Simulating Gene Expression Data To Estimate Sample Size For Class and Biomarker Discovery

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

***Abstract*—With modern advances in high-throughput technologies to measure gene expression profiles, researchers are eager to identify biomarkers that indicate pathogenic processes or pharmacologic responses. However, insufficient statistical power, often due to the limited sample sizes in real experiments, has hindered progress in this area. Realistic simulations can provide data to better estimate sample sizes and better evaluate analytical methods. Existing simulation tools have focused more on the technology and less on the biological complexity of patients and outcomes. In this paper, we describe an R package of gene expression simulation tools to address this problem. Our model incorporates both biological and technical noise on top of the true signal, transcriptional status, and block structures that mimic gene networks. More importantly, to simulate the multi-hit model of cancer development, our tool contains latent variables that link gene expression with binary outcome and survival data. We demonstrate the use of this R package by providing examples of simulated cancer subtype recovery and biomarker discovery.**

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

I. Introduction

The "Ultimate Microarray Prediction, Inference, and Reality Engine" (Umpire) is an R package that allows researchers to simulate complex, realistic microarray data [1]. Simulations are useful for designing experiments and for evaluating proposed analytical methods. The simulation of microarray gene expression data sets has a long history: many of the earliest simulation tools focused on the simulation of microarray images, and were useful for developing better image processing algorithms [2]–[4]. Other simulation tools have attempted to explicitly model the steps in a microarray experiment, including printing, hybridization, dye effects, and scanning [5], [6]. As with many of the early statistical simulations [7]–[10], 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 populations [11]–[14]. Moreover, none of the existing simulation tools was designed to focus on the biological diversity related to such important outcomes as treatment response or survival.

To address this gap, we developed the `Umpire` package, which incorporates a heterogeneous model consistent with the multiple hit theory of carcinogenesis [15], [16]. Our package uses latent variables to simulate the connections between gene expression and either binary or time-to-event outcomes. Latent variables, also called hidden variables, are usually inferred from other variables rather than being observed directly [17]. For example, the latent variables in our simulation can be cancer subtypes that correspond to different survival rates, or biomarker expression levels that are linked with different treatment effects.

Advances in high-throughput technologies for gene expression measurement have spurred the development of analytical methods for dealing with the explosion of large amounts of biological data [18]–[21]. Three major questions addressed by these technologies are class comparison, class discovery, and class prediction [22]. The goal in class comparison is to find biological entities whose distributions differ among some pre-defined sample groups. Methods for class comparison include gene-by-gene t-tests or ANOVA coupled with multiple testing adjustments [23]. Class discovery involves performing unsupervised analyses to "learn" or "discover" subgroup structures in the data. The current state-of-the-art has evolved reasonable methods for class discovery, such as hierarchical clustering coupled with resampling techniques to assess robustness [24]. The goal of class prediction is to formulate gene signatures from a training data set, and then use the signatures to assign new samples to known classes [25]. The performance of class prediction methods is assessed with a rigorous approach involving independent testing data. There are some known pitfalls to building predictive models from microarray gene expression data that need special attention [26], [27]. Some studies have tried different strategies to boost the performance of class prediction [28], [29]. However, even though class prediction is the most important of the three problems, there is less agreement on the best (or even consistently good) methods for discovering complex models that can accurately predict biologically relevant outcomes such as treatment response or survival.

In spite of the difficulty in class prediction, there is an explosion of interest in biomarker research with the goal of incorporating biomarkers into drug development and leading to personalized medicine [30]–[37]. For example, about 30% of patients with breast cancer over-express the protein HER2, a member of the human epidermal growth factor receptor family. These patients do not respond to standard therapy, but benefit from Herceptin treatment in combination with chemotherapy [38]. This example illustrates the potential utility of biomarkers for patient selection. By selecting patients based on their biomarker profiles, we hope to enrich the pool of patients who have a greater probability of response to alternative treatment plans. If successful, this approach could lead to cheaper and faster clinical trials than the conventional ones.

