A single nucleotide polymorphism (SNP) is a single nucleotide
generically to the two alleles in any SNP as alleles.
two possibilities are referred to as
same species. For most SNPs, only two bases are observed. The
Availability: provided by Affymetrix, Birdseed. The main advantage of the new
CRLMM version 2 outperforms CRLMM version 1 and the algorithm
samples. Using three independent datasets, we demonstrate that the
development of quality metrics for SNPs, samples and batches of
are that we now account for variability across batches and improve
the multi-level model used by CRLMM version 1. T wo key differences
© The Author 2009. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oxfordjournals.org
INTRODUCTION

ABSTRACT
Motivation: Genome-wide association studies (GWAS) are used to
discover genes underlying complex, heritable disorders for which
less powerful study designs have failed in the past. The number of
GWAS has skyrocketed recently with findings reported in top
journals and the mainstream media. Microarrays are the genotype
calling technology of choice in GWAS as they permit exploration
of more than a million single nucleotide polymorphisms (SNPs)
simultaneously. The starting point for the statistical analyses used by
GWAS to determine association between loci and disease is making

genotype calls (AA, AB or BB). However, the raw data, microarray
probe intensities, are heavily processed before arriving at these calls.

Various sophisticated statistical procedures have been proposed for
transforming raw data into genotype calls. We find that variability in
microarray output quality across different SNPs, different arrays and
different sample batches have substantial influence on the accuracy
of genotype calls made by existing algorithms. Failure to account for
these sources of variability can adversely affect the quality of findings
reported by the GWAS.

Results: We developed a method based on an enhanced version of
the multi-level model used by CRLMM version 1. Two key differences
are that we now account for variability across batches and improve
the call-specific assessment of each call. The new model permits
the development of quality metrics for SNPs, samples and batches of
samples. Using three independent datasets, we demonstrate that the
CRLMM version 2 outperforms CRLMM version 1 and the algorithm
provided by Affymetrix, Birdseed. The main advantage of the new
approach is that it enables the identification of low-quality SNPs,
samples and batches.

Availability: Software implementing of the method described in this
article is available as free and open source code in the crlmm
R/BioConductor package.
Contact: rafael@jhu.edu
Supplementary information: Supplementary data are available at
Bioinformatics online.

1 INTRODUCTION

A single nucleotide polymorphism (SNP) is a single nucleotide
DNA variation occurring in the genomes of individuals from the
same species. For most SNPs, only two bases are observed. The
two possibilities are referred to as alleles. Typically, one is less
common and is called minor allele. In this article, we will refer
generically to the two alleles in any SNP as alleles A and B.

Because we have have two copies of each autosomal chromosome
(maternal and paternal), there are three possible allele combinations
at each SNP: AA, AB and BB. These are referred to as genotypes.
Variations in the DNA is a subject of great importance as they can
elucidate, for example, how humans develop diseases and respond
to treatments. Association studies enable testing for relationships
between alleles and phenotypes, e.g. disease status. In the past,
association studies would screen through hundreds of SNPs carefully
selected to be near candidate genes. Today, microarray technology
permits the screening of millions of SNPs across the entire genome
and has revolutionized these studies, which are now referred to as
genome-wide association studies (GWAS).

Results from large GWAS, for diseases such as bipolar disorder,
coronary artery disease, Crohn’s disease, hypertension, rheumatoid
arthritis, Types 1 and 2 diabetes (Wellcome Trust Case Control
Consortium, 2007), diabetic nephropathy (Mueller et al., 2006)
and kidney dysfunction (Bash et al., 2009), have received much
attention. During the last two years, we have seen a large increase
of these studies and many more are in the works. Currently, the
typical data analysis procedure is to genotype a large number
(thousands) of cases and controls using microarrays and search for
SNPs that are statistically associated with disease. However, the
process of converting raw intensities into genotype calls consists of
complicated statistical manipulation of noisy data and many

genotype calls are uncertain. A common analysis approach is simply to perform χ^2-tests to evaluate the association of the declared
genotypes and disease, without accounting for uncertainty. As shown
by Ruczinski et al. (http://www.bepress.com/jhubiostat/paper181)
via simulation, the failure to account for genotype uncertainty
properly can produce inefficient or invalid associations. Of course,
a valid quantification of uncertainty is a prerequisite to using it in
association studies and we focus on this aspect.

In Section 2, we outline the statistical problem and describe
previous works on genotyping, focusing on CRLMM version 1,
whose model is the starting point for the findings we report on
this article. In Section 3, we outline the model and describe
estimation procedures. In Section 4, we demonstrate the utility
of our methodology with three datasets. Finally, in Section 5, we
summarize and discuss our findings.

