UMPIRE: Ultimate Microarray Prediction, Inference, and Reality Engine

Jiexin Zhang and Kevin R. Coombes
Department of Bioinformatics and Computational Biology
University of Texas M.D. Anderson Cancer Center
Houston, TX 77005, USA
kcoombes@mdanderson.org

***Abstract*—High-throughput measurements of gene expression pose a challenge to analysts attempting to learn models that predict treatment response or survival. One possible explanation for the lack of significant progress in this area is the limited sample size of most experiments. Realistic simulations could help with the development and assessment of analytical methods; however, existing simulation tools have focused more on the technology and less on the biological complexity. In this paper, we introduce a package of simulation tools to address this problem. Our model incorporates additive and multiplicative noise, transcriptional activity or inactivity, and block correlation structures. More importantly, it models the multi-hit theory of cancer via latent variables that link gene expression, binary outcome, and survival data. We illustrate the use of the simulation package by showing that standard analysis methods (i.e., univariate Cox models) are only likely to recover the true structure with more samples than are included in most current studies of survival.**

***Keywords*-gene expression; microarray; simulation; class prediction; multi-hit theory of cancer**

I. Introduction

The introduction of gene expression microarrays in the 1990's ushered in an era of high-throughput biology that has required the development of novel methods for the statistical and computational analysis of large biological datasets. Richard Simon and colleagues [1] identified three kinds of problems addressed by these technologies: class comparison, class discovery, and class prediction. The current state-of-the-art has evolved reasonable methods for class comparison (e.g., gene-by-gene t-tests or ANOVA coupled with estimates of the false discovery rate) and class discovery (e.g., hierarchical clustering coupled with resampling techinques to assess robustness) [2]. However, there is less agreement on the best (or even consistently good) methods for discovering complex models that can accurately predict biologically relevent outcomes such as treatment response or survival.

Part of the difficulty is that prediction is inherently harder than class comparison or class discovery. It is conceivable that the the number of samples (typically between 100 and 300) included in most of the current studies is simply inadequate to learn effective predictive models. It is, however, extremely difficult to assess this possibility. Although some progress has been made for binary classifiers [3]–[5], we do not have general theoretical ways to justify formal sample size computations that address the combination of feature selection and model building that goes into the discovery of predictive models from high-throughput biological datasets. Nor is it possible to collect gene expression data on $10,000$ patients in order to test empirically how many samples are really needed to learn good predictive models.

The obvious solution is to use simulation. If we can simulate many datasets, of different sizes, with realistic biological properties, then we can use those datasets to evaluate proposed methods for class prediction. The simulation of microarray gene expression datasets has a long history. However, none of the existing simulation tools was designed to focus on the biological diversity related to such important outcomes as treatment response or survival. Many of the earliest simulation tools focused on the simulation of microarray images, and were useful for developing better image processing algorithms [6]–[8]. Other simulation tools have attempted to explicitly model the steps in a microarray experiment, including printing, hybridization, dye effects, and scanning [9], [10]. As with many of the early statistical simulations [11]–[14], however, most tools use a model that simply compares two homogeneous populations of samples. Even more recent and more detailed simulations still assume that the data come from two homogenous popluations [5], [15]–[17].

To address this gap, we have developed a simulation package that incorporates a heterogeneous model that is consistent with the multiple hit theory of carcinogenesis [18], [19]. Moreover, our package uses latent variables to simulate the connections between gene expression and either binary or time-to-event outcomes.

II. Homogeneous gene expression model

Version 1.0 of the Ultimate Microarray Prediction, Inference, and Reality Engine (Umpire) is an R package that allows researchers to simulate complex, realistic microarray data that is linked to binary or time-to-event outcomes. The package is available from the R reposi-

 ISBN: 978-1-61208-137-3

tory at http://bioinformatics.mdanderson.org/OOMPA; detailed instructions on how to install the package can be found at http://bioinformatics.mdanderson.org/Software/OOMPA.

The fundamental object in `Umpire` is a "random-vector generator" (RVG), which is represented by the `Engine` class. Equivalently, each `Engine` object represents a specific multivariate distribution, from which random vectors can be generated using the generic `rand` method. In Version 1.0 of `Umpire`, we include three basic components for these kinds of distributions: independent normal, independent log normal, and multivariate normal. A general `Engine` is simply a list of RVG components. Because `Umpire` is implemented using S4 classes in R, adding additional components to implement alternative models of gene expression generation is a straightforward application of object-oriented programming.

