5-22-2009

FROZEN ROBUST MULTI-ARRAY ANALYSIS (fRMA)

Matthew N. McCall
Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health

Benjamin M. Bolstad
Department of Biostatistics, University of California, Berkeley

Rafael A. Irizarry
Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, rafa@jhu.edu

Frozen Robust Multi-array Analysis (fRMA)

Matthew N. McCall, Benjamin M. Bolstad, Rafael A. Irizarry*

May 22, 2009

Abstract

Robust Multi-array Analysis (RMA) is the most widely used preprocessing algorithm for Affymetrix and Nimblegen gene-expression microarrays. RMA performs background correction, normalization, and summarization in a modular way. The last two steps require multiple arrays to be analyzed simultaneously. The ability to borrow information across samples provides RMA various advantages. For example, the summarization step fits a parametric model that accounts for probe-effects, assumed to be fixed across arrays, and improves outlier detection. Residuals, obtained from the fitted model, permit the creation of useful quality metrics. However, the dependence on multiple arrays has two drawbacks: (1) RMA cannot be used in clinical settings where samples must be processed individually or in small batches and (2) data sets preprocessed separately are not comparable. We propose a preprocessing algorithm, frozen RMA (fRMA), which allows one to analyze microarrays individually or in small batches and then combine the data for analysis. This is accomplished by utilizing information from the large publicly available microarray databases. In particular, estimates of probe-specific effects and variances are precomputed and frozen. Then, with new data sets, these are used in concert with information from the new array(s) to normalize and summarize the data. We find that fRMA is comparable to RMA when the data are analyzed as a single batch and outperforms RMA when analyzing multiple batches. The methods described here are implemented in the R package *frma* and are currently available for download from the software section of http://rafalab.jhsph.edu

1 Introduction

Affymetrix and Nimblegen expression microarrays are composed of oligonucleotide probes 25 base-pairs in length. These are designed to match transcripts of interest and are referred to as perfect match (PM) probes. Genes are typically represented by groups of these probes referred to as probe-sets. The typical probe-set is comprised of 11 probes. Each array contains tens of thousands of probe-sets. Mismatch (MM) probes are also included; however, because RMA does not use MMs and Affymetrix appears to be phasing them out, we do not discuss MMs here.

Statistical analysis of these arrays begins with the data generated from scanning and consists of reducing the data from the probe level to the gene level in a step referred to as preprocessing. There are numerous preprocessing algorithms. Bolstad (2004) provides an extensive list of the various algorithms and compares them based on the Affymetrix HGU133a Spike-In data set. He finds that, in general, methods that fit models across arrays outperform methods that process each array separately. Therefore, it is not surprising that the most popular preprocessing algorithms perform multi-array analysis: these include RMA (Irizarry *et al.*, 2003), gcRMA (Wu *et al.*, 2004), MBEI (Li and Wong, 2001), and PLIER (Affymetrix, 2005). In this paper we focus on RMA, the most widely used procedure; however, the ideas presented here can be applied to most multi-array methods.

Like most preprocessing algorithms, RMA performs three steps: *background correction*, *normalization*, and *summarization*. The last two steps require multiple arrays, and we briefly review them below. The background correction step is performed on each array individually, and we do not discuss it here. We refer the reader to Bolstad (2004) for a detailed explanation of the background correction procedure.

Once probe intensities have been background corrected, a normalization step is required to remove variation due to target preparation and hybridization. This is necessary to make data from different arrays comparable. Using a spike-in experiment, Bolstad *et al.* (2003) demonstrated that quantile normalization has the best overall performance among various competing methods. This algorithm forces the probe intensity distribution to be the same on all the arrays. To create this *reference distribution* each quantile is averaged across arrays.

After background correction and normalization, we are left with the task of summarizing probe intensities into gene-expression to be used in downstream analysis. A simple approach is to report the mean or median of the PM intensities in each probe-set; however, this approach fails to take advantage of the well documented *probe-effect*. Li and Wong (2001) first observed that the within-array variability between probes within a probe-set is typically greater than the variability of an individual probe across arrays. To address this, Irizarry *et al.* (2003) proposed the following probe-level model:

$$Y_{ijn} = \theta_{in} + \phi_{jn} + \varepsilon_{ijn}, \qquad (1)$$

with Y_{ijn} representing the log_2 background corrected and normalized intensity of probe $j \in 1, \ldots, J_n$, in probe-set $n \in 1, \ldots, N$ on array $i \in 1, \ldots, I$. Here θ_{in} represents the expression of probe-set n on array i, and ϕ_{jn} represents the probe-effect for the jth probe of probe-set n. Measurement error is represented by ε_{ijn}. Note that θ is the parameter of interest as it is interpreted as gene-expression.