Appropriate experimental designs are crucial to the biomarker discovery process. Sample size determination is a critical step in experimental design to ensure sufficient statistical power for making inferences about a population from a sample [39]. It is conceivable that the number of samples (typically between 100 and 300) included in most current studies is simply inadequate to learn effective predictive models. On one hand, the soundness of analytical tools cannot be evaluated accurately given the small sample size and the unknown "ground truth" of biology. On the other hand, biological changes can be masked by noise, which requires large number of samples in order to reveal the true signal. It is, however, extremely difficult to assess the possibility that more samples (and how many more) would convey sufficient predictive power. Although some progress has been made for binary classifiers [13], [40], [41], we do not have general theoretical methods 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 data sets. 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 data sets, of different sizes, with realistic biological properties, then we can use those data sets to evaluate proposed methods for class prediction. Using the `Umpire` simulation package, we can generate realistic data to help answer the questions above. In the following sections, we first elaborate the design of `Umpire` and the parameters we implemented in the current version. We then discuss results from two sets of simulations to demonstrate the use of `Umpire` for cancer subtype recovery and biomarker discovery, respectively.

II. Homogeneous gene expression model

We begin by describing the underlying statistical model for simulating gene expression data that is implemented in the `Umpire` package. The fundamental object is a "random-vector generator" (RVG), which represents a specific multivariate distribution from which random vectors can be generated.

A. Additive and Multiplicative Noise

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

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

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. Usually the microarray noise is considered a combination of additive and multiplicative components [42]. 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 contributing some level of signal at all genes. The noise model is represented in the `Umpire` package by the `NoiseModel` class. The object-oriented and modular design makes it possible to add more elaborate noise models in the future, such as those described by Nykter and colleagues [5].

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 hyper-parameters 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 the Affymetrix GeneChip® human 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, ξ, 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 ξ and standard deviation $0.3\,\xi$. The case $\xi = 1$ is special, since we take the standard deviation of the block size to be zero so all genes are independent. At the other extreme, $\xi = 1000$ simulates large networks involving many genes.

The correlation matrix (Ω_b) for a block b, has 1's on the diagonal and ρ_b or $-\rho_b$ 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 let θ denote the portion of negatively correlated genes within a block. In the simplest scenario, all genes in the same block have the same positive correlation ρ_b. In a more complicated scenario, $\theta = 0.5 - |x - 0.5|$ where x follows a beta distribution. Three types of x are considered: (1) $x \sim \text{Beta}(1,1)$, so θ is uniformly distributed between 0 and 0.5; (2) $x \sim \text{Beta}(5,5)$, so θ is likely to be close to 0.5; or (3) $x \sim \text{Beta}(0.5, 0.5)$, so θ is likely to be close to 0. Our pilot study showed that different θ's do not have a pronounced impact on the parameters of interest (data not shown). So, we only discuss the results obtained from $\theta = 0.5 - |x - 0.5|$ where $x \sim \text{Beta}(1,1)$.

The log expression values of genes within a block follow a multivariate normal (MVN) distribution. The mean vector is defined by μ_g as defined previously, and the covariance matrix Σ is defined as:

$$\Sigma_{i,j} = \Omega_{i,j} * \sigma_{g_i} * \sigma_{g_j}$$

where σ_{g_i} defines the standard deviation of gene i, which follows the inverse gamma distribution as described previously. More elaborate models can also be generated, by altering the variances or the correlation structure within the block.

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 whole blocks 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 [15] and extended by Alfred Knudson in 1971 [16]. 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 [43], 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. The absolute change of the mean expression values on log scale for a block of genes is given by $\Delta_g \sim \text{Gamma}(\alpha, \beta)$. Both parameters for this gamma distribution are set to 10 so that the absolute fold change on the log2 scale is 1, and the long tail on the right hand side of the distribution allows a few genes to have large fold changes. A gene in the changed block is randomly assigned to be up-regulated or down-regulated in cancer patients.

IV. Implementation

The statistical model that we have just described is implemented using S4 classes in the R statistical software programming environment. Version 1.2.3 of the `Umpire` package is available from the R repository at http://bioinformatics.mdanderson.org/OOMPA; detailed instructions on how to install the package can be found at http://bioinformatics.mdanderson.org/Software/

