Tools for Spike-in Data Analysis and Visualization (spkTools)

Matthew N. McCall

October 24, 2025

Contents

1 Introduction **1**

2 SpikeInExpressionSet **2**

3 spkTools Methods **2**

3.1 Relating nominal concentrations across data sets 2

3.2 Accuracy assessment 3

3.3 ALE strata 3

3.4 Precision assessments 3

3.5 Performance assessments 4

3.6 Imbalance measure 5

4 Example **5**

1 Introduction

As the number of users of microarray technology continues to grow, so does the importance of platform assessments and comparisons. Spike-in experiments have been successfully used for internal technology assessments by microarray manufacturers and

for comparisons of competing data analysis approaches. The microarray literature is saturated with statistical assessments based on spike-in experiment data. Unfortunately, the statistical assessments vary widely and are applicable only in specific cases. This has introduced confusion into the debate over best practices with regards to which platform, protocols, and data analysis tools are best. Furthermore, cross-platform comparisons have proven difficult because reported concentrations are not comparable. We present a novel statistical solution that enables cross-platform comparisons, and propose a comprehensive procedure for assessments based on spike-in experiments. The ideas are implemented in a user friendly Bioconductor package: spkTools.

This document describes spkTools, which implements the methods suggested in the Nucleic Acids Research paper, *Consolidated strategy for the analysis of microarray spike-in data* (McCall & Irizarry 2008). While ideally someone interested in using this package would use that paper as a guide, sections relating specifically to the spkTools package are included here.

2 SpikeInExpressionSet

This package defines a new S4 class that extends the *ExpressionSet* class to include a matrix of nominal concentrations; this new class is called a *SpikeInExpressionSet*. The functions implemented in this package take an object of this type as their input and produce the tables and plots presented in this paper. Of particular interest is the function `spkAll`, which is a wrapper function for all the functions contained in this package. When run on a *SpikeInExpressionSet* object, it produces the full complement of tables and plots shown in this paper and saves them with easily recognizable filenames. Although this package was designed with the intent of producing the full array of results for each experiment, the functions can also be applied separately with a few exceptions where the output of one function is required as the input of another.

3 spkTools Methods

3.1 Relating nominal concentrations across data sets

Our solution to the problem of mapping was to replace each nominal concentration with the average log expression across arrays (ALE) for genes spiked in at that concentration. This approach assures that performance assessments based on spike-in data are related to expression measurements that are defined consistently across platforms: low, medium, and high ALE values correspond to low, medium, and high observed expression values respectively.

3.2 Accuracy assessment

With the ALE values in place, we were ready to adapt some of the existing statistical assessments to cross-platform comparisons. We started with a basic assessment of accuracy: the *signal detection slope*. Microarrays are designed to measure the abundance of sample RNA. In principle, we expect a doubling of nominal concentration to result in a doubling of observed intensity. In other words, on the $\log_2$ scale, the slope from the regression of expression on nominal concentration can be interpreted as the expected observed difference when the true difference is a fold change of 2. Thus, an optimal result is a slope of one, and values higher and lower than one are associated with over and under estimation respectively.

3.3 ALE strata

It has been noted that at very high and very low concentrations one typically observes lower slopes compared to those seen at medium concentrations. To address this, we consider the signal detection slopes for genes spiked-in at low, medium, and high ALE values. We implemented a data-driven approach to selecting these two cut-offs.

We defined f to be the function that maps nominal log concentration x to expected observed concentration $f(x)$. Using a cubic spline, fitted to the observed data, we obtained a parametric representation of f. We then looked for concentrations for which clear changes in sensitivity occurred, i.e. values of x with large slope changes. Note that large changes in slope result in local maxima in the absolute value of the second derivative of f. For each platform, the absolute value of the second derivative f'' showed two clear local maxima. For each platform we mapped each concentration x to its corresponding empirical percentile $\Phi(x)$ and plotted $|f''(x)|$ against $\Phi(x)$. The percentiles that maximized the slope change were similar across platforms. The modes for the average curve were 0.615 and 0.993. Therefore, for the purpose of this comparison, we assigned as *low*, ALE values less than the 60th percentile of the distribution of background RNA. Similarly we defined as *high* ALE values above the 99th percentile. The remaining ALE values, between the 60th and 99th percentile, were denoted as *medium*. Our choice of cut-points was further motivated by observing that for the Affymetrix data the 60th percentile provided a good cut-off for distinguishing genes called present from genes called absent.