For identifiability, the probe effects are constrained within a probe-set to sum to zero; this can be interpreted as assuming that on average the probes accurately measure the true gene-expression. Note that, given this necessary constraint, a least squares estimate of θ would not change if the probe effects were ignored, i.e $\phi_{in} = 0$ for all i and n. However, outliers are common in microarray data, and robust estimates of the θs do change if we include the probe-effect parameters. This is illustrated by Figure 1. In this figure a value that appears typical when studying the data of just one array, is clearly detected as an outlier when appropriately measuring the probe effect. The figure also demonstrates that failure to appropriately down-weight this probe can result in a false difference when comparing two arrays.

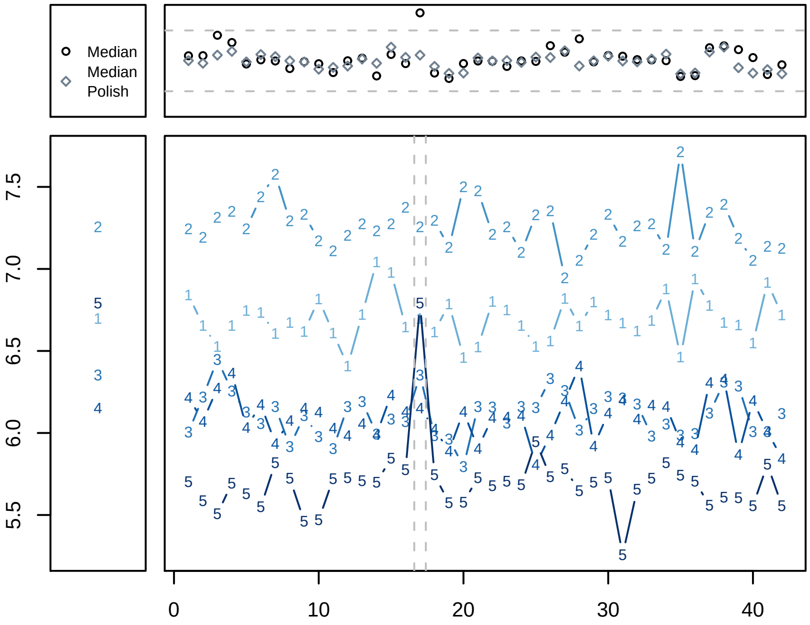

Figure 1: Log_2 probe-level expression for the WDR1 gene across 42 arrays from the Affymetrix HGU133a Spike-In experiment. The large pane shows expression values for 5 probes designed to measure the same gene across the 42 arrays. The left most pane shows the expression data for only array 17, and the top pane shows the expression values from median and median polish. Ignoring the probe effect amounts to looking at only the left pane where probe 5 does not appear to be an outlier. The large pane shows that probe 5 is easily detected as an outlier on array 17 when fitting a multi-array model. The top pane shows the advantage of multi-array methods which account for the probe-effect – the median expression value for array 17 is over-expressed, but the expression value from median polish is not.

To estimate the θs robustly, the current implementation of RMA in the Bioconductor R package *affy* (Gautier *et al.*, 2004) uses median polish, an ad-hoc procedure developed by Tukey (1977). However, Model 1 can be fit using more statistically rigorous procedures such as M-estimation techniques (Huber *et al.*, 1981). An implementation of this approach is described in Bolstad (2004) and is implemented in the Bioconductor R package *affyPLM*. In this implementation, the standard deviation of the measurement errors is assumed to depend on probe-set but not probe nor array, i.e. we assume $Var(\varepsilon_{ijn}) = \sigma_n^2$ does not depend on i or j.

Originally, median polish was used over more statistically rigorous procedures due to its computational simplicity. Median polish has remained the default in the Bioconductor implementation of RMA because competing procedures have not outperformed it in empirically-based comparisons (data not shown). However, the M-estimators provide an advantage for the development of quality metrics since estimates of σ_n^2 and standard error calculations for the estimates of θ are readily available. Therefore, both median polish and M-estimator approaches are currently widely used.

Although multi-array methods typically outperform single-array ones, they come at a price. For example, a logistics problem arises from the need to analyze all samples at once which implies that data sets that grow incrementally need to be processed every time an array is added. More importantly, as we demonstrate later, artifacts are introduced when groups of arrays are processed separately. Therefore, available computer memory limits the size of an experiment and the feasibility of large meta-analyses. Furthermore, for microarrays to be used in clinical diagnostics, they must provide information based on a single array.