A. Additive and Multiplicative Noise

The observed signal, Y_{gi}, for gene g in sample i is:

$$Y_{gi} = S_{gi} * \exp(H_{gi}) + E_{gi}$$

where

$$S_{gi} = \text{true biological signal}$$

$$H_{gi} = \text{multiplicative noise}$$

$$E_{gi} = \text{additive noise.}$$

The noise model represents technical noise that is layered on top of any biological variability when measuring gene expression in a set of samples. For example, background noise is usually additive, while the variation between the signal pixels is multiplicative noise. We modeled additive and multiplicative noise as normal distributions:

$$E_{gi} \sim \text{Normal}(\nu, \tau)$$

$$H_{gi} \sim \text{Normal}(0, \phi)$$

Note that we allow the additive noise to include a bias term (ν) that may represent, for example, a low level of cross-hybridization providing some level of signal at all genes. The noise model is represented in the `Umpire` package by the `NoiseModel` class. Again, the object-oriented and modular design make it possible to add more elaborate noise models in the future, such as those described by Nykter and colleagues [9].

B. Active and Inactive Genes

We model the true biological signal S_{gi} as a mixture:

$$S_{gi} \sim (1 - z_g) * \delta_0 + z_g * T_{gi}$$

In this model, δ_0 is a point mass at zero, z_g defines the activity state (1 = active, 0 = inactive), and T_{gi} is the expression of a transcriptionally active gene. By allowing for some genes to be transcriptionally inactive, this design takes into account that the transcriptional activity of most genes is conditional on the biological context. Activity is modeled in `Umpire` using a binomial distribution, $z_g \sim Binom(p_0)$.

C. Expression Distributions

For most purposes, we assume that the expression, T_{gi}, of a transcriptionally active gene follows a log-normal distribution, $\log(T_g) \sim Normal(\mu_g, \sigma_g)$. In a class of samples, the mean expression of gene g on the log scale is denoted by μ_g and the standard deviation on the log scale is σ_g. Both μ_g and σ_g are properties of the gene itself and the sample class. Within a given simulation, we typically place hyperdistributions on the log-normal parameters μ_g and σ_g. We take $\mu_g \sim Normal(\mu_0, \sigma_0)$ to have a normal distribution with mean μ_0 and standard deviation σ_0. We take σ_g to have an inverse gamma distribution with *rate* and *shape* parameters. Reasonable values for the hyperparameters can be estimated from real data. For instance, $\mu_0 = 6$ and $\sigma_0 = 1.5$ are typical values on the log scale of a microarray experiment using Affymetrix arrays. The parameters for the inverse gamma distribution are determined by the method of moments from the desired mean and standard deviation; we have found that a mean of 0.65 and a standard deviation of 0.01 (for which $rate = 28.11$ and $shape = 44.25$) produce reasonable data.

D. Correlated blocks of genes

Biologically, genes are usually interconnected in networks and pathways. In fact, clustering methods are often used to group genes into correlated blocks. Thus, it is natural to simulate microarray experiments from this perspective. In our simulations, we usually allow the mean block size, bs, to range from 1 to 1000, and the sizes of gene blocks to vary around the pre-defined mean block size. To be more specific, the block size follows a normal distribution with mean bs and standard deviation $0.3{*}bs$. The case $bs = 1$ is special, since we take the standard deviation of the block size to be zero so all genes are independent. The correlation matrix for a block b, has 1's on the diagonal and ρ_b in the off-diagonal entries. We usually allow $\rho \sim Beta(pw, (1-p) * w)$ to follow a beta distribution with parameters $p = 0.6$ and $w = 5$.

We mentioned above that some genes would be transcriptionally inactive under certain biological conditions. Instead of simulating this active status for genes individually, we simulate the whole block of genes being transcriptionally active or inactive. This models the idea that the entire pathway or network could be turned on or off under certain biological conditions.

III. The Multi-hit Model of Cancer

The multiple hit theory of cancer was first proposed by Carl Nordling in 1953 [18] and extended by Alfred