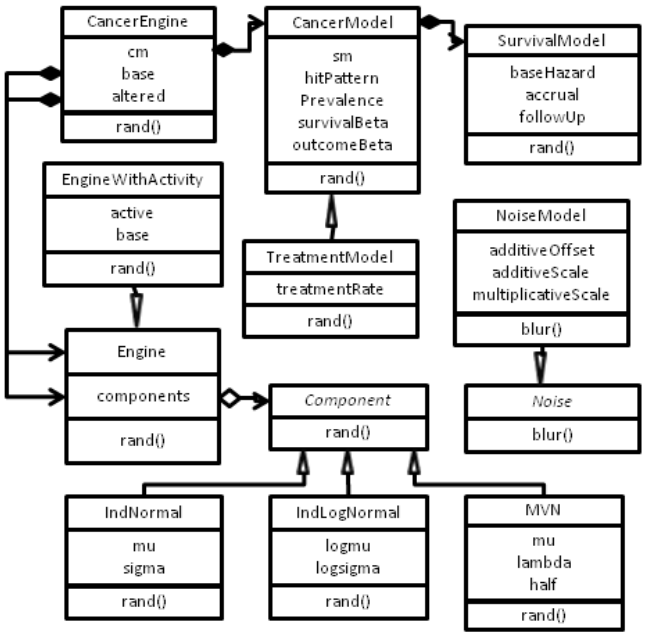

Fig. 1. UML diagram of classes in the Umpire package.

OOMPA. Figure 1 presents a diagram of the class structure using the Unified Modeling Language (UML). The main class, `CancerEngine`, contains one `CancerModel` and two `Engine` objects. The `CancerModel` object is used to simulate clinical data, including cancer subtypes, binary outcomes, and survival times. This object contains a matrix of hit patterns and a vector of prevalences that characterize the cancer subtypes being simulated.

The basic survival model assumes that the survival times follow an exponential distribution; other survival distributions can be simulated by deriving a subclass of `SurvivalModel`. The `rand()` method for `SurvivalModel`s takes an optional extra parameter, β, that represents a vector of logarithmic hazard ratios to modify the survival distributions for individual patients depending on the latent pattern of hits and, possibly, the treatment they receive.

Each `Engine` is used to simulate vectors of gene expression data. An `Engine` is a list of components; for the simulations described in this paper, we use a combination of `IndependentNormal` and `MVN` (multivariate normal) components. Additional components can be derived from the abstract `Component` class to simulate data from other distributions. For example, one might use Poisson distributions or negative binomial distributions to simulate the kinds of count-based gene expression data that are produced by next generation sequencing technologies. The pair of `Engine` objects in a `CancerEngine` represent the baseline gene expression (with no hit) and the altered gene expression that occurs in the presence of a hit; which expression pattern is used for any simulated sample depends on the subtype and hit pattern generated by the associated `CancerModel`.

Noise is applied to simulated gene expression data, using the `blur()` method, after the "true" signal is simulated. In the simulation presented here, we use a straightforward model of additive and multiplicative white noise. The general design, however, allows for the incorporation of more elaborate noise models by deriving additonal subclasses of the abstract `Noise` class.

The block structure is only indirectly specified by the class structure. For the simulations presented here, we implement it by constructing `Engine` objects consisting of `MVN` components with block sizes drawn from an appropriate distribution.

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

V. Simulation Results

To illustrate the usage of the `Umpire` package, we performed two sets of simulation of microarray data with associated survival data.

A. Cancer Subtype Recovery

In the first simulation, we assumed that there are 20 possible hits (H1 to H20), and that 5 hits at a time define a cancer subtype. We also 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 assumed that there were 100 additional 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.

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 [44] 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

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
H1	4	0.291	0	8	10
H2	2	0.366	0	5	11
H3	1	0.090	0	3	11
H4	0	0.278	0	1	0
H5	1	1.428	0	2	2
H6	3	0.313	0	1	2
H7	0	0.496	0	0	0
H8	1	-0.428	1	5	13
H9	3	-2.135	6	34	40
H10	0	0.631	2	1	0
H11	1	0.047	17	38	44
H12	2	0.422	0	13	27
H13	2	1.062	1	7	12
H14	0	1.433	0	2	0
H15	2	2.514	0	6	15
H16	1	-0.384	0	3	3
H17	1	-0.841	1	10	14
H18	2	0.299	0	13	16
H19	2	1.358	10	25	32
H20	2	-1.674	6	35	41
FP	0	0.000	30	37	61

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 2 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 gene memberships in cancer-related gene blocks, with white representing genes belonging to non-cancer-related blocks, which are false positives. When using 100 samples, not all patients with different cancer subtypes are well separated. We observe distinct gene expression patterns in patients with subtype 1 and 5, but not in other patients. On the gene level, the 74 significant genes come from 8 cancer-related gene blocks. With 500 samples, all six cancer subtypes are well separated by clustering, and their distinct gene expression patterns are visible in the heatmap. On the gene level, the 354 significant genes cover 16 out of 20 cancer-related gene blocks. In both heatmaps, the false positive genes, represented by the white color bar, are recognizable by their lack of correlation with other selected genes.