3.4 Precision assessments

To complete our comparison we needed to assess specificity. Because the majority of microarray studies rely on relative measures (e.g. fold change) as opposed to absolute

ones, we focused on the precision of the basic unit of relative expression: log-ratios. We adapted the precision assessment of Cope et al. that focused on the variability of log-ratios generated by comparisons expected to produce log-ratios of 0. Our set of comparisons was created by making all possible comparisons between spiked-in transcripts across arrays in which they had the same nominal concentration and from all possible comparisons within the background RNA. We referred to this group of comparisons as the *Null* set. The standard deviation (SD) of these log-ratios served as a basic assessment of precision and has a useful interpretation: it is the expected range of observed log-ratios for genes that are not differentially expressed.

Because specificity varies with nominal concentration, we stratified these comparisons into low, medium, and high ALE values. Many outliers were observed on each platform. This was expected given the documented problem of cross-hybridization. Because a platform with larger SD and small outliers might be preferable to one with a smaller SD but large outliers we included the 99.5th percentile of the null distribution as a second summary assessment of specificity. Note that in a typical experiment close to 0.5% of null genes are expected to exceed this value, which translates to approximately 100 genes on whole genome arrays. We also included comparisons of spike-ins expected to yield a certain fold change. These serve to further demonstrate the variability of relative expression across ALE strata. They also serve as a rough illustration of the accuracy of log-ratios for each ALE strata.

3.5 Performance assessments

Precision and accuracy assessments on their own may not be of much practical use. However, the summary statistics described above can be easily combined to answer any practical question, as long as it can be posed in a statistical context. We focus on two summaries related to the common problem of detecting differentially expressed genes. Note that we purposely developed summaries that do not directly penalize for a lack of accuracy and precision as long as the real differences are detected. However, as expected, detection ability was highly dependent on accuracy and precision.

For the first example, we computed the chance that, when comparing two samples, a gene with true log fold change $\Delta = 1$ will appear in a list of the top 100 genes (highest log-ratios). We refer to this quantity as the *probability of being at the top* (POT) and recommend computing it separately in each ALE strata. Specifically, we assume that the log-ratios in each ALE strata follow a normal distribution with mean and variance estimated from the data (accuracy slope and standard deviation) and compute the probability that a random variable from that distribution exceeds the 99.5th percentile of the null distribution.

As a second example, we computed the expected size of a gene list one would have to

consider to find n genes that have a true log fold change Δ. To perform this calculation we assumed m_1 genes were differentially expressed and m_0 were not. Note that $m_1 + m_0$ is the number of genes on the array. Furthermore, we assumed that the true log-ratios in each ALE strata followed a normal distribution with mean and variance estimated from the data (accuracy slope and standard deviation). The empirical distribution was used for the null genes. With these assumptions in place we computed the gene list size for $n = 10$, $m_1 = 100$, and $m_0 = 10000$, we calculate the gene list size, N, required to obtain $n = 10$ true fold changes. We refer to this quantity as the *gene-list needed to detect n true-positives (GNN)*. Again, we recommend computing it separately in each ALE strata.

3.6 Imbalance measure

Those interested in taking advantage of our methodology should know that an important requirement is a spike-in experimental design that does not confound nominal concentrations and genes. A large source of variability in microarray data is the probe-effect and these vary across platforms. We fitted an ANOVA model to describe the probe effect for each platform. Note that if nominal concentrations are confounded with genes, it becomes impossible to separate differences due to signal detection from differences in probe affinities. Many of the previously published spike-in experiments suffer from this confounding effect. To quantify design imbalance we used the following measure of imbalance developed by Wu (1981):

$$IB = \sum_{i=1}^{p} \lambda_i \sum_{u_i}^{r_i} \Big(\sum_{t=1}^{T} n_t^2(u_i) - \frac{1}{T} n^2(u_i) \Big) \tag{1}$$

where i denotes each covariate, λ_i an optional weight associated with each covariate, u_i are the possible levels for covariate i, t represents the treatment levels, $n_t(u_i)$ is the number of units with its ith covariate at level u_i receiving treatment t, and $n(u_i)$ is the total number of units with its ith covariate at level u_i. In our case, the two covariates are probe and array, and the treatment is nominal concentration. Since imbalance is defined as a weighted sum of the imbalance due to each covariate, we chose to report the probe and array imbalance separately to give a better understanding of the source of the imbalance in each design. In order to not penalize large designs, we divided the probe imbalance by the number of probes and the array imbalance by the number of arrays.