2 CONVERTING RAW INTENSITIES TO
GENOTYPE CALLS

The first step, referred to as preprocessing, converts raw microarray
intensities into quantities proportional to the amount of DNA in the
target sample associated with each alleles A and B for each SNP.
Quantifying uncertainty in genotype calls

We denote these summarized intensities by $I_A$ and call genotypes based on the association. A naive approach to genotyping is to set confidence thresholds for each SNP. This requirement stems from the distribution of intensities $I_A$, into posterior probabilities of the three possible genotypes ($AA$, $AB$ and $BB$) and thereby providing a confidence measure that can be used to decide which calls to omit or to introduce the appropriate genotype uncertainty when assessing association.

A naive approach to genotyping is to set confidence thresholds and call genotypes based on the $I$s being above or below these thresholds. For example, to call an AA genotype one might require that $I_A > I_B$. Unfortunately, the probe effect, described in detail in the microarray literature (Irizarry et al., 2003; Li and Wong, 2001a, b; Naef and Magnasco, 2003; Wu et al., 2004), requires a different cutoff for each SNP. This requirement stems from the fact that the abundance of each SNP allele is measured with different probes, having different sequences and therefore different hybridization properties, resulting in a large SNP to SNP variability in the distribution of intensities $I_A$ and $I_B$ (Fig. 1; color version online).

Competing genotype calling algorithms use different strategies for determining these SNP-specific cutoffs. Many use unsupervised clustering, like the Dynamic Model (DM)-based algorithm (Di et al., 2005) and CHIAMO (Wellcome Trust Case Control Consortium, 2007). The more successful algorithms train on data for which genotypes are known, for example, BRLMM (Affymetrix, 2006), CRLMM version 1 (Carvalho et al., 2007), BRLMM-P (Affymetrix, 2007) and Birdseed (Korn et al., 2008). For most SNPs on these training arrays, we have independent genotype calls for 270 HapMap samples (The International HapMap Consortium, 2003). These calls are based on consensus results from various technologies and are considered a gold standard.

The density of SNP microarrays has increased considerably in the last 5 years, going from a few thousands to roughly one million SNPs per chip. Nowadays it is not uncommon to find GWAS that target thousands of samples simultaneously and genotyping algorithms have gone through major modifications to accommodate such changes. CRLMM version 1 was one example of such developments, as it extended the ideas available at the time to provide means to efficiently account for various batch-related effects, outperforming competing algorithms (Lin et al., 2008). We treat it as the leading genotyping algorithm and use its model as a starting point for our work.

CRLMM uses HapMap calls to define known genotypes, which in turn permit us to define a training set. With the training data in place, Carvalho et al. (2007) describe a supervised learning approach based on a two-stage hierarchical model. Unlike other algorithms, CRLMM models $M \equiv \log_2 (I_A/I_B)$ instead of the intensity pair. This choice makes CRLMM more robust to probe effects because the probe effects of the two allele probes have similar additive effects and so partially cancel. This is demonstrated by Figure 2. To account for a well-described (Affymetrix, 2006, 2007; Carvalho et al., 2007) dependence of $M$ on the overall intensity $S \equiv \log_2 (\sqrt{7MB})$, Carvalho et al. (2007) fit splines using a mixture model and correct the bias with the fitted curves. Then, for a given SNP, the distribution of $M_i$, conditioned on genotype, is modeled as Gaussian. To account for the remaining probe effect, each SNP $i=1, \ldots, I$ has a different mean $\mu_i$ and standard deviation (SD) $\sigma_i$. Sample means and SDs from the training data are used to estimate the $\mu$s and $\sigma$s. However, due to low minor allele frequencies, even this large training dataset

![Fig. 1.](https://academic.oup.com/bioinformatics/article-abstract/26/2/242/209014/1)

![Fig. 2.](https://academic.oup.com/bioinformatics/article-abstract/26/2/242/209014/2)
provides relatively few data points for the rare genotype in some SNPs.

A hierarchical model is used to improve the precision of the model parameters for these SNPs. Carvalho et al. (2007) make use of an empirical Bayes approach in which the means, conditioned on genotype, follow a multivariate normal distribution and the variances follow an inverse gamma distribution. The approach permits CRLMM to borrow strength from other SNPs. To make calls, CRLMM treats the estimated parameters as known and computes posterior probabilities for each genotype given the observed log-ratio $M$. The posteriors are then used as a confidence measure. Lin et al. (2008) found that the confidence measures provided by CRLMM version 1 were not optimal and proposed an ad hoc adjustment based on a training approach. CRLMM version 1 uses these adjusted confidence measures.