Two multi-array tasks that current single array methods can not perform are: (1) computing the reference distribution used in quantile normalization and (2) estimating the ϕs and $Var(\varepsilon_{ij})$ in Model (1). Katz *et al.* (2006) proposed performing these tasks by running RMA on a reference database of biologically diverse samples. The resulting probe-effect estimates, $\hat{\phi}$, and the reference distribution used in the quantile normalization step were stored or *frozen* for future use. For a single new array they proposed the following algorithm: (1) background correct as done by RMA, (2) force the probe intensity to have the same distribution as the frozen reference distribution (quantile normalization), and (3) for each probe-set report the median of $y_{ij} - \hat{\phi}_j$. They showed that this algorithm outperforms earlier attempts at single-array preprocessing such as MAS5.0 but falls short of RMA.

Katz *et al.* (2006) assumed that the ϕ_js are constant across studies; however, we find that some probes behave differently from study to study. Note that to measure the gene expression in a sample, a sequence of steps are carried out: (1) target preparation, (2) hybridization, and (3) scanning. We define a microarray *batch* as a group that underwent these steps in the same lab during the same time period. This should not be confused with an *experiment*, a group of arrays intended to be used collectively to address a question. Experiments are typically composed of one or more batches. Because laboratory technician experience and various environmental factors can alter the results of these steps (Fare *et al.*, 2003; Irizarry *et al.*, 2005), one must be careful when comparing microarray data generated under different conditions. These between-batch differences are commonly referred to as *batch effects.* Figure 2 demonstrates that some probes behave differently from batch to batch even after quantile normalization. If

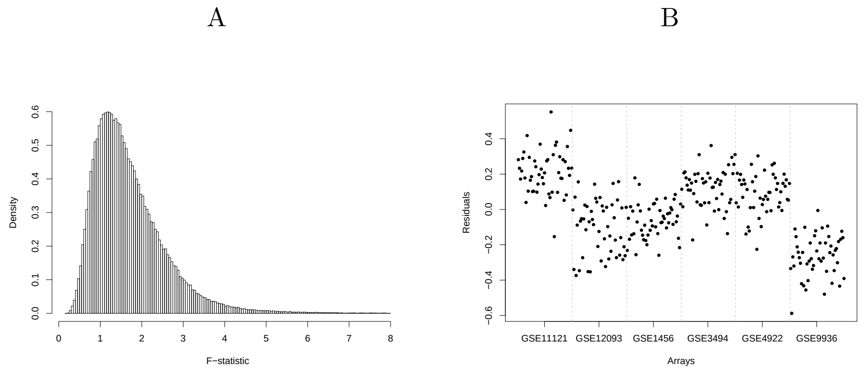

Figure 2: Plots demonstrating batch specific probe effects. (A) Histogram of F-statistics comparing between batch versus within batch variability for each probe in the database. An F-statistic of 1.31 corresponds to a p-value of 0.01 when testing the null hypothesis: no batch specific probe-effect. Over half the probes have an F-statistic greater than 1.31 showing strong evidence against the probe effects being constant between batches. (B) Residuals for a probe obtained by fitting a probe-level linear model to 300 arrays - 50 from each of 6 different breast tumor studies. This is an example of a probe that shows much greater variability between batches than within.

enough probes behave this way, then it is no surprise that a procedure that estimates probe effects for the batch in question, such as RMA, outperforms the method proposed by Katz *et al.* (2006). In this paper, we propose a methodology that takes this probe/batch interaction into account to produce an improved single array method. Furthermore, we noticed that even within batches, variability differs across probes (see Figure 3). Current approaches assume $Var(\varepsilon_{ij})$ is constant across probes. An approach that weights probes according to their precision is more appropriate.

In this paper we expand upon the work of Katz *et al.* (2006) to develop fRMA – a methodology that combines the statistical advantages of multi-array analysis with the logistical advantages of single-array algorithms. In Section 2 we describe the new procedure and the model that motivates it. In Section 3 we demonstrate the advantages of fRMA. Finally, in Section 4 we summarize the findings.