 ISBN: 978-1-61208-137-3

Knudson in 1971 [19]. The basic idea is that cancer can only result after multiple insults (mutations; hits) to the DNA of a cell. We use the combinatorics of multiple hits to simulate heterogeneity in the population. Let H be the number of possible hits (typically on the order of 10 to 20). We define a cancer subtype as a collection of hits (usually 5 or 6 out of those possible). Each subtype has a prevalence; by default, each subtype is equally likely to occur in the population. To simulate a set of patients, we start by assigning them to one of the cancer subtypes (with probabilities equal to the prevalences). We then use the individual hits as (unobserved) latent variables that influence gene expression, survival, and binary outcomes. Specifically, let Z_h be a binary variable that indicates the presence ($Z_h = 1$) or absence ($Z_h = 0$) of a hit h. Then the probability p of an unfavorable (binary) outcome is simulated from a logistic model

$$log\left(\frac{p}{1-p}\right) = \sum_{h=1}^{H} \beta_i Z_i,$$

where the parameters $\beta_i \sim N(0, \sigma_B)$ are simulated from a normal distribution. We simulate survival times from a Cox proportional hazards model, with

$$h(t) = h_0(t) \sum_{h=1}^{H} \alpha_i Z_i,$$

where $h_0(t)$ can be taken to be any desired survival model (usually exponential) and the coefficients $\alpha_i \sim N(0, \sigma_A)$ can be taken to be either independent of or related to the β_i depending on the goal of the simulation. Finally, each hit is assumed to affect the expression of one correlated block of genes (representing the effect on a single biologically pathway) by altering the mean expression of the genes in that block. More elaborate models can also be generated, by altering the variances or the correlation structure within the block.

IV. Simulation Results

To illustate the `Umpire` simulation package, we have simulated a microarray data set with associated survival data. We assumed that there are 20 possible hits, and that 5 hits at a time defined a cancer subtype. For this simulation, we assumed that there were 6 distinct, equally likely, cancer subtypes. As above, each of the 20 hits corresponds to a correlated block of gene expression and also affects survival. We also assumed that there were 100 correlated blocks of genes that were unrelated to cancer or to survival. Blocks were simulated to contain a mean of 100 genes with a standard deviation of 30. Gene means, standard deviations, and correlation structures were simulated using the distributions and hyperparameters described above. We simulated survival by assuming an exponential baseline hazard function.

Table I
Number of significant genes, by sample size and FDR.

	N = 100	N = 300	N = 500
FDR = 0.01	12	86	144
FDR = 0.05	22	135	209
FDR = 0.1	37	169	253
FDR = 0.2	74	249	354
FDR = 0.3	127	346	446

We analyzed the simulated data using an approach that is common in the field. Specifically, we fit gene-by-gene univariate Cox proportional hazards models. We recorded the p values for a log-rank test of the significance of each gene. We then fit a beta-uniform mixture (BUM) model to the set of p-values, and used the BUM model to estimate the false discovery rate (FDR). Table I shows the number of genes called significant as a function of the FDR and the sample size. For an FDR of 20%, Table II separates these results into groups depending on the membership of genes in different correlated blocks. Recall that 20 correlated blocks of genes were associated with cancer-related hits; the blocks of "irrelevant" genes are collected in the row of the table labeled "FP" to denote obvious false positive findings. The first column of Table II shows the number of cancer subtypes (patterns) that included each hit; the second column shows the coefficient of that (latent) hit in the simulated survival model. Note that even though there were 20 possible hits, four of them (G4, G7, G10, and G14) were not actually included in the patterns of 5 hits that defined the 6 cancer subtypes in this simulation. Using 100 samples, we only discovered multiple genes that represented 5 of the cancer-related gene blocks. Using 500 samples, we discovered multiple genes representing all 16 "active" cancer-related gene blocks.

Figure 1 displays heatmaps of the genes selected as significant at the 20% FDR level using either 100 or 500 samples. The color bar along the top reflects the true cancer subtype for each patient. The color bar along the side displays the cancer-related gene block, with false positive genes colored white. When using 100 samples, only two or three of the six cancer subtypes can be seen in the heatmap, and only four of the cancer-related gene blocks. With 500 samples, all six cancer subtypes are visible in the heatmap, along with almost all of the cancer-related gene blocks. In both heatmaps, the false positive genes are recognizable by their lack of correlation with other selected genes.

V. Conclusion

We have described the Umpire simulation package and shown that it can be used to simulate microarray data that is related to survival outcomes in complex ways. An initial simulation using this package suggests, using

 ISBN: 978-1-61208-137-3

Table II
NUMBER OF SIGNIFICANT GENES AS A FUNCTION OF THE SAMPLE SIZE AND THE TRUE HIT STATUS.

