. the standard error for θ is infinite. The same issue arises in the more complex beta-binomial models considered here. Essentially if all the reads in an experiment show the same allele then the mle for the allelic bias parameter (on the logit scale) is $\pm\infty$. This could happen due to low coverage, but it could also happen at high coverage sites if the allelic bias is very strong.

This issue appears to arise in the data analyses used to produce Figure 3 (I did not check whether it arises in the simulations). In Figure 3 there appear many mles (y axis) taking values near $\pm(5$ to $6)$; however, my brief investigations of the data suggested that most of these likely correspond to genes where all the reads come from one allele, and so the mle is actually $\pm\infty$ as above. (That these infinite mles are computed to be near ± 6 is presumably due to an issue with the numerical maximization method used to compute the mle.)

I suspect that the problems with ash observed in Fig 3 stem from this issue: the mle for these situations where all the reads come from one allele are very unstable, and have a very large standard error (technically infinite, although for numeric reasons finite values are used) and these large standard errors cause these mles to be shrunk excessively.

A simple fix for this problem, and one I suggest the authors try, is to add a pseudo-count (say 1, or 0.5) to the counts for *each* allele in the data before computing "mles" and corresponding standard errors.

Pseudo-counts are commonly used to improve stability of mles in this type of situation. Indeed, adding pseudo-counts can be viewed as a simple kind of shrinkage method, so it seems reasonable to compare the more sophisticated EB methods with the simple pseudo-count method. For most genes the point estimates and standard errors will be very little affected by the addition of a small pseudo-count; but for the problematic genes with infinite mle the pseudo-count will stabilize the point estimate and reduce the standard error. I suspect entering the stabilized estimates + standard errors into ash will greatly reduce the problems observed with use of the mles in Figure 3.

(Incidentally, Xing, Carbonetto and Stephens [arXiv:1605.07787¹](#) encounter a closely-related issue when using ash to smooth Poisson data; they solved this using a slightly different approach that is conceptually similar to adding a pseudo-count.)

3. Subsetting results based on shrinkage amounts and "true" values:

In several places the paper reports error measures on subsets of the results. For example, in Table

1 lines 2-4 involve subsets of results chosen based on the true effect size or shrinkage amount (which depends on the true effect). Although tempting, this type of result is hard to interpret. For example, even the optimal shrinkage rule (i.e. the one that uses the correct prior, likelihood and loss function) may not perform uniformly better than the mle on subsets that are chosen in this way. Thus the sentence on p7 ("For instance, among genes with effect sizes greater than two...") may also be true for the optimal shrinkage rule, and so does not constitute direct evidence for "overshrinkage". (I agree there is overshrinkage, but this is not the right way to show it). Comparisons like p9 ("Among genes that were shrunk..."), which stratify by the amount of shrinkage, have the same problem because the amount of shrinkage depends on the true value and not only on the observed value.

It is much cleaner and easier to interpret results if they are subsetting based on the *observed* effect (mle), rather than the true effect. This is because the optimal shrinkage rule is still optimal for *any subset chosen based only on the observed data*. (For this reason you could also subset based on other features of the observed data, like total allele count.) For example, if a method is worse than the mle for the subset of results where the mle is >4 then this is indeed evidence of a problem of some kind.

4. Computation: speed vs accuracy:

When comparing with other methods/implementations there should be some assessment not only of speed, but of accuracy of the different implementations (meaning the accuracy with which they optimize the log-likelihood, rather than the accuracy of the point estimates). Fast answers are easy if you do not care about accuracy....

E.g. I suggest boxplots of $\text{loglik}(\text{method}) - \text{loglik}(\text{apeglm-new})$ for each method, to show that the apeglm-new solution is consistently as high in log-likelihood as other methods (or nearly so). Are there convergence criteria decisions to be made that might affect the trade-off between speed and accuracy?

5. Reproducibility:

I congratulate the authors on making all their code and data available. After a few tweaks to the code I was able to run the code used to produce Figures 1-3. However, my version of Fig 3 looked different from the one in the paper - my figure had different colors and some points seemed to be missing on my figure. I do not know the reasons for this.