2 Methods

We assume the following probe level model:

$$Y_{ijkn} = \theta_{in} + \phi_{jn} + \gamma_{jkn} + \varepsilon_{ijkn}. \tag{2}$$

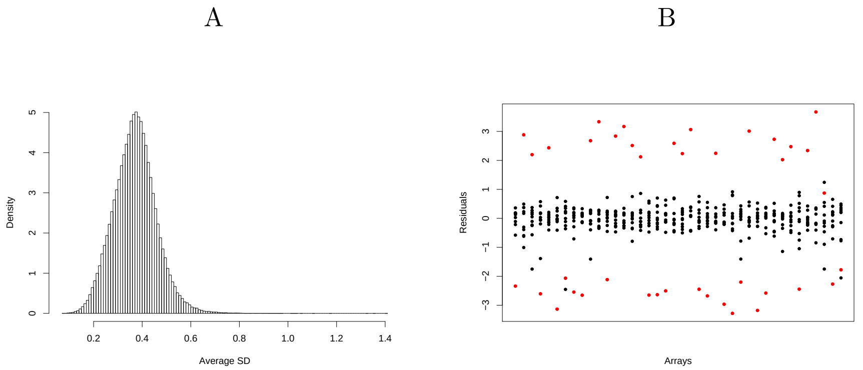

Figure 3: Plots demonstrating that different probes show different variability. (A) Histogram of average within-batch residual standard deviation. The long right tail demonstrates that some probes are far more variable than others within a batch. Treating these as being equally reliable as less variable probes produces suboptimal results. (B) Residuals from fitting a probe-level linear model to a batch of 40 arrays from GSE1456. The probe in red shows considerably more variability than the other probes within the same probe-set.

The parameters and notation here are the same as those in Model (1) with a few exceptions. First, we added the notation $k \in 1, \dots, K$ to represent batch and a random effect term, γ, that explains the variability in probe effects across batches. Note that for batch k, we can think of $\phi_{jn} + \gamma_{jkn}$ as the batch-specific probe-effect for probe j in probe-set n. In our model, the variance of the random effect is probe specific, $Var(\gamma_{jkn}) = \tau^2_{jn}$. The second difference is that we permit the within-batch probe variability to depend on probe as well, i.e. $Var(\varepsilon_{ijkn}) = \sigma^2_{jn}$.

The first step in our procedure was to create a reference distribution, to be used in quantile normalization, and to estimate the ϕs, τs and σs from a fixed set of samples. To accomplish this, we created a database of 850 samples from the public repositories GEO (Edgar *et al.*, 2002) and ArrayExpress (Parkinson *et al.*, 2008). We refer to these as the training data set. We selected the arrays to balance studies and tissues. Specifically, we generated all the unique experiment/tissue type combinations from roughly 6000 well-annotated samples. We then randomly selected 5 samples from each experiment/tissue type combination with at least 5 samples. This resulted in 170 experiment/tissue type combinations. The GEO accession numbers for all 850 samples can be found in Supplementary Table 1.

The standard way to fit model (2), a random effects model, to data known to have outliers is not straight forward. Therefore, we adopted a modular approach, which we describe in detail here. First, we fit model (1) using a robust procedure to obtain $\hat{\phi}_j$ and $\hat{\theta}_i$ for each sample i and probe j. We then used the residuals,

$r_{ijkn} = Y_{ijkn} - (\hat{\theta}_i + \hat{\phi}_j)$ to estimate the variance terms τ^2 and σ^2. Specifically, we defined $\hat{\tau}^2_{jn} = \frac{1}{K}\sum_k(\bar{r}_{.jkn} - \bar{r}_{.j.n})^2$ and $\hat{\sigma}^2_{jn} = \frac{1}{IK}\sum_k\sum_i(r_{ijkn} - \bar{r}_{.jkn})^2$ where $\bar{r}_{.jkn} = \frac{1}{I}\sum_i r_{ijkn}$ and $\bar{r}_{.j.n} = \frac{1}{IK}\sum_k\sum_i r_{ijkn}$.

With these estimates in place we were then ready to define a preprocessing procedure for single arrays and small batches. We motivate and describe these next.

2.1 fRMA algorithm

First, we background corrected each new array in the same manner as the training data set. Remember RMA background correction is a single array method. Second, we quantile normalized each of the new arrays to the reference distribution created from the training data set. The final step was to summarize the probes in each probe-set. Note that the arrays analyzed are not part of any of the batches represented in the training data set. For presentation purposes, we denote the new batch by l.

The first task in the summarization step was to remove the global batch effect from each intensity and create a probe-effect-corrected intensity:

$$Y^*_{ijln} \equiv Y_{ijln} - \hat{\phi}_{jn} = \theta_{in} + \gamma_{jln} + \varepsilon_{ijln}$$

The second task was to estimate the θs from these data using a robust procedure. A different approach was used for single array summarization and batch summarization.

2.1.1 Single array

Here we dropped the i and l notation because we are analyzing only one array and one batch.