The strategies used to train the genotyping algorithm proposed by Carvalho et al. (2007) and the one presented on this article require an expert intervention. Currently, there is no automatic solution for this issue, but extending the algorithm to other platforms is possible, as done by Ritchie et al. (2009) for illumina ininium beadChips.

3 THE MODEL

3.1 Motivation

Procedures used in association studies are highly susceptible to problems due to inaccuracies induced by the genotyping procedures. Genotyping algorithms currently available lack methods to identify SNPs and batches that, if not handled in the early stages of the investigation, are likely to unfavorably affect results.

The current approach to determine association between SNPs and disease is to perform an association test between genotypes and outcomes, e.g. a $\chi^2$-test for discrete outcomes. The SNPs with too low confidence scores are ‘set aside’, but the confidence cutoff is quite arbitrary and can affect results. Importantly, because there is more uncertainty associated with heterozygous calls (AB) than with homozygous calls (AA, BB), specifying a single cutoff for both (the current practice) can lead to bias due to informative missingness. Since CRLMM version 1 and other calling algorithms are model based as are assessments of association, a natural extension is to develop association tests based on genotype probabilities rather than hard calls. Marchini et al. (2007) and Plagnol et al. (2007) use such probability-based calls to combine results across different platforms. Ruczinski et al. (http://www.bepress.com/jhubio/bost/181) demonstrate that using probability-based calls improves the power of GWAS.

The posterior probabilities provided CRLMM version 1 have three crucial limitations:

1. The posteriors are overly optimistic in favor of the genotype attaining the highest probability. The main reason for this is that the actual tails of the conditional distributions of $M$ are longer than predicted by the Gaussian assumption. Figure 3 shows one example in which one observation has posterior of one and is correct, yet, the call is wrong.

2. The statistical uncertainty of estimates from the training step is ignored, resulting in overconfident calls for minor alleles.

3. We have observed that the genotype parameters shift from batch to batch and these batch effects are not in the model used by CRLMM version 1. As a result, batches of questionable quality are not detected by the CRLMM version 1 algorithm.

The third point is particularly troublesome. A logistics problem with these large GWAS is that hybridizations need to be processed in batches. Because DNA samples are stored in 96-well plates and robots make it convenient to run all samples in a plate at once, plates are usuallyconfounded with hybridization times. To make matters worst, it is rarely the case that a GWAS randomizes or controls for plate when storing samples. Therefore, it is common that plate and outcome of interest are at least partially confounded. Therefore, if genotyping algorithms do not appropriately assess these batch effects, it will be difficult if not impossible, to distinguish real from artifactual associations. The new methods presented in this article, successfully detect problematic batches, by simply inspecting some of the estimated model parameters.

To address these deficits, we have developed an enhancement to the model used by CRLMM version 1 that provides much improved posterior probabilities and a powerful probability-based approach to detecting problematic SNPs and batches. We demonstrate that these SNP and batch quality metrics combined with improved confidence scores can effectively identify low-quality elements and significantly improve the accuracy of the genotype calls to be used on downstream analyses.

3.2 The enhanced hierarchical model

We structure our analysis via the hierarchical model,

$$Z_i \ iid \ \text{trinomial} \left( \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right)$$

$$\mu_{ij} | Z_{ij} = g \ iid \ \mathcal{N}(0, \sigma^2_{\mu})$$

$$\lambda_{ij} | \mu_{ij}, Z_{ij} = g \ iid \ \mathcal{N}(0, \lambda_{ij})$$

$$M_{ijk} | \mu_{ij}, \lambda_{ij} = f_{ijk}(g_{ijk}) + \mu_{ij} + \lambda_{ij} + \sigma_{ijk} e_{ijk}$$

$$\epsilon_{ijk} | \mu, \lambda \ iid \ \mathcal{N}(0, \sigma^2_{\epsilon})$$

Index $i = 1, \ldots, I$ represents SNP, $j = 1, \ldots, J$ represents the batch, $k = 1, \ldots, K$ represents the sample and $g = AA, AB$ or BB is the genotype. The $Z$’s are unobserved, true genotypes, the $M$’s are