4 Example

```
> library(spkTools)
```

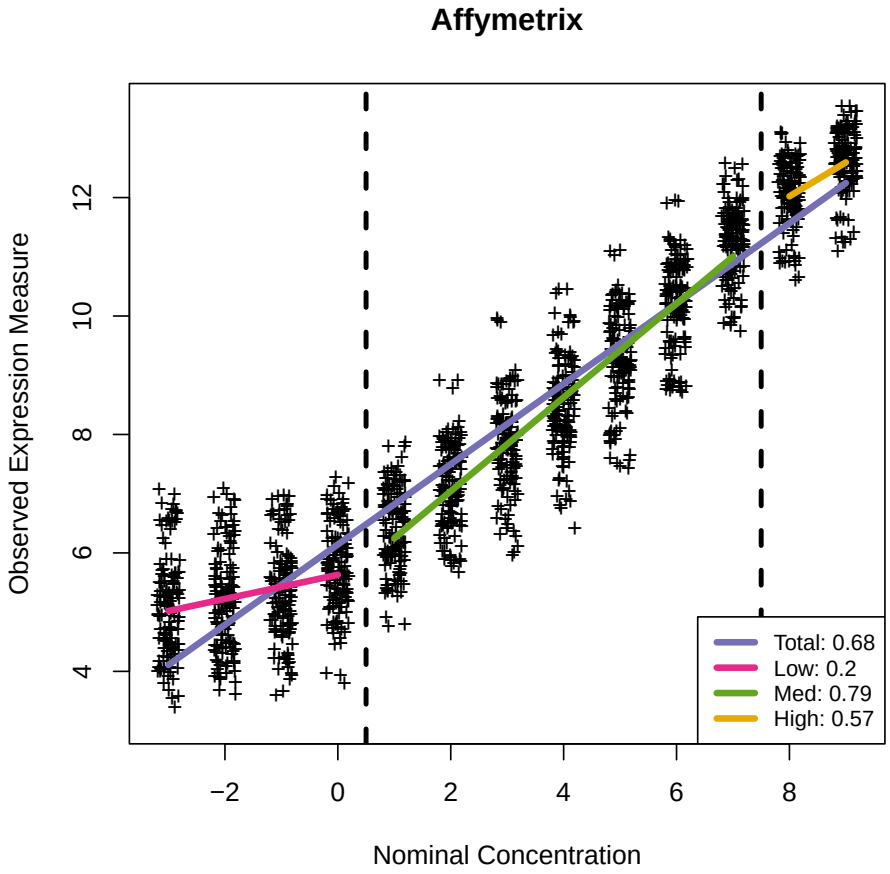
Load a SpikeInExpressionSet object :

```
> data(affy)
> object <- affy
```

Set a few parameters:

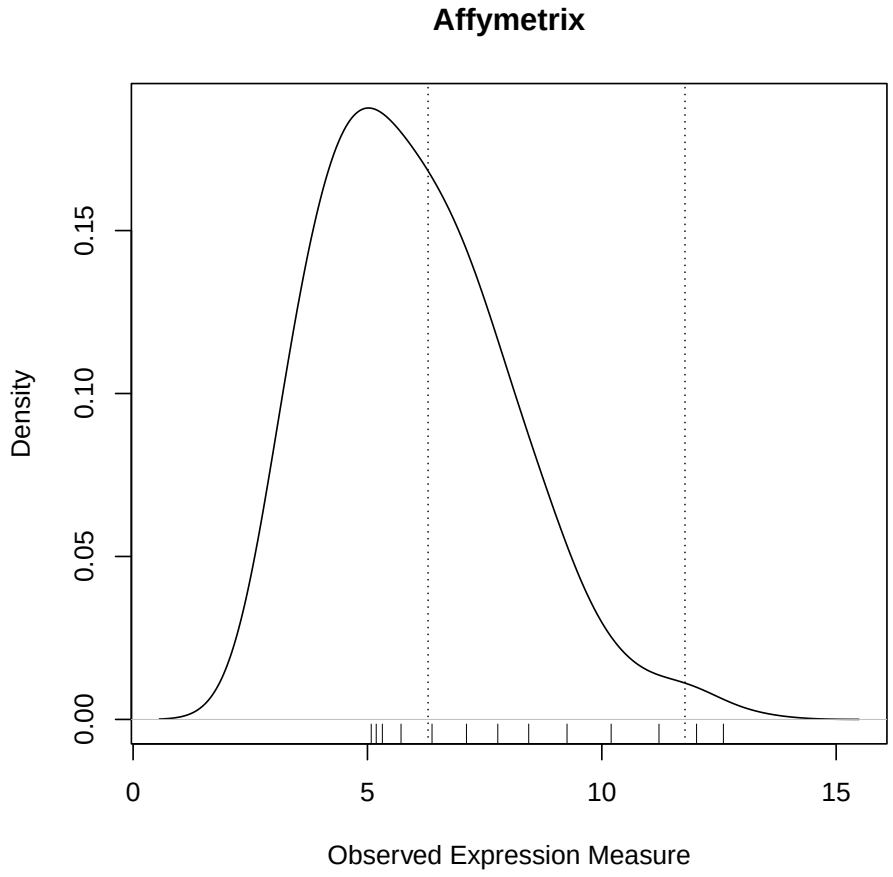
```
> fc=2
> label="Affymetrix"
> par(mar=c(3,2.5,2,0.5), cex=1.8)
```

```
> spkSlopeOut <- spkSlope(object, label, pch="+")
```