	Patterns	Alpha	N = 100	N = 300	N = 500
G1	4	0.291	0	8	10
G2	2	0.366	0	5	11
G3	1	0.090	0	3	11
G4	0	0.278	0	1	0
G5	1	1.428	0	2	2
G6	3	0.313	0	1	2
G7	0	0.496	0	0	0
G8	1	-0.428	1	5	13
G9	3	-2.135	6	34	40
G10	0	0.631	2	1	0
G11	1	0.047	17	38	44
G12	2	0.422	0	13	27
G13	2	1.062	1	7	12
G14	0	1.433	0	2	0
G15	2	2.514	0	6	15
G16	1	-0.384	0	3	3
G17	1	-0.841	1	10	14
G18	2	0.299	0	13	16
G19	2	1.358	10	25	32
G20	2	-1.674	6	35	41
FP	0	0.000	30	37	61

a plausible set of biologically meaningful parameters, that studies to discover signatures that predict time-to-event outcomes may need more than the 100 samples that have frequently been used in practice. More detailed simulation studies will be required to test this idea further.

The results of the simulation also suggest that we may need better methods for combining gene expression values into predictive signatures. First, the common statistical approach that tries to optimize the coefficients of all 354 selected genes using 500 samples is unlikely to succeed. Moreover, since we know "ground truth" for this particular simulation, we know that there are 16 independent factors that influence survival. From the heatmap on the bottom of Figure 1, we would also estimate that there are many distinct expression patterns that contribute to survival. This observation suggests two possible approaches. On the one hand, we could group correlated genes together into simpler factors that can be included in predictive models. For example, we could perform a principal components analysis and use the first few principal components (PCs) as predictors. For our simulated data, a scree plot of the variance explained by each PC suggests that there are approximately five non-random PCs (data not shown). A Cox proportional hazards models identifies all five of those PCs as significant predictors of survival (data not shown). On the other hand, the same heatmap indicates the presence of six subtypes of cancer. An alternative approach would be to use those six subtypes as a categorical predictor; a Cox model successfully identifies these categories as significant predictors (data not shown). In this case, the obvious next step would be to develop a robust multi-category classifier.

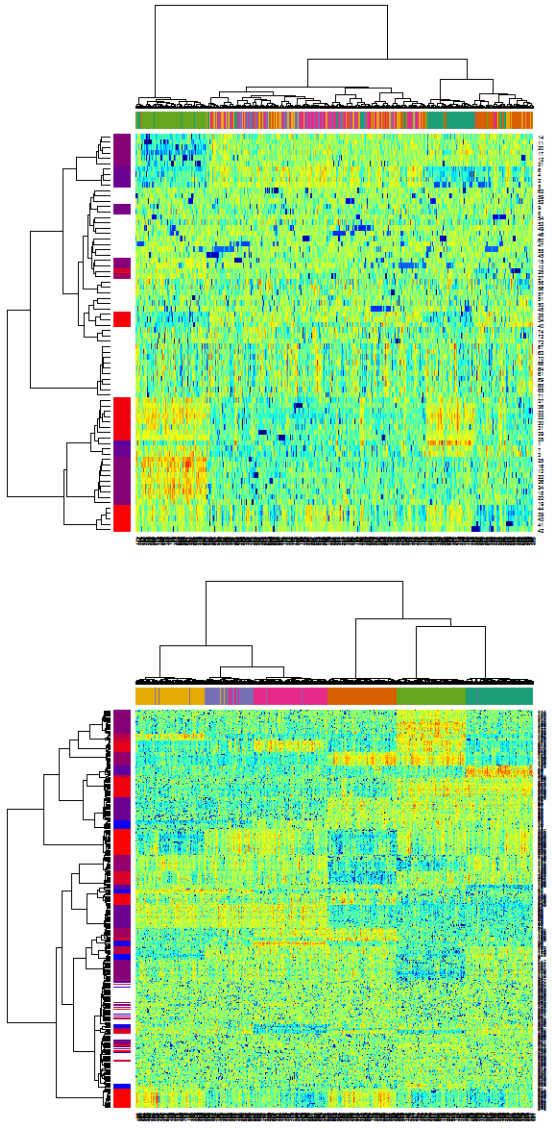

Figure 1. Heatmaps of the significant genes at FDR = 20% using 100 (top) or 500 (bottom) samples.

We do not pursue these approaches in the current paper. However, the Umpire simulation package provides the tools that are necessary to evaluate a range of analytical methods on data sets with different sizes and properties. The availability of this tool should contribute to the development of better methods to learn useful predictors of biologically relevant outcomes.

ACKNOWLEDGMENT

This research was supported by grants P30 CA016672, R01 CA123252, P50 CA070907, and P50 CA140388

 ISBN: 978-1-61208-137-3

from the National Cancer Institute of the United States National Institutes of Health.

This document was prepared using Sweave, a literate programming tool for the R statistical software environment. Complete source code, including all code necessary to run the simulations and generate the figures and tables, is available upon request.