Reproducibility would have been made easier by avoiding the use of absolute file paths. I also suggest not defining functions that operate on global variables (e.g. `subsetCalculations = function(sub){...}`) since they are more likely to lead to reproducibility problems.

I was unable to run the code to perform the computation time comparisons (Figure 4), since it errored out. Again I do not know the reason, but it could be due to differences in the package versions I used compared with the authors. I did not have time to troubleshoot this.

6. Miscellaneous other comments:

For Table 3, I think it should be noted that the coverage probability is expected to be <0.95 because you are looking at how often the interval covers the *estimate* in the larger dataset, and not the *true* value. This makes it a hard to compare the methods here because it isn't clear what the right coverage is.

p12: "ash would most likely perform best in a situation where most effects were small". I don't see any evidence for this here (e.g. in the normal simulation ash performs fine) and indeed no reason to expect it to be true a priori. I think this statement should be removed.

7. Minor comments:

- p3: "When a subject is heterozygous for a gene at a particular SNP"; this wording seemed awkward to me.
- p3: "... making it the most robust and reliable when dealing with small sample sizes"; this conclusion ("making it") seemed not to follow directly from the first part of the sentence.
- p4: "Apeglm shrinks the effect of one predictor at a time": I think this sentence might work better at the start of the paragraph, before specifying the prior used.
- p5: "guided by the author's claim": this is not just a claim, it is a theorem dating back to the 1950s (see original paper for citations).
- p5: diallel typo?
- p5: use of beta for the mean of the exponential distribution is confusing as beta is already used elsewhere.
- p9: "We also conducted..." This did not seem worth reporting to me. The difference in sample size (5 vs 5 instead of 4 vs 4) is too small to expect that the results would be very different.
- p9: In the paragraph "Both apeglm and MLE..." the acknowledgement that comparing against CAT in a hold-out set is potentially problematic is a bit buried in the middle of the paragraph. It would seem better to acknowledge this up front. Given the problems with CAT acknowledged here I suggest removing that figure (Fig 3d) or moving to an Appendix.
- Figure 5: this should have a y axis that starts at 0.

References

1. Xing Z, Carbonetto P, Stephens M: Flexible signal denoising via flexible empirical Bayes shrinkage. *arXiv*. 2019. [Reference Source](#)

Is the rationale for developing the new method (or application) clearly explained?

Yes

Is the description of the method technically sound?

Yes

Are sufficient details provided to allow replication of the method development and its use by others?

Partly

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bayesian statistics; statistical genetics

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 01 Nov 2020

Josh Zitovsky, University of North Carolina at Chapel Hill, Chapel Hill, USA

Summary:

The paper presents new implementations of shrinkage methods for beta binomial models, implemented in the R software package *apeglm*. One potential application of these models is estimating allele-specific biases in various sequencing-based assays (and differences in bias between groups), and the paper focuses on this application.

The performance of the shrinkage methods is assessed via simulation and real data analysis (using performance on hold-out data as a performance metric), and the shrinkage methods implemented here are found to be competitive with another shrinkage approach (adaptive shrinkage, *ash*), and consistently outperform the *mle*. The new implementations are also shown to be computationally faster than existing implementations (eg *aod* or the previous version of *apeglm*).

The paper is generally well written, and carefully done, with some exceptions I note later. The new implementations seem likely to be useful in a range of applications. Certainly the use of shrinkage methods in these types of applications is to be encouraged, and I congratulate the authors for leading the way on this. I hope they will find my report helpful in revising their work.

I was instructed "Please indicate clearly which points must be addressed to make the article scientifically sound." I believe points 2-4 below are most important to address to make the

article scientifically sound.

Thank you for your constructive comments and careful evaluation of our software and analysis. We found your report helpful and have tried our best to address all of your concerns. Point-by-point responses are provided below.