We estimated θ_n with a robust mean that weights each of the data points by the inverse of its variance:

$$\mathrm{Var}(Y^*_{jn}) = \tau^2_{jn} + \sigma^2_{jn}.$$

The log gene-expression was then estimated by the weighted mean:

$$\hat{\theta}_n = \sum_{j=1}^{J_n} \frac{w_{jn}}{v_{jn}} Y^*_{jn} / \sum_{j=1}^{J_n} \frac{w_{jn}}{v_{jn}}$$

with $v_{jn} = \hat{\tau}^2_{jn} + \hat{\sigma}^2_{jn}$ and w_{jn}, the weights obtained from a M-estimator procedure. This statistic has an intuitive interpretation – probes with large batch to batch (τ) or array to array (σ) variation should be down-weighted, as well as, intensities that are outliers (small w).

2.1.2 Batch of arrays:

Here we dropped the l notation because this method is intended to be applied to arrays from the same batch.

Note that the probe-effect corrected data $\mathbf{Y}^*_{jn} \equiv \{Y^*_{ijn}\}_{i=1,\dots,I}$ are correlated because they share the random effect γ_{jn}. We therefore implemented a robust procedure that accounts for this correlation. We rewrote model (2) in matrix notation:

$$\mathbf{Y}^*_n = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\delta}.$$

Here $\mathbf{Y}^*_n \equiv (\mathbf{Y}^*_{1,n}, \dots, \mathbf{Y}^*_{J_n,n})'$ is a vector of all the probe-effect corrected intensities for probe-set n, $\boldsymbol{\theta} \equiv (\theta_{1n}, \dots, \theta_{In})'$ are the parameters of interest, $\mathbf{X} \equiv 1_{(J_n \times 1)} \otimes I_{(I \times I)}$ is a matrix of indicator variables, and $\boldsymbol{\delta}$ is a vector of correlated errors with covariance matrix $\boldsymbol{\Sigma} \equiv (\tau_{1,n}, \dots, \tau_{J_n,n})' \times 1_{(1 \times J_n)} \otimes 1_{(I \times I)} + (\sigma_{1,n}, \dots, \sigma_{J_n,n})' \times 1_{(1 \times J_n)} \otimes I_{(I \times I)}$. Here $\otimes$ is used to represent the Kronecker Product. Note that the entries of $\boldsymbol{\Sigma}$ were estimated from the training set and treated as known. Therefore, we can easily rotate the intensities into independent identically distributed data: $\mathbf{Z} \equiv \boldsymbol{\Sigma}^{-1/2}\mathbf{Y}$. We then estimated the transformed parameters $\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\theta}$ using a standard M-estimator. Note that the final estimate can be expressed as a weighted least squares estimate:

$$\hat{\boldsymbol{\theta}} = (\mathbf{X}'\boldsymbol{\Sigma}^{-1/2}\mathbf{W}\boldsymbol{\Sigma}^{-1/2}\mathbf{X})^{-1}\boldsymbol{\Sigma}^{-1/2}\mathbf{W}\boldsymbol{\Sigma}^{-1/2}\mathbf{Y}^*_n$$

with $\mathbf{W}$ a diagonal matrix of the weights obtained from the robust procedure.

This estimate also has an intuitive interpretation as probes with large correlation get down-weighted and correlation is taken into account in the definition of distances used to define an outlier. Note that if just one of the entries of $\mathbf{Y}^*_{jn}$ is large in absolute value, it is likely an outlier, but, if all entries are large, it is probably due to a large batch-specific probe effect and is not considered an outlier.

3 Results

To demonstrate the utility of the fRMA algorithm, we compared it to RMA. First we assessed the preprocessing algorithms in terms of accuracy, precision, and overall performance as done by *affycomp* and *spkTools* (Irizarry *et al.*, 2006; McCall and Irizarry, 2008). Then we assessed robustness to batch effects using two publicly available data sets.

We used the Affymetrix HGU133a spike-in data set to calculate measures of accuracy, precision, and overall performance. To assess accuracy, we calculated the *signal detection slope*, the slope from regressing observed expression on nominal concentration in the log_2 scale. It can be interpreted as the expected difference in observed expression when the true difference is a fold change of 2; as such, the optimal result is one. To assess precision, we computed the standard deviation of null log-ratios and the 99.5th percentile of the null distribution. Here *null* refers to the transcripts which were not spiked in and therefore should not be differentially expressed. The first precision measure is an estimate of the expected spread of observed log-ratios for non-differentially expressed genes. The second precision measure assesses outliers; we expect 0.5% of non-differentially expressed genes to exceed this value. Lastly, we calculated two measures of overall performance – the signal-to-noise ratio (SNR) and the probability of a gene with a true log_2 fold change of 2 being in a list of the 100 genes with the greatest fold change (POT). These measures were computed in 3 strata based on average expression across arrays. For a more detailed explanation of these measures, see McCall and Irizarry (2008).