![Fig. 3. An example of an SNP with three clear clusters: the calls derived from the algorithm are represented by colors (AA: green; AB: orange and BB: violet). The observation with the red circle around it was incorrectly called BB and, under the normal assumption for the residuals, the posterior was 0.999. With the assumption that the residuals follow a t-distribution, the posterior was penalized and reduced to 0.500.](https://example.com/fig3.png)
observed log-ratios, $\mu_j = (\mu_{iAA}, \mu_{iAB}, \mu_{iBB})'$ represents the shifts for SNP $j$, $\lambda_j = (\lambda_{iAA}, \lambda_{iAB}, \lambda_{iBB})'$ denotes batch effects associated to SNP $j$ and batch $j$, $\sigma_{ijg}$ is the SNP-specific variance for genotype $g$ and accounts for the fact that different SNPs have different scales of variation around the predicted cluster centers, the $d_g$ are the degrees of freedom associated to the variance, $2\nu$ of a typical SNP. Both $d_g$ and $\sigma_{ijg}$ are estimated from the training data using the empirical Bayes approach described in Smyth (2004).

Data exploration demonstrates that, for large and small intensity values, $M$ for the AA and BB genotypes are shrunken toward 0 (Fig. 8.1 from Carvalho et al., 2007). As done by Carvalho et al. (2007), we account for this intensity-dependent bias with the deterministic function $f_{jk}(\cdot, g) = 0$ for all $j,k$. Differences across genotypes (e.g. $M$'s for AA are on average larger than $M$'s for AB) are absorbed into $f$. These functions are estimated in a separate step, as described in detail by Carvalho et al. (2007), and are treated as known.

The model used in CRLMM version 1 assumes that $\epsilon_{ijkg}$ is normally distributed. The scale factor $\sigma_{ig}$ is needed because log-ratios for different SNPs present different levels of variation around the predicted region centers. Because, for some genotypes, some SNPs have very few observations, an empirical Bayes approach is applied to borrow information from other SNPs. The prior used for $\sigma_{ig}$ is the inverse-$\chi^2$ with $d_g$ degrees of freedom. This is a convenient prior that provides closed form solutions. Estimation is performed as described by Smyth (2004). Recently, we observed that outliers were common and, as a consequence, CRLMM version 1 confidence scores were overoptimistic. To avoid fitting a different error model to each SNP, we adapted the data analytic approach by changing the distribution of $\epsilon_{ijkg}$ from a standard normal to a $\chi^2$ with 6 degrees of freedom. The model and fitting procedure for the scaling factor were kept the same as CRLMM version 1. We used the training data to arrive at the choice of six and this worked well in the test sets.

Note that if one has an error term and needs to add a scaling factor, in an empirical Bayes context, one needs to estimate its distribution. Regardless of the model choice, the resulting random variable is the ratio (or product) of two others. The use of the inverse-$\chi^2$ distribution is both convenient and effective; other choices that provide sufficient flexibility would produce similar results.

### 3.3 Estimating parameters

Note that model (1) has $I \times (2 + J) \times 3 + 12$ parameters. With $I = 906600$ SNPs, these are too many for a global estimation procedure to be practical. In this section, we describe an effective approximate modular procedure. In the first step, we take advantage of the existing training data to estimate the $\mu$'s. Then, for each new batch $j$, we treat the $\mu$'s as known and estimate the $\lambda_j$. Both steps implement a two-stage approach wherein robust least squares parameter estimates are produced, along with their standard errors, and then these estimates fed into a second stage that shrinks to improve precision. Our approach permitted us to produce priors without the need of a non-linear algorithm. This was an important feature given the size of the typical datasets: one million SNPs and several hundred thousands samples distributed across dozens of batches. This approach resulted in a powerful software tool that outperforms the default algorithm in computation speed.

#### 3.3.1 Estimating SNP-specific shifts

To estimate $\mathbf{V}$ we use an empirical Bayes approach (Louis and Carlin, 2009). We start by obtaining robust versions of the sample means and variances of the training data to estimate the $\mu$'s and $\sigma$'s by $\hat{\mu}_{ig}$ and $\tilde{\sigma}_{ig}$. These robust estimates are used to account for the $t$-distributed errors. Since the training dataset is considered the reference from which batches deviate, we assume $\lambda = 0$, and thus $\hat{\mu}_{ig}$ and $\tilde{\sigma}_{ig}$ are unbiased estimates. Then, $\mathbf{V}$ is estimated by the sample variance-covariance of $\vec{\mu} = (\hat{\mu}_{iAA}, \hat{\mu}_{iAB}, \hat{\mu}_{iBB})'$, $i = 1, \ldots, I$, producing $\mathbf{V}$. Note that some genotypes will have very few points available in the training data to use in estimating the $\mu$'s and $\sigma$'s and the estimate will be imprecise. Now, to borrow strength across SNPs we use $\mathbf{V}$ to shrink the $\hat{\mu}_{ijg}$ using the posterior distribution formula for a multivariate Gaussian:

$$
\hat{\mu}_{ijg} = (\mathbf{W}_i^{-1} + \mathbf{S}_{g ij}^{-1})^{-1} \mathbf{W}_i^{-1} \hat{\mu}_{ijg}
$$

with $\mathbf{W}_i$ a diagonal matrix with entries $\tilde{\sigma}_{ijg}^2$, $g = 1, \ldots, 3$ and $\tilde{\sigma}_{ijg}$ the number of points available in the training data to estimate $\hat{\mu}_{ijg}$. Similarly, we shrink the variance estimates (Smyth, 2004), which protects against biases that can be induced by small sample size situations:

$$
\tilde{\sigma}_{ijg}^2 = \frac{(N_{ijg} - 1)\tilde{\sigma}_{ijg}^2 + d_g \sigma_{ijg}^2}{(N_{ijg} - 1) + d_g}
$$

for $N_{ijg} > 1$.

When $N_{ijg} \leq 1$, we simply use the posteriors $\tilde{\sigma}_{ijg}^2$. These computations use the training data and most users will not have access to it. Therefore, we save the $\hat{\mu}_{ijg}$'s, $\tilde{\sigma}_{ijg}$'s and $N_{ijg}$'s and include them as part of the software that implements CRLMM version 2.

#### 3.3.2 Estimating batch-specific shifts

Here, we describe the two-stage approach used to estimate $\lambda_j$ for each batch $j = 1, \ldots, J$. The general idea was to use the previously estimated SNP-specific shift parameters, $\hat{\mu}_{ijg}$'s and $\tilde{\sigma}_{ijg}$'s, to produce preliminary posteriors for each genotype. These were used to create a pseudo-training dataset. The $\tilde{\lambda}_{ijg}$ were then estimated following a procedure similar to the one used to estimate $\mu$. Some details follow.

The first step is to obtain starting values for the posteriors by assuming there is no batch-specific shift, $\lambda = 0$ and that the SNP-specific shifts $\mu$ are known:

$$
\hat{\mu}_{ijg}^0 = \Pr(Z_{ijk} = g|Z_{ijk} = g, \mu_{ijg}, \nu_{ijg} = 0, \sigma_{ijg} = \sigma_{ijg}).
$$

We then assign a genotype to each SNP for each sample in the batch by simply maximizing these posteriors:

$$
\hat{\nu}_{ijg}^0 = \arg\max_g \hat{\mu}_{ijg}^0.
$$

A pseudo-training dataset was created with these calls. The expected value of $M_{ijk}$ conditioned on $Z_{ijk} = g$ is $\mu_{ijg}^0 + \hat{\lambda}_{ijg}$. We therefore assume that the average (in practice we compute a robust average) deviation

$$
\hat{\nu}_{ijg} = \frac{1}{N_{ijg}} \sum_{k \in X_{ijg}} (M_{ijk} - f_{jk}(S_{ijk}) - \hat{\mu}_{ijg}).
$$

with $X_{ijg} \equiv \{k \text{ such that } Z_{ijk}^0 = g \}$ and $N_{ijg}$ is the number of elements in $X_{ijg}$, is an unbiased estimate of $\hat{\lambda}_{ijg}$. In the second stage, $\tilde{\lambda}_{ijg}$ is estimated with the sample variance-covariance of $\hat{\lambda}_{ijg} = (\hat{\lambda}_{iAA}, \hat{\lambda}_{iAB}, \hat{\lambda}_{iBB})'$, $i = 1, \ldots, I$. With $\tilde{\lambda}_{ijg}$, the
estimate of $U_j$, in place, we shrink the $\hat{\lambda}_{ij}$ as done in (2):

$$\hat{\lambda}_{ij} = (U_j^{-1} + W_i^{-1})^{-1}W_i^{-1}\lambda_{ij}$$

(3)

with $W_i$ as above.

### 3.4 Producing posteriors

Using the CRLMM version 1, posterior calls were particularly overconfident. This is consistent with the fact that the estimated $\mu_i$ are assumed to be known. We developed a procedure that permits us to account for the uncertainty associated with estimating the SNP- and batch-specific shifts. In this section, we illustrate the idea by demonstrating the approach when there are no batch-specific shifts and the $c$’s are normally distributed. In the Supplementary Material, we describe the details needed for the full model, including the batch-specific shifts and the $t$-distribution assumption.