1. A note on differences between the shrinkage methods:

One thing that I felt was missing from the paper was a qualitative summary of how the two shrinkage methods used here differ from one another. Both are a form of Empirical Bayes shrinkage, but they use different prior families, different likelihoods, and different point estimate strategies: `apeglm` uses a Cauchy prior, with beta-binomial likelihood, and posterior mode point estimate; whereas `ash` uses a more flexible unimodal prior (which includes Cauchy as a special case), a normal approximation to the likelihood, and uses a posterior mean point estimate. So the trade-off here is that `ash` is using an approximate likelihood, but a more flexible prior and arguably a more principled point estimate (posterior mean is optimal under mean squared error).

I think many readers might benefit from this "high-level" summary of the differences.

Another important point, which will come up later, is that when using `ash` the user has a choice of how to make the normal approximation. Specifically `ash` requires the user to provide point estimates ($\hat{\beta}$) and standard errors ($\hat{s}$), with the goal that $\hat{\beta} \sim N(\beta, \hat{s})$, where β is true value that is being estimated.

So there is not only one way to apply `ash` to a problem, but many different ways depending on the choice of point estimate $\hat{\beta}$. The mle is one natural choice, but in this application there can be problems with infinite mles; see 2. Below.

We agree that there are important methodological differences between the methods, and that a high-level summary of these differences would be beneficial to the readers. We have added a paragraph highlighting these differences in the second-to-last paragraph of the "Estimation methods" subsection of the "Methods" section. Among other differences, we highlight the increased flexibility of `ash`'s prior and its ability to handle non-ML estimators. Additional details regarding the methodology of these methods have also been added to the sections where `apeglm` and `ash` were initially introduced.

2. On dealing with infinite mles:

To explain the issue with infinite mles, consider first a simple binomial experiment $X \sim \text{Bin}(n, p)$ in which we observe $X=0$. Then the mle for p is 0, and the mle for $\theta = \log(p/(1-p))$ is $-\infty$. Similarly, if $X=n$ the mle for θ is ∞ . Also, in both cases, the standard error for θ is infinite. The same issue arises in the more complex beta-binomial models considered here.

Essentially if all the reads in an experiment show the same allele then the mle for the allelic bias parameter (on the logit scale) is $\pm\infty$. This could happen due to low coverage, but

it could also happen at high coverage sites if the allelic bias is very strong.

This issue appears to arise in the data analyses used to produce Figure 3 (I did not check whether it arises in the simulations). In Figure 3 there appear many mles (y axis) taking values near $\pm(5 \text{ to } 6)$; however, my brief investigations of the data suggested that most of these likely correspond to genes where all the reads come from one allele, and so the mle is actually $\pm\text{-Infinity}$ as above. (That these infinite mles are computed to be near ± 6 is presumably due to an issue with the numerical maximization method used to compute the mle.)

I suspect that the problems with ash observed in Fig 3 stem from this issue: the mle for these situations where all the reads come from one allele are very unstable, and have a very large standard error (technically infinite, although for numeric reasons finite values are used) and these large standard errors cause these mles to be shrunk excessively.

A simple fix for this problem, and one I suggest the authors try, is to add a pseudo-count (say 1, or 0.5) to the counts for *each* allele in the data before computing "mles" and corresponding standard errors.

Pseudo-counts are commonly used to improve stability of mles in this type of situation. Indeed, adding pseudo-counts can be viewed as a simple kind of shrinkage method, so it seems reasonable to compare the more sophisticated EB methods with the simple pseudo-count method. For most genes the point estimates and standard errors will be very little affected by the addition of a small pseudo-count; but for the problematic genes with infinite mle the pseudo-count will stabilize the point estimate and reduce the standard error. I suspect entering the stabilized estimates + standard errors into ash will greatly reduce the problems observed with use of the mles in Figure 3.

(Incidentally, Xing, Carbonetto and Stephens arXiv:1605.07787¹ encounter a closely-related issue when using ash to smooth Poisson data; they solved this using a slightly different approach that is conceptually similar to adding a pseudo-count.)