First, the measures described above were calculated treating the data as a single batch; the results can be seen in Table 1. Then, the same data were preprocessed in the three original batches in which the data were generated; the results for these analyses can be seen in Table 2. In both tables we also report the results from preprocessing the data with fRMA one array at a time. When the data were preprocessed as a single batch, RMA outperformed fRMA in the medium stratum based on SNR and POT. But in the low and high strata, RMA and fRMA performed comparably. When we processed the data in batches, fRMA outperformed RMA in all three strata primarily due to better precision.

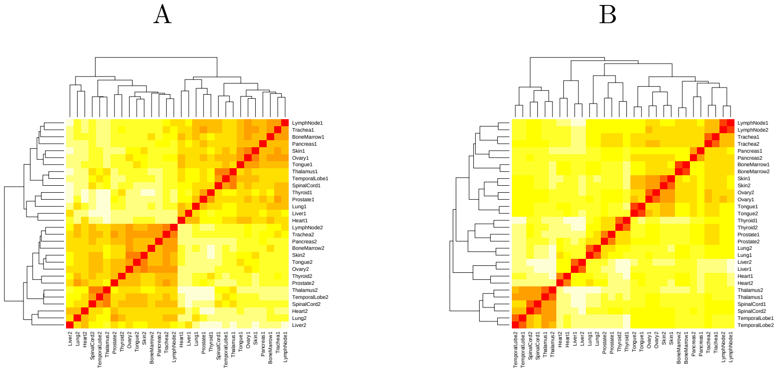

Figure 4: Heatmaps of 15 tissue types hybridized on 2 arrays each and preprocessed in 2 batches – (A) was preprocessed using RMA and (B) was preprocessed using single array fRMA.

To assess the effect of combining data preprocessed separately, we created two artificial batches each containing the same 15 tissues from the E-AFMX-5 data set (Su *et al.*, 2004). We then analyzed each batch separately with RMA and fRMA. After obtaining a matrix of expression values for each batch, we performed hierarchical clustering on the combined expression matrix. Figure 4 shows that when the samples were preprocessed with RMA, they clustered based on the artificial batches, but when they were preprocessed with fRMA, they clustered based on tissue type.

Next, we compared batch effects using a publicly available breast cancer data set of 159 Affymetrix HGU133a arrays accessible at the NCBI GEO database (Edgar *et al.*, 2002), accession GSE1456 (Pawitan *et al.*, 2005). The dates on which the arrays were generated varied from June 18th, 2002 to March 8th, 2003. We grouped the data into 6 batches based on these dates (see Supplementary Table 2) and processed each batch separately using fRMA. We also processed

Low					
	Accuracy	Precision		Performance	
Preprocessing	Slope (SD)	SD	99.5%	SNR	POT
fRMA - batch	0.26 (0.24)	0.10	0.36	2.60	0.33
RMA	0.25 (0.32)	0.10	0.36	2.50	0.36
fRMA - single array	0.25 (0.25)	0.11	0.45	2.27	0.22

Medium					
	Accuracy	Precision		Performance	
Preprocessing	Slope (SD)	SD	99.5%	SNR	POT
RMA	0.83 (0.37)	0.09	0.40	9.22	0.88
fRMA - batch	0.80 (0.35)	0.09	0.44	8.89	0.85
fRMA - single array	0.82 (0.37)	0.10	0.49	8.20	0.82

High					
	Accuracy	Precision		Performance	
Preprocessing	Slope (SD)	SD	99.5%	SNR	POT
fRMA - single array	0.58 (0.18)	0.06	0.23	9.67	0.98
RMA	0.57 (0.18)	0.06	0.22	9.50	0.97
fRMA - batch	0.62 (0.20)	0.07	0.25	8.86	0.97

Table 1: For these results we treated the spike-in data as a single batch. For each of the intensity strata we report summary assessments for accuracy, precision, and overall performance. The first column shows the signal detection slope which can be interpreted as the expected observed difference when the true difference is a fold change of 2. In parenthesis is the standard deviation of the log-ratios associated with non-zero nominal log-ratios. The second column shows the standard deviation of null log-ratios. The SD can be interpreted as the expected range of observed log-ratios for genes that are not differentially expressed. The third column shows the 99.5th percentile of the null distribution. It can be interpreted as the expected minimum value that the top 100 non-differentially expressed genes will reach. The fourth column shows the ratio of the values in column 1 and column 2. It is a rough measure of signal to noise ratio. The fifth column shows the probability that, when comparing two samples, a gene with a true log fold change of 2 will appear in a list of the 100 genes with the highest log-ratios.

the entire data set as one batch using RMA. Table 3 shows that there are statistically significant differences in average expression from batch to batch when the data are processed with RMA. These differences are not present when the data are analyzed using fRMA.