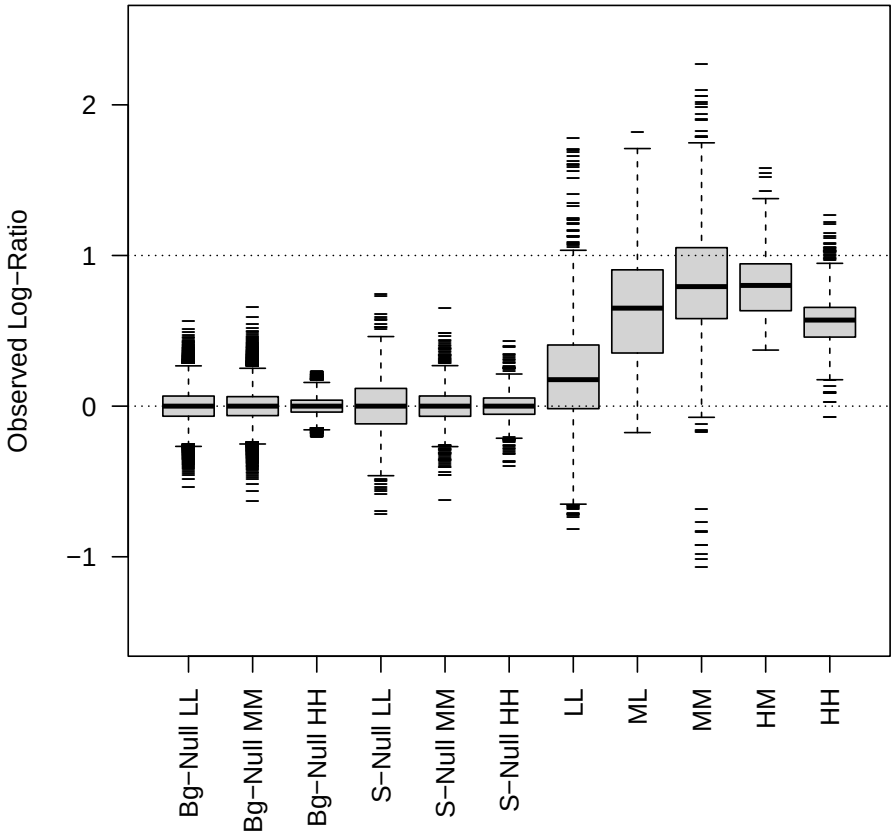
Observed versus nominal values: This plot depicts expression values plotted against the log (base 2) of the reported nominal concentration. The regression slope obtained utilizing all the data and the regression slopes obtained within each ALE value strata are shown. The slope of each line is reported in the legend. The vertical lines divide the ALE strata.

```
> spkDensity(object, spkSlopeOut, cuts=TRUE, label)
```