As you suspected, there were indeed genes with "truly infinite" MLEs, but due to numerical reasons, were given finite estimates by the apeglm package. As you suggested, we have now performed additional analyses adding a pseudocount to each allele prior to computing MLEs, and compared the performance of the resulting ML, apeglm and ash estimates to those not involving pseudocounts for the simulations. We also attempted to remove the infinite ML genes prior to analysis. Results can be found in Table 1, Table 2 and Supplementary Figure 3.

3. Subsetting results based on shrinkage amounts and "true" values:

In several places the paper reports error measures on subsets of the results. For example, in Table 1 lines 2-4 involve subsets of results chosen based on the true effect size or shrinkage amount (which depends on the true effect). Although tempting, this type of result is hard to interpret. For example, even the optimal shrinkage rule (i.e. the one that uses the correct prior, likelihood and loss function) may not perform uniformly better than the mle on subsets that are chosen in this way. Thus the sentence on p7 ("For instance, among genes with effect sizes greater than two...") may also be true for the optimal shrinkage rule,

and so does not constitute direct evidence for "overshrinkage". (I agree there is overshrinkage, but this is not the right way to show it). Comparisons like p9 ("Among genes that were shrunk..."), which stratify by the amount of shrinkage, have the same problem because the amount of shrinkage depends on the true value and not only on the observed value.

It is much cleaner and easier to interpret results if they are subsetted based on the *observed* effect (mle), rather than the true effect. This is because the optimal shrinkage rule is still optimal for *any subset chosen based only on the observed data*. (For this reason you could also subset based on other features of the observed data, like total allele count.) For example, if a method is worse than the mle for the subset of results where the mle is >4 then this is indeed evidence of a problem of some kind.

Shrinkage in the first manuscript was defined as the movement of apegglm and ash estimates from the MLE toward zero. As apegglm, ash and ML estimates are all functions of the observed data, the degree of shrinkage is also a function of observed data and thus we felt that subsetting by shrinkage was valid. However, we do agree with your concern that subsetting by true effect sizes may cause difficulty in contrasting procedures with each other with respect to the optimal shrinkage rule, and thus have removed results of mean absolute error stratified by the true effect sizes. Moreover, per your suggestion, we have added stratification of MAE by total gene counts and MLE magnitude. We also added MA plots, which illustrates how the amount of shrinkage differs by total gene counts and MLE size (these plots were previously in the Supplemental Material, but have been moved to the main paper).

4. Computation: speed vs accuracy:

When comparing with other methods/implementations there should be some assessment not only of speed, but of accuracy of the different implementations (meaning the accuracy with which they optimize the log-likelihood, rather than the accuracy of the point estimates). Fast answers are easy if you do not care about accuracy....

E.g. I suggest boxplots of $\loglik(\text{method}) - \loglik(\text{apeglm-new})$ for each method, to show that the apegglm-new solution is consistently as high in log-likelihood as other methods (or nearly so). Are there convergence criteria decisions to be made that might affect the trade-off between speed and accuracy?

We agree that an assessment of numerical accuracy is important in showcasing our package, and have adding such assessments in the new version of the manuscript. We focused our analysis of numerical accuracy on genes such that the difference in an estimated coefficient between apegglm and the other packages were non-negligible (above 0.01), and among those genes reported the differences in log-likelihood. A high-level overview of the results is present in the last paragraph of the "Computational Performance of Apegglm" subsection of the "Results" section, and a detailed summary of the results was added to the Supplementary Methods section. Overall, we found that our package is, in addition to its estimation speed, also numerically accurate.

5. Reproducibility:

I congratulate the authors on making all their code and data available. After a few tweaks to the code I was able to run the code used to produce Figures 1-3. However, my version of Fig 3 looked different from the one in the paper - my figure had different colors and some points seemed to be missing on my figure. I do not know the reasons for this.

Reproducibility would have been made easier by avoiding the use of absolute file paths. I also suggest not defining functions that operate on global variables (e.g. `subsetCalculations = function(sub){...}`) since they are more likely to lead to reproducibility problems.