Low					
	Accuracy	Precision		Performance	
Preprocessing	Slope (SD)	SD	99.5%	SNR	POT
fRMA - batch	0.26 (0.26)	0.10	0.33	2.60	0.39
fRMA - single array	0.25 (0.25)	0.11	0.45	2.27	0.22
RMA	0.25 (0.32)	0.14	0.47	1.79	0.25

Medium					
	Accuracy	Precision		Performance	
Preprocessing	Slope (SD)	SD	99.5%	SNR	POT
fRMA - batch	0.81 (0.36)	0.09	0.40	9.00	0.87
fRMA - single array	0.82 (0.37)	0.10	0.49	8.20	0.82
RMA	0.83 (0.39)	0.12	0.46	6.92	0.83

High					
	Accuracy	Precision		Performance	
Preprocessing	Slope (SD)	SD	99.5%	SNR	POT
fRMA - batch	0.58 (0.20)	0.06	0.21	9.67	0.97
fRMA - single array	0.58 (0.18)	0.06	0.23	9.67	0.98
RMA	0.58 (0.20)	0.08	0.26	7.25	0.95

Table 2: Just as in Table 1 but processed in three batches then combined for analysis.

4 Discussion

We have described a flexible preprocessing algorithm for Affymetrix expression arrays that performs well whether the arrays are preprocessed individually or in batches. The algorithm follows the same three steps as current algorithms: background correction, normalization, and summarization. Specifically, we have improved upon the summarization step by accounting for between-probe and between-batch variability.

Table 1 demonstrated that when analyzing batches of data together, RMA performed slightly better. Table 2 showed that when analyzing data in batches, fRMA consistently outperformed RMA. Specifically, fRMA showed greater precision than RMA.

Perhaps the greatest disadvantage of multi-array preprocessing methods is the inability to make reliable comparisons between arrays preprocessed separately. Figure 4 showed the potentially erroneous results that one might obtain when combining data preprocessed separately. Furthermore, Table 3 showed that even if it were computationally feasible to preprocess all the data simultaneously with RMA, it would be unwise to do so due to batch effects. Unlike RMA, fRMA accounts for these batch effects and thereby allows one to combine data from different batches for down-stream analysis.

As more data become publicly available, methods that allow simultaneous analysis of thousands of arrays become necessary to make use of this wealth of

RMA			
Coefficients	Estimate	Std. Error	P-value
Intercept	6.758	0.003	<0.001
batch2	-0.014	0.003	<0.001
batch3	0.002	0.004	0.658
batch4	-0.004	0.004	0.338
batch5	-0.010	0.003	0.004
batch6	-0.021	0.004	<0.001

fRMA			
Coefficients	Estimate	Std. Error	P-value
Intercept	7.183	0.002	<0.001
batch2	≈0.000	0.003	0.949
batch3	≈0.000	0.003	0.937
batch4	≈0.000	0.003	0.885
batch5	≈0.000	0.003	0.968
batch6	-0.001	0.003	0.656

Table 3: This table displays coefficients obtained from regressing gene expression on array batch. RMA shows a significant batch effect while fRMA does not.

data. fRMA allows the user to preprocess arrays individually or in small batches and then combine the data to make inferences across a wide range of arrays. This ability will certainly prove useful as microarrays become more common in clinical settings.

The preprocessing methods examined here can be summarized based on what information they use to estimate probe effects. RMA uses only the information present in the data being currently analyzed; whereas fRMA utilizes both the information present in the data being analyzed and the information from the database. By using both sources of information, fRMA is able to perform well across a variety of situations.

The fRMA methodology can be easily extended to provide quality metrics for a single array. Brettschneider *et al.* (2008) demonstrate that the Normalized Unscaled Standard Error (NUSE) can detect aberrant arrays when other quality metrics fail. The NUSE provides a measure of precision for each gene on an array relative to the other arrays. Precisions is estimated from the RMA model residuals. Therefore, the NUSE is multi-array on two counts. Using the fRMA methodology, one can develop a single-array version of NUSE. Precision can be estimated from the residuals described in Section 2 and the relative precision can be computed relative to all the arrays in the training dataset.