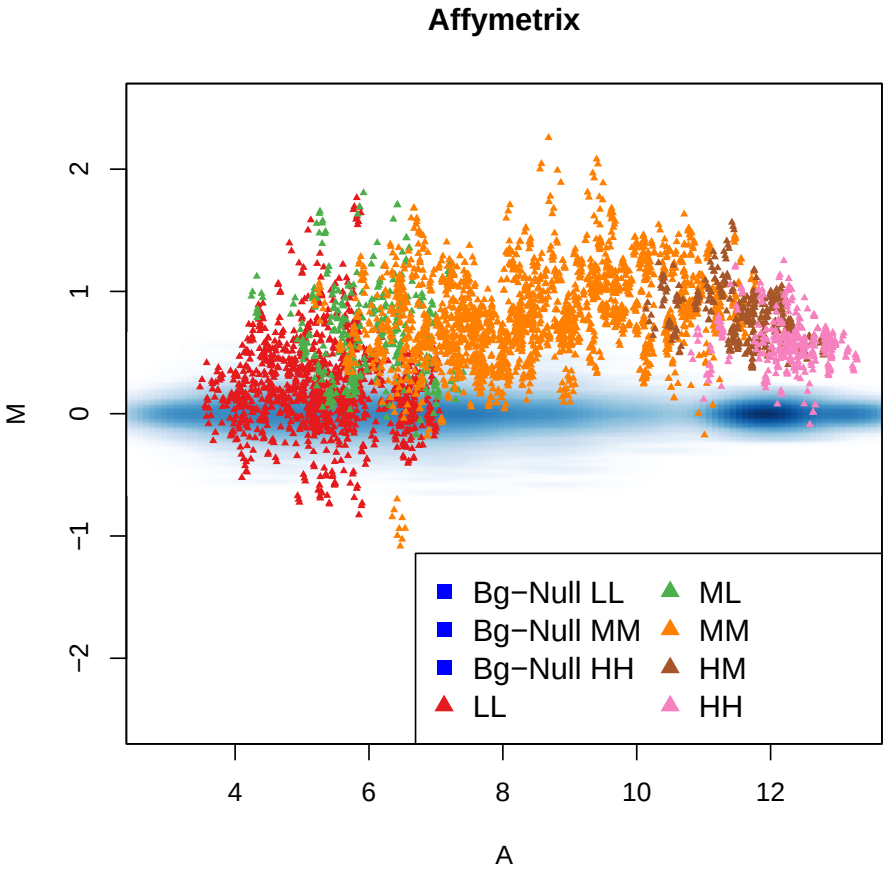
Empirical densities: This plot depicts the empirical density of the average (across arrays) expression values for the background RNA. The tick marks on the x-axis show the average expression at each nominal concentration. The dotted lines represent the cut points for low, medium, and high ALE values.

```
> spkBoxOut <- spkBox(object, spkSlopeOut, fc)
> plotSpkBox(spkBoxOut, fc, ylim=c(-1.5,2.5))
> sbox <- summarySpkBox(spkBoxOut)
```



Log-ratio distributions: This plot depicts the distribution of observed log ratios for a given nominal fold change. The log ratios are stratified by the ALE strata into which the two nominal concentrations fall. The null distributions' log-ratios are divided into background RNA (Bg-Null) and spike-ins at the same nominal concentration (S-Null), for each bin. The dotted horizontal lines represent the expected or nominal log-ratios: zero for the null distribution and one for the other comparisons.

```
> spkMA(object, spkSlopeOut, fc, label=label, ylim=c(-2.5,2.5))
```



MA plots: For each platform, we performed all pair-wise comparisons of the arrays. From each comparison we computed the log-ratio (M) and average expression value (A) for each gene. These plots show M plotted against A. To avoid drawing hundreds of points on top of each other we use a smooth scatter plot which shows the distribution of these points: dark and light shades of blue show high and low concentrations of points respectively. Points not associated with the spike-in transcripts (expected M=0) that achieved fold changes above 2 are shown as large blue dots. The points associated with spike-in transcripts with nominal fold changes of 2 are shown as triangles. The different colors denote the ALE groups.

```
> vtmp <- spkVar(object)
> sv <- as.numeric(vtmp[,2][-nrow(vtmp)])
> bin <- c("Low", "Med", "High")
> bins <- bin[spkSlopeOut$breaks[2,]]
> tab1 <- data.frame(NominalConc=2^spkSlopeOut$breaks[1,],
+                    AvgExp=round(spkSlopeOut$avgExp,1),
+                    PropGenesBelow=round(spkSlopeOut$prop,2),
+                    ALEStrata=bins,
+                    SD=round(sv,2))
> colnames(tab1) <- c("Nominal Conc",
+                     "Avg Expression",
+                     "Prop of Genes Below",
+                     "ALE Strata",
+                     "Std Dev")
```