I was unable to run the code to perform the computation time comparisons (Figure 4), since it errored out. Again I do not know the reason, but it could be due to differences in the package versions I used compared with the authors. I did not have time to troubleshoot this.

We apologize for the reproducibility issues in the first version of the paper. Briefly, the issues you reported stemmed from two underlying causes: 1) the version of the `apeglm` package in the devel branch at the time of publication did not match the version used in the manuscript; 2) we accidentally uploaded the wrong scripts to Zenodo. We have now correctly identified the `apeglm` package version in the manuscript (v1.11.2) and replaced the scripts in Zenodo with the correct ones. All scripts should now run without issues and output the same numbers and plots as shown in the paper. Moreover, we have removed absolute file paths and do not use global variables in our functions (some of the local variables defined within functions might share names with global variables created later on, but our functions no longer call global variables directly).

6. Miscellaneous other comments:

For Table 3, I think it should be noted that the coverage probability is expected to be <0.95 because you are looking at how often the interval covers the **estimate** in the larger dataset, and not the **true** value. This makes it a hard to compare the methods here because it isn't clear what the right coverage is.

Due to concerns posed by yourself and other reviewers, we have completely rewritten our analysis of real data to focus on more qualitative results, and have mostly left evaluations of accuracy to the simulations, where the true simulation parameters are known. Among other changes, we do not evaluate or assess coverage probabilities of estimators when analyzing the real data.

p12: "ash would most likely perform best in a situation where most effects were small". I don't see any evidence for this here (e.g. in the normal simulation ash performs fine) and indeed no reason to expect it to be true a priori. I think this statement should be removed.

We have removed this statement.

7. Minor comments:

- p3: "When a subject is heterozygous for a gene at a particular SNP"; this wording seemed awkward to me.

We have changed the wording to "When a subject is heterozygous at a particular SNP within an exon of a gene"

- p3: "... making it the most robust and reliable when dealing with small sample sizes"; this conclusion ("making it") seemed not to follow directly from the first part of the sentence.

We have changed this from "the most robust and reliable" to just "robust and reliable".

- p4: "Apeglm shrinks the effect of one predictor at a time": I think this sentence might work better at the start of the paragraph, before specifying the prior used.

We have made the suggested change.

- p5: "guided by the author's claim": this is not just a claim, it is a theorem dating back to the 1950s (see original paper for citations).

Apologies for the confusion. We have changed it from "guided by the author's claim" to "guided by the fact" and have cited both as well as the original 1950's citation.

- p5: diallel typo?

In our original manuscript, we had the term "diallel cross", we did not find a typo.

- p5: use of beta for the mean of the exponential distribution is confusing as beta is already used elsewhere.

We changed the notation for the mean parameter from beta to mu.

- p9: "We also conducted..." This did not seem worth reporting to me. The difference in sample size (5 vs 5 instead of 4 vs 4) is too small to expect that the results would be very different.

We have removed this result.

- p9: In the paragraph "Both apegglm and MLE..." the acknowledgement that comparing against CAT in a hold-out set is potentially problematic is a bit buried in the middle of the paragraph. It would seem better to acknowledge this up front. Given the problems with CAT acknowledged here I suggest removing that figure (Fig 3d) or moving to an Appendix.

Please see our response to your concerns in point #6.

- Figure 5: this should have a y axis that starts at 0.

Unfortunately, the y-axis for figure 5 of the initial version of the paper (renamed Figure 8 in version 2) is on the log-scale, which means we cannot start it at zero. Using a log scale is necessary due to the very different computational times of the apegglm and aod packages and the difference in how well they scale with increasing numbers of covariates. We considered changing the figure to start the y-axis at a smaller positive number (eg 10, 1, 0.1 etc.) but we ultimately decided against this as the exact cut-point at which to start the y-axis would have been arbitrary and there would have been a large amount of unnecessary white space between the plots and the x-axis (due to the fact that the y-axis is measured on the log scale).

Competing Interests: No competing interests were disclosed.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research