Consider the simplified model with no batch effect (thus $j$ is omitted):

$$\begin{align*}
M_k | Z_{ik} = g, \mu_i = \hat{\mu}_i & = f_{k}\delta(S_{ik}) + \hat{\mu}_i + \epsilon_{ik}, \\
\forall [M_k | Z_{ik} = g] & = \sqrt{\sum \frac{\lambda_{ik}}{\sigma^2_i}} \mathbb{N}(\mu_i, \sigma_i^2) + \mathbb{E}(\epsilon_{ik}) = \mathbb{E}(\epsilon_{ik})
\end{align*}$$

(4)

with $\epsilon$ normally distributed with mean 0 and variance $\sigma^2_i$. In our approach, we estimate with a shrunken version of the sample average, but for simplicity we will assume we used the sample average. In this case, the estimated SNP-specific shifts, $\hat{\mu}_i$, are normally distributed with mean 0 and variance $\sigma^2_i$, with $N_{ij}$ the number of points available in the training data to estimate $\mu_{ij}$ as in (3). We can then show that

$$\begin{align*}
\mathbb{E}[M_k | Z_{ik} = g] & = \mathbb{E}(\mathbb{E}[M_k | Z_{ik} = g, \mu_{ij}]) = \mathbb{E}(f_{k}\delta(S_{ik}) + \hat{\mu}_i + \epsilon_{ik}) \\
= f_{k}\delta(S_{ik}) + \mathbb{E}(\epsilon_{ik}) + \mathbb{E}(\mathbb{E}(\epsilon_{ik})) = f_{k}\delta(S_{ik}) + \mathbb{E}(\epsilon_{ik})
\end{align*}$$

(5)

The non-zero correlations permitted us to borrow strength across SNPs. The methodology described here is available via the crlmm R/BioConductor package. To demonstrate its performance, we compared CRLMM version 2 with Birdseed, the standard genotyping tool for SNP 6.0 arrays, on the 270 HapMap samples. On this set, the maximum amount of memory used by CRLMM version 2, during preprocessing, was 3.2 GB. After preprocessing, the memory usage was reduced to 2 GB. CRLMM version 2 needed 52 min to complete the task. Birdseed used 845 MB for most of the process, increasing slowly to 900 MB and took 150 min. The comparisons were executed on a four-processors system (3 GHz Dual-Core AMD Opteron Processor 2222) with 32 GB RAM.

We demonstrate the utility of this approach in Section 4. For SNPs, we can quantify quality by assigning a posterior probability of being an outlier to each shift, i.e. $\mu_i$ or $\lambda_i$. Using the fitted prior distributions for $\mu_i$ and $\lambda_i$, we introduce a density function $h_0$ for outlying $\mu$ and compute the posterior probability:

$$\begin{align*}
Pr(\text{shift } i \text{ is outlier}(\mu_i) = h_0(\mu_i)/\phi(\mu_i)
\end{align*}$$

with $\phi(\mu_i) = (2\pi)^{-3/2} |\mathbf{V}|^{-1/2} \exp(-1/2 \mathbf{V}^{-1} \mu_i)$. A practical choice for $h$ is the 3D uniform distribution covering all possible values of $\mu$. We perform a similar computation for $\lambda_{ij}$ for each batch $j$. To illustrate the advantage of the empirical Bayes approach, we plotted the $\lambda_{iAA}$ versus $\lambda_{iAB}$ and $\lambda_{iAA}$ versus $\lambda_{iBB}$ (Fig. 4). The large number of SNPs permitted us to borrow strength across SNPs. The non-zero correlations permitted us to borrow strength across genotypes.

### 3.6 Software

### 3.5 Quality scores

Carvalho et al. (2007) present a powerful procedure for detecting problematic arrays based on the estimated $f$. Here, we present a quality assessment procedure for SNPs and hybridization batches. The quality of batch $j$ can be quantified by the diagonal entries of $U_j$. The non-zero correlations permitted us to borrow strength across SNPs. The non-zero correlations permitted us to borrow strength across genotypes.

The methodology described here is available via the crlmm R/BioConductor package. To demonstrate its performance, we compared CRLMM version 2 with Birdseed, the standard genotyping tool for SNP 6.0 arrays, on the 270 HapMap samples. On this set, the maximum amount of memory used by CRLMM version 2, during preprocessing, was 3.2 GB. After preprocessing, the memory usage was reduced to 2 GB. CRLMM version 2 needed 52 min to complete the task. Birdseed used 845 MB for most of the process, increasing slowly to 900 MB and took 150 min. The comparisons were executed on a four-processors system (3 GHz Dual-Core AMD Opteron Processor 2222) with 32 GB RAM.