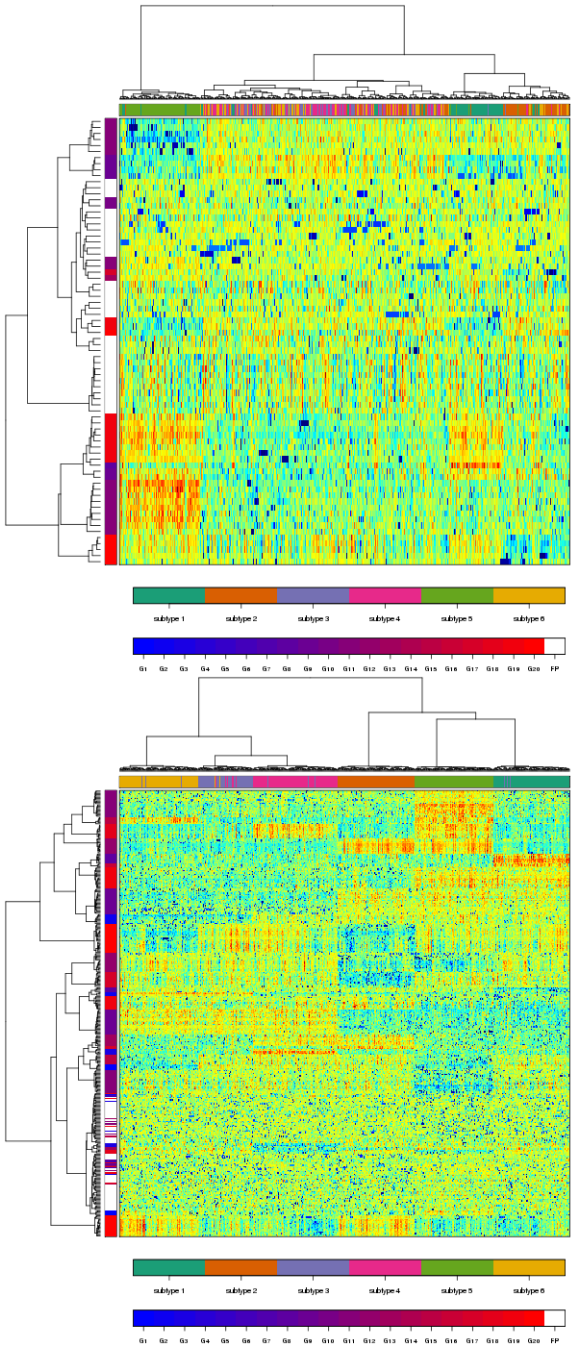

Fig. 2. Heatmaps of the significant genes at FDR = 20% using 100 (top) or 500 (bottom) samples.

B. Patient Selection In Clinical Trials

The second set of simulations involves biomarker identification and patient selection during clinical trials. We assume that a randomized clinical trial is conducted with two arms with equal probability to compare the performance of some standard therapy with a potentially better alternative therapy. The hazard ratio between the alternative treatment and the standard treatment in the full population is called HR_{trt}. We simulated time-to-event outcome, which might represent overall survival, progression-free survival, or other similar clinically relevant endpoints, between the two arms. Note that other types of endpoints can be easily added to the `Umpire` package. A latent variable L indicates whether each patient is marker-positive ($M+$) or marker-negative ($M-$). The time-to-event outcome is linked with the treatment and the latent variable. Only $M+$ patients will benefit from the alternative treatment.

Genes in five correlated blocks out of 100 total blocks are

differentially expressed between $M+$ and $M-$ groups. The goal is to identify some complex (probably multivariate) marker that separates the initial patient population into two groups ($M+$ and $M-$), such that the hazard ratio between treatment arms in the $M+$ group, HR_{M+}, is a substantial improvement over the hazard ratio HR_{Trt} in the full population.