	Nominal Conc	Avg Expression	Prop of Genes Below	ALE Strata	Std Dev
1	0.12	5.10	0.37	Low	0.87
2	0.25	5.20	0.40	Low	0.90
3	0.50	5.30	0.42	Low	0.74
4	1.00	5.70	0.50	Low	0.72
5	2.00	6.40	0.62	Med	0.82
6	4.00	7.10	0.73	Med	0.79
7	8.00	7.80	0.83	Med	0.68
8	16.00	8.40	0.88	Med	0.67
9	32.00	9.30	0.94	Med	0.79
10	64.00	10.20	0.97	Med	0.72
11	128.00	11.20	0.98	Med	0.54
12	256.00	12.00	1.00	High	0.49
13	512.00	12.60	1.00	High	0.51

Nominal concentration to ALE mapping: This table contains summary measures specific to each nominal spike-in level. The first column shows the nominal concentrations as originally reported. The second column shows the average of all observed expression values associated with the row's nominal concentration. The third column shows the proportion of background RNA with expression values less than the average expression value. The fourth column shows the ALE strata associated with the row's nominal concentration. Finally, the fifth column shows the standard deviation of all observed expression values associated with the row's nominal concentration.

```
> AccuracySlope <- round(spkSlopeOut$slopes[-1], digits=2)
> AccuracySD <- round(spkAccSD(object, spkSlopeOut), digits=2)
> pot <- spkPot(object, spkSlopeOut, AccuracySlope, AccuracySD,
+               precisionQuantile=.995)
> PrecisionSD <- round(sbox$madFC[1:3], digits=2)
> PrecisionQuantile <- round(pot$quantiles, digits=2)
> SNR <- round(AccuracySlope/PrecisionSD, digits=2)
> POT <- round(pot$POTs, digits=2)
> tab2 <- data.frame(AccuracySlope=AccuracySlope,
+                    AccuracySD=AccuracySD,
+                    PrecisionSD=PrecisionSD,
+                    PrecisionQuantile=PrecisionQuantile,
+                    SNR=SNR,
+                    POT=POT)
```

	AccuracySlope	AccuracySD	PrecisionSD	PrecisionQuantile	SNR	POT
Low	0.20	0.31	0.10	0.35	2.00	0.31
Med	0.79	0.35	0.09	0.37	8.78	0.89
High	0.57	0.15	0.06	0.19	9.50	0.99

Assessment results: For each of the ALE 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.

```
> bals <- round(spkBal(object))
> anv <- round(spkAnova(object), digits=2)
> tab3 <- t(c(anv,bals))
```

	spike	probe	array	error	Probe Imbalance	Array Imbalance
1	2.48	0.54	0.17	0.47	0.00	0.00

ANOVA results: To understand the variability contributed by differences in nominal concentrations, probe effect, and array, we fitted a 3-way ANOVA model containing only main effects to the expression values from the spike-in transcripts. The estimated standard deviation of each effect is shown in the first three columns. The forth column shows the standard deviation of the error term. Finally, a measure of the amount of confounding between nominal concentration and the other two effects is included in columns five and six. We use the measure presented by Wu in Technometrics (1981), Volume 23, Number 1. An optimal design, such as a Latin Square, will have a measure of 0 for each imbalance. The more confounding the larger these values. Because Affymetrix using a latin square design, there is no imbalance.

References

L.M. Cope, R.A. Irizarry, H.A. Jaffee, Z. Wu, and T.P. Speed. A benchmark for Affymetrix GeneChip expression measures. *Bioinformatics*, 20:323–331, 2004.

Robert C Gentleman, Vincent J. Carey, Douglas M. Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, Laurent Gautier, Yongchao Ge, Jeff Gentry, Kurt Hornik, Torsten Hothorn, Wolfgang Huber, Stefano Iacus, Rafael Irizarry, Friedrich Leisch, Cheng Li, Martin Maechler, Anthony J. Rossini, Gunther Sawitzki, Colin Smith, Gordon Smyth, Luke Tierney, Jean Y. H. Yang, and Jianhua Zhang. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5:R80, 2004. URL `http://genomebiology.com/2004/5/10/R80`.

R.A. Irizarry, B.M. Bolstad, F. Collin, L.M. Cope, B. Hobbs, and T.P. Speed. Summaries of Affymetrix GeneChip probe level data. *Nucleic Acids Research*, 31(4):e15, 2003.

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

R.A. Irizarry, L.M. Cope, and Z. Wu. Feature-level exploration of a published Affymetrix GeneChip control dataset. *Genome Biology*, 7(8):404, 2006a.

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

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

C.F. Wu. Iterative Construction of Nearly Balanced Assignments I: Categorical Covariates. *Technometrics*, 23(1):37–44, 1981.