Finally, note that fRMA requires a large database of arrays of the same platform. Currently, our software only handles two human arrays: HGU133A and HGU133Plus2. However, we have downloaded all available raw data (CEL) files for five other popular platforms and expect to have software for these in the near future.

Acknowledgments

We thank Terry Speed for suggesting the name fRMA. We thank the maintainers of GEO and ArrayExpress for all the hard work that goes into keeping these databases organized. We thank Michael Zilliox, Harris Jaffe, Marvin Newhouse, and Jiong Yang for help downloading and storing data.

Funding National Institutes of Health (GM083084 to R.I., RR021967 to R.I., T32GM074906 to M.M.).

References

Affymetrix, I. (2005). Guide to probe logarithmic intensity error (PLIER) estimation. *Technical Note*.

Bolstad, B. (2004). *Low-level analysis of high-density oligonucleotide array data: background, normalization and summarization*. Ph.D. thesis, University of California.

Bolstad, B., Irizarry, R., Astrand, M., and Speed, T. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, **19**(2), 185–193.

Brettschneider, J., Collin, F., Bolstad, B., and Speed, T. (2008). Quality assessment for short oligonucleotide microarray data. *Technometrics*, **50**(3), 241–264.

Edgar, R., Domrachev, M., and Lash, A. (2002). Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, **30**(1), 207.

Fare, T., Coffey, E., Dai, H., He, Y., Kessler, D., Kilian, K., Koch, J., LeProust, E., Marton, M., Meyer, M., *et al.* (2003). Effects of atmospheric ozone on microarray data quality. *Analytical chemistry*, **75**(17), 4672.

Gautier, L., Cope, L., Bolstad, B., and Irizarry, R. (2004). affyanalysis of Affymetrix GeneChip data at the probe level. *Bioinformatics*, **20**(3).

Huber, P., Wiley, J., and InterScience, W. (1981). *Robust statistics*. Wiley New York.

Irizarry, R., Hobbs, B., Collin, F., Beazer-Barclay, Y., Antonellis, K., Scherf, U., and Speed, T. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, **4**(2), 249.

Irizarry, R., Warren, D., Spencer, F., Kim, I., Biswal, S., Frank, B., Gabrielson, E., Garcia, J., Geoghegan, J., Germino, G., *et al.* (2005). Multiple-laboratory comparison of microarray platforms. *Nature Methods*, **2**(5), 345–350.

Irizarry, R., Wu, Z., and Jaffee, H. (2006). Comparison of Affymetrix GeneChip expression measures. *Bioinformatics*, **22**(7), 789–794.

Katz, S., Irizarry, R., Lin, X., Tripputi, M., and Porter, M. (2006). A summarization approach for Affymetrix GeneChip data using a reference training set from a large, biologically diverse database. *BMC Bioinformatics*, **7**(1), 464.

Li, C. and Wong, W. (2001). Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *Proceedings of the National Academy of Sciences*, **98**(1), 31.

McCall, M. and Irizarry, R. (2008). Consolidated strategy for the analysis of microarray spike-in data. *Nucleic Acids Research*, **36**(17), e108.

Parkinson, H., Kapushesky, M., Kolesnikov, N., Rustici, G., Shojatalab, M., Abeygunawardena, N., Berube, H., Dylag, M., Emam, I., Farne, A., *et al.* (2008). ArrayExpress update–from an archive of functional genomics experiments to the atlas of gene expression. *Nucleic Acids Research*.

Pawitan, Y., Bjöhle, J., Amler, L., Borg, A., Egyhazi, S., Hall, P., Han, X., Holmberg, L., Huang, F., Klaar, S., *et al.* (2005). Gene expression profiling spares early breast cancer patients from adjuvant therapy: derived and validated in two population-based cohorts. *Breast Cancer Research*, **7**(6), R953–R964.

Su, A., Wiltshire, T., Batalov, S., Lapp, H., Ching, K., Block, D., Zhang, J., Soden, R., Hayakawa, M., Kreiman, G., *et al.* (2004). A gene atlas of the mouse and human protein-encoding transcriptomes. *Proceedings of the National Academy of Sciences*, **101**(16), 6062–6067.

Tukey, J. (1977). *Exploratory Data Analysis*. Addison-Wesley.

Wu, Z., Irizarry, R., Gentleman, R., Martinez-Murillo, F., and Spencer, F. (2004). A model-based background adjustment for oligonucleotide expression arrays. *Journal of the American Statistical Association*, **99**(468), 909–917.