The implementation of this algorithm follows the standards used by CRLMM version 1: it provides the pair genotype call and confidence score for each sample at every available SNP and does not perform any type of automatic filtering, which is left to the researcher.
4 RESULTS

We assessed the performance of CRLMM version 2 with a comparison to CRLMM version 1 and Birdseed, the default algorithm provided by the manufacturer. We use three datasets:

(A) 143 HapMap samples hybridized by Affymetrix on Affymetrix SNP 6.0 arrays.

(B) 55 HapMap samples hybridized at Johns Hopkins on Affymetrix SNP 6.0 arrays.

(C) 3050 samples from the GoKinD dataset (Mueller et al., 2006), hybridized on Affymetrix SNP 5.0 arrays, made available through the Genetic Association Information Network (GAIN).

We used HapMap samples because knowing the ‘truth’ permitted us to effectively assess our methodology. Note that, although the same samples, the hybridizations used here were not the same as the set used to train our algorithm. Dataset C provided a large set with 34 different batches defined by the 96-well plate in which the samples were stored. To assess performance with this dataset we computed the concordance between calls obtained by running the algorithm on all samples to calls obtained by running the algorithms by batch. We obtained calls for each dataset with Birdseed, CRLMM version 1 and CRLMM version 2.

The major additions introduced by CRLMM version 2 and described below are (A) the set of metrics for the assessment of SNP Quality Control (SNP QC) (Section 4.1) and batch quality (Section 4.2) and (B) well-calibrated confidence scores. These metrics in conjunction to the Signal-to-Noise Ratio or SNR (Carvalho et al., 2007) offer a powerful set of tools for the quality evaluation of the genotyping procedure. The researcher is then able to flag low-quality SNPs, samples and batches. The appropriate use of these instruments enhances performance and reliability of the genotype calling algorithm, as the aforementioned sections in addition to Sections 4.3 and 4.4 demonstrate.

4.1 SNP quality metrics

For Datasets A and B, we computed SNP Quality Control (QC) Scores as described previously. Namely, for each SNP, we calculated the posterior probability of the estimated \( \lambda \) not being an outlier. This is of great importance for researchers, as it provides means of identifying SNPs whose genotype calls are very accurate. In practical terms, this means that if the investigator uses SNPs with higher scores, it is unlikely that many mistakes will be observed. To demonstrate the utility of this metric, we stratified SNPs by the quality score reported for Datasets A and B, and created Accuracy versus Drop Rate (ADR) plots for each stratum, shown by Figure 5. Note, that by restricting attention to SNPs with QC scores above 0.25, we obtained near perfect results. For Datasets A and B, 98.63% and 99.18% of the SNPs surpassed this cutoff, as Table 1 shows.

<table>
<thead>
<tr>
<th>SNP QC</th>
<th>Dataset A</th>
<th>Dataset B</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01–0.05</td>
<td>0.0027</td>
<td>0.0016</td>
</tr>
<tr>
<td>0.05–0.25</td>
<td>0.0047</td>
<td>0.0039</td>
</tr>
<tr>
<td>0.25–1.00</td>
<td>0.9863</td>
<td>0.9918</td>
</tr>
</tbody>
</table>

Fig. 5. ADR plots for Datasets A and B. SNPs were stratified by their quality scores and ADR curves were produced for each stratum. The scores are shown to successfully identify SNPs with lower accuracies. The removal of such SNPs significantly increases the method’s accuracy.

4.2 Batch quality metrics

For Dataset C, we do not have reference calls to compare against, like we do for samples that are part of HapMap. Therefore, we generated calls in two ways: (i) by using and (ii) by ignoring batch information. We then computed the concordance between these two sets of calls for each batch after dropping calls whose confidence scores were below the 5th percentile. These concordances are represented on the y-axis of Figure 6A and B. We considered batches with lower concordance to be problematic. The percentage of samples with signal-to-noise ratios, as defined by Carvalho et al. (2007), below five was the best predictor of low-quality batches, as Figure 6A shows. The next significant improvement introduced by CRLMM version 2, the batch quality score predicted low quality as well, as demonstrated by Figure 6B.

Our batch quality score also effectively predicted the differences in accuracy observed in Figure 7. Note that Datasets A and B had batch quality scores of 0.0337 and 0.0745, respectively. For each SNP, we count how many samples were classified into each genotype cluster and denote it by \( N_{ijg} \). Using \( \lambda_{ijg} \) and \( N_{ijg} \), we estimate the shift that can be attributed to each observation in that match the suggested SNP QC threshold (0.25).
The batch quality metric is then determined by the average variance

\[
\text{batchQC}(q) = \frac{\text{trace}\left(\Delta_j^2\right)}{3}
\]

for which we recommend a threshold \(q = 0.70\).