We simulated survival using an exponential baseline hazard function with a median progression-free survival time of 18 weeks. The true benefit in the patients who have the marker is simulated as $HR_{M+} = 0.55$. Assuming that 30% of patients contain this marker, as in the example of HER2 described above, we simulated different sizes of patient cohorts ranging from 100 to 1500. Each scenatio was simulated 10 times for variance estimation. We also simulated independent testing data sets of size 200.

For each training data set, we performed K-means clustering [45] on each gene with $K = 2$. To select potential biomarkers, we searched for genes whose two groups corresponded to different hazard ratios between the two treatment arms. We fit gene-by-gene univariate Cox proportional hazards models. The p-values corresponding to the interaction term between treatment and the gene grouping are further modeled using the BUM model to estimate the FDR. With FDR cutoff 20%, we selected significant genes for each set of training data. Similarly, K-means clustering was performed on each gene in the test data sets. We then calculated the percentage of significant genes voting for $M+$ as a multivariate predictor that a patient is $M+$. Figure 3 shows receiver operating characteristic (ROC) curves [46] of the predictions in the testing data sets for different size training data sets. We observe that more training samples yield more accurate predictions. In this simulation, the area under the ROC curve (AUC) is larger than 0.9 when the number of patients is at least 500.

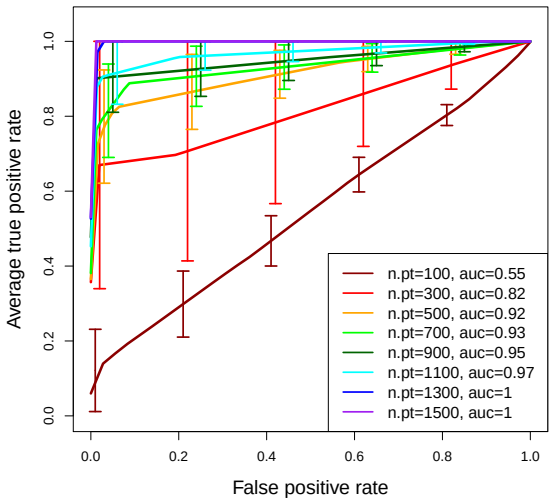

Fig. 3. ROC curves for patient selection with markers identified from different sized patient cohorts. The vertical bars correspond to standard error from 10 simulation, and the AUCs are shown in the legend.

VI. Conclusion

We have described the `Umpire` R package and shown that it can be used to simulate microarray data that is related to survival outcomes in complex ways. In our simulation, many assumptions are based on our extensive experience derived from working with real Affymetrix GeneChip® data sets. We recognize that some of the modeling assumptions that we used might seem simplified considering the complex biology. However, one advantage of implementing `Umpire` with S4 classes in R is that the package is flexible enough to allow easy addition of components representing alternative models of gene expression.

The two sets of simulations that we have presented, which use a plausible set of biologically meaningful parameters, suggest that both class discovery studies and biomarker discovery studies looking for signatures to predict time-to-event outcomes may need more than the 100–300 samples that have frequently been used in practice. In order to elucidate the true subgroup structure, our first simulation required about 500 samples. In order to discover biomarker signatures that could identify a subgroup of patients more likely to respond to an alternative treatment, our second simulation also0 required at least 500 patients. In this context, it is interesting to note that the ongoing effort of The Cancer Genome Atlas (TCGA) to apply comprehensive high-throughput molecular biology techniques to a variety of different cancers intends to study about 500 samples of each type [47].

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 2, 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, there are approximately five non-random PCs; the appropriate number of PCs in a real data set could potentially be estimated from a scree plot of the amount of variance explained by each PC. The selected PCs could then be used as predictors in a Cox proportional hazards model. 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; these could also be tested in a Cox model. In this case, the obvious next step would be to develop a robust multi-category classifier.

We do not pursue these approaches further in the

current paper. However, the `Umpire` 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 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] J. Zhang and K. R. Coombes, "UMPIRE: Ultimate microarray prediction, inference, and reality engine," in *BIOTECHNO 2011, The Third International Conference on Bioinformatics, Biocomputational Systems and Biotechnologies*, 2011, pp. 121–125.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

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

[8] 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.

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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

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

[16] 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.

[17] D. Borsboom, G. Mellenbergh, and J. van Heerden, "The theoretical status of latent variables," *Psychological Review*, vol. 10, no. 2, pp. 203–219, 2003.