References

[1] R. M. Simon, E. L. Korn, L. M. McShane, M. D. Radmacher, G. W. Wright, and Y. Zhao, *Design and Analysis of DNA Microarray Investigations*, ser. Statistics for Biology and Health. New York, NY: Springer-Verlag, 2003.

[2] D. B. Allison, X. Cui, G. P. Page, and M. Sabripour, "Microarray data analysis: from disarray to consolidation and consensus," *Nat Rev Genet*, vol. 7, no. 1, pp. 55–65, 2006.

[3] K. K. Dobbin and R. M. Simon, "Sample size planning for developing classifiers using high-dimensional DNA microarray data," *Biostatistics*, vol. 8, no. 1, pp. 101–17, 2007.

[4] K. K. Dobbin, Y. Zhao, and R. M. Simon, "How large a training set is needed to develop a classifier for microarray data?" *Clin Cancer Res*, vol. 14, no. 1, pp. 108–14, 2008.

[5] C. F. Aliferis, A. Statnikov, I. Tsamardinos, J. S. Schildcrout, B. E. Shepherd, and F. E. Harrell Jr., "Factors influencing the statistical power of complex data analysis protocols for molecular signature development from microarray data," *PLoS One*, vol. 4, no. 3, p. e4922, 2009.

[6] C. K. Wierling, M. Steinfath, T. Elge, S. Schulze-Kremer, P. Aanstad, M. Clark, H. Lehrach, and R. Herwig, "Simulation of DNA array hybridization experiments and evaluation of critical parameters during subsequent image and data analysis," *BMC Bioinformatics*, vol. 3, p. 29, 2002.

[7] Y. Balagurunathan, E. R. Dougherty, Y. Chen, M. L. Bittner, and J. M. Trent, "Simulation of cDNA microarrays via a parameterized random signal model," *J Biomed Opt*, vol. 7, no. 3, pp. 507–23, 2002.

[8] D. S. Lalush, "Characterization, modeling, and simulation of mouse microarray data," in *Methods of Microarray Data Analysis III*, S. M. Lin and K. F. Johnson, Eds. Boston: Kluwer Academic Publishers, 2003, pp. 75–92.

[9] M. Nykter, T. Aho, M. Ahdesmaki, P. Ruusuvuori, A. Lehmussola, and O. Yli-Harja, "Simulation of microarray data with realistic characteristics," *BMC Bioinformatics*, vol. 7, p. 349, 2006.

[10] C. J. Albers, R. C. Jansen, J. Kok, O. P. Kuipers, and S. A. van Hijum, "Simage: simulation of DNA-microarray gene expression data," *BMC Bioinformatics*, vol. 7, p. 205, 2006.

[11] K. Dobbin and R. Simon, "Comparison of microarray designs for class comparison and class discovery," *Bioinformatics*, vol. 18, no. 11, pp. 1438–45, 2002.

[12] A. Szabo, K. Boucher, W. L. Carroll, L. B. Klebanov, A. D. Tsodikov, and A. Y. Yakovlev, "Variable selection and pattern recognition with gene expression data generated by the microarray technology," *Math Biosci*, vol. 176, no. 1, pp. 71–98, 2002.

[13] I. Lonnstedt and T. Speed, "Replicated microarray data," *Statistica Sinica*, vol. 12, pp. 31–46, 2002.

[14] M. S. Pepe, G. Longton, G. L. Anderson, and M. Schummer, "Selecting differentially expressed genes from microarray experiments," *Biometrics*, vol. 59, no. 1, pp. 133–42, 2003.

[15] P. de Valpine, H. M. Bitter, M. P. Brown, and J. Heller, "A simulation-approximation approach to sample size planning for high-dimensional classification studies," *Biostatistics*, vol. 10, no. 3, pp. 424–35, 2009.

[16] R. S. Parrish, H. J. Spencer III, and P. Xu, "Distribution modeling and simulation of gene expression data," *Computational Statistics and Data Analysis*, vol. 53, pp. 1650–1660, 2009.

[17] Y. Guo, A. Graber, R. N. McBurney, and R. Balasubramanian, "Sample size and statistical power considerations in high-dimensionality data settings: a comparative study of classification algorithms," *BMC Bioinformatics*, vol. 11, p. 447, 2010.

[18] C. O. Nordling, "A new theory on cancer-inducing mechanism," *Br J Cancer*, vol. 7, no. 1, pp. 68–72, 1953.

[19] J. Knudson, A. G., "Mutation and cancer: statistical study of retinoblastoma," *Proc Natl Acad Sci U S A*, vol. 68, no. 4, pp. 820–3, 1971.

 ISBN: 978-1-61208-137-3