### 4.3 Overall accuracy

We then compared overall accuracy using Datasets A and B. We calculated accuracy, i.e., proportion of correct calls, for calls with confidence scores above a given cutoff. Various cutoffs were considered. We then plotted accuracy against the proportion of calls below the confidence cutoffs. The ADR plots, Figure 7, demonstrated that, overall, CRLMM version 2 outperformed the other two algorithms.

### 4.4 Posteriors

To assess the validity of the posteriors, we compared observed accuracy with reported posteriors. Specifically, we stratified calls by their associated posterior and, for each strata, we computed the proportion of correct calls. We then plotted these against each other with the expectation that they fall on the identity line. Although CRLMM version 1 does not use posteriors as a confidence measure, we obtained the posteriors by modifying its code. CRLMM version 2 is compared with CRLMM version 1, which is clearly too optimistic. The dashed lines represent homozygotes, dotted lines the heterozygotes and the solid lines the overall accuracies.

### 5 DISCUSSION

We have presented a multi-level enhancement to the CRLMM model described by Carvalho et al. (2007). Our sole objective is to provide accurate genotype calls, calibrated confidence scores and quality metrics based on observed intensities of SNP probes and a set of parameter estimates (location and scale) obtained from a training dataset. This strategy does not make use of any other information, like known copy number regions. Our model accounts for three levels of variability in SNP array data: (i) SNP-specific shifts, (ii) hybridization batch shifts to each SNP and (iii) heavy tailed measurement error. By explicitly modeling these sources of uncertainty, the estimated posterior probabilities are much improved as compared with those offered by CRLMM version 1. We also incorporate the variability associated with estimating model parameters with training data. Our approach produced priors with superior properties to those produced by CRLMM version 1. The refinements will improve the accuracy of downstream results obtained from probability-based association tests such as the one described by Ruczinski et al. (http://www.bepress.com/jhubiostat/paper181).

We have also described methodology useful for detecting problematic SNPs and hybridization batches. We find the latter contribution particularly important. Adapting analysis tools to deal with hybridization batch effects should be a priority of analysis groups working with GWAS data. Due to experimental logistics, GWAS rarely control or randomize for well-plate, for example, when using external controls. Therefore, an undetected problematic batch could make it difficult, if not impossible, to distinguish reported associations from artificial ones such driven by hybridization batches. We have presented a powerful solution that predicts problematic batches and can be easily incorporated into any analysis pipeline.

### ACKNOWLEDGEMENTS

We thank the Genetic Association Information Network (GAIN) and James Warram for the GoKinD data; Simon Cawley, Affymetrix and Dan Arking for HapMap gold standard calls. CRLMM version 2 is compared with CRLMM version 1, which is clearly too optimistic. The dashed lines represent homozygotes, dotted lines the heterozygotes and the solid lines the overall accuracies.

Fig. 7. ADR plots for Datasets A and B. For the first set, CRLMM version 2 outperforms both Birdseed and CRLMM version 1. For the second set, it outperforms the other two methods roughly at a drop rate of 6%. Also note that the accuracy on the second dataset is lower when compared with the first one, indicating significant variation on the quality of the two sets.

Fig. 8. For Dataset A, calls were stratified by their associated posterior. For each strata the observed accuracy was computed by comparing to HapMap gold standard calls. CRLMM version 2 is compared with CRLMM version 1, which is clearly too optimistic. The dashed lines represent homozygotes, dotted lines the heterozygotes and the solid lines the overall accuracies.
for GoKinD is provided by the National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK). The GoKinD collection of DNA was genotyped through the GAIN program, with the support of the FNHI and NIDDK. The dataset used for the analyses described in this manuscript were obtained from the database of Genotype and Phenotype (dbGaP) found at http://www.ncbi.nlm.nih.gov/gap through dbGaP accession number phs000018.v1.p1.

Funding: National Institutes of Health grants R01GM083084 from the National Institute of General Medicine; 5R01RR021967 from the National Center for Research Resource; R01 DK061662 from the National Institute of Diabetes; Digestive and Kidney Institute; R01 GM083084, a CTSA grant to the Johns Hopkins Medical Institutions; doctoral scholarship awarded by the Brazilian Funding Agency CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior).

Conflict of Interest: none declared.

REFERENCES