[18] N. Belacel, Q. Wang, and M. Cuperlovic-Culf, "Clustering methods for microarray gene expression data." *OMICS.*, vol. 10, no. 4, pp. 507–31, 2006.

[19] W. Kong, C. Vanderburg, H. Gunshin, J. Rogers, and X. Huang, "A review of independent component analysis application to microarray gene expression data." *Biotechniques*, vol. 45, no. 5, pp. 501–20, 2008.

[20] J. Chen, "Key aspects of analyzing microarray gene-expression data," *Pharmacogenomics*, vol. 8, no. 5, pp. 473–82, 2007.

[21] G. Hatfield, S. Hung, and P. Baldi, "Differential analysis of dna microarray gene expression data," *Mol Microbiol.*, vol. 47, no. 4, pp. 871–7, 2003.

[22] 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.

[23] S. Dudoit and M. van der Laan, *Multiple Testing Procedures with Applications to Genomics*, ser. Springer Series in Statistics. New York, NY: Springer, 2008.

[24] 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.

[25] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield, and E. Lander, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, pp. 531–7, 1999.

[26] A. Dupuy and R. Simon, "Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting." *J Natl Cancer Inst.*, vol. 99, no. 2, pp. 147–57, 2007.

[27] S. Michiels, S. Koscielny, and C. Hill, "Prediction of cancer outcome with microarrays: a multiple random validation strategy." *Lancet*, vol. 365, no. 9458, pp. 488–92, 2005.

[28] J. Deutsch, "Evolutionary algorithms for finding optimal gene sets in microarray prediction." *Bioinformatics*, vol. 19, no. 1, pp. 45–52, 2003.

[29] R. Simon, M. Radmacher, K. Dobbin, and L. McShane, "Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification." *J Natl Cancer Inst.*, vol. 95, no. 1, pp. 14–8, 2003.

[30] Q. Ye, L. Qin, M. Forgues, P. He, J. Kim, A. Peng, R. Simon, Y. Li, A. Robles, Y. Chen, Z. Ma, Z. Wu, S. Ye, Y. Liu, Z. Tang, and X. Wang, "Predicting hepatitis B virus-positive metastatic hepatocellular carcinomas using gene expression profiling and supervised machine learning." *Nat Med.*, vol. 9, no. 4, pp. 416–23, 2003.

[31] D. Danila, K. Pantel, M. Fleisher, and H. Scher, "Circulating tumors cells as biomarkers: progress toward biomarker qualification," *Cancer J.*, vol. 17, no. 6, pp. 438–50, 2011.

[32] T. Sigdel and M. Sarwal, "Recent advances in biomarker discovery in solid organ transplant by proteomics." *Expert Rev Proteomics.*, vol. 8, no. 6, pp. 705–15, 2011.

[33] J. Kalinina, J. Peng, J. Ritchie, and E. Van Meir, "Proteomics of gliomas: initial biomarker discovery and evolution of technology." *Neuro Oncol.*, vol. 13, no. 9, pp. 926–42, 2011.

[34] P. Rakowska and M. Ryadnov, "Nano-enabled biomarker discovery and detection." *Biomark Med.*, vol. 5, no. 3, pp. 387–96, 2011.

[35] J. Ross, "Biomarker-based selection of therapy for colorectal cancer." *Biomark Med.*, vol. 5, no. 3, pp. 319–32, 2011.

[36] S. Dupouy, N. Mourra, V. Doan, A. Gompel, M. Alifano, and P. Forgez, "The potential use of the neurotensin high affinity receptor 1 as a biomarker for cancer progression and as a component of personalized medicine in selective cancers." *Biochimie.*, vol. 93, no. 9, pp. 1369–78, 2011.

[37] E. Galanis, W. Wu, J. Sarkaria, S. Chang, H. Colman, D. Sargent, and D. Reardon, "Incorporation of biomarker assessment in novel clinical trial designs: personalizing brain tumor treatments." *Curr Oncol Rep.*, vol. 13, no. 1, pp. 42–9, 2011.

[38] J. Ross, E. Slodkowska, W. Symmans, L. Pusztai, P. Ravdin, and G. Hortobagyi, "The HER-2 receptor and breast cancer: ten years of targeted anti-HER-2 therapy and personalized medicine." *Oncologist*, vol. 14, pp. 320–368, 2009.

[39] R. Fisher, *The Design of Experiments*. Macmillan, 1971.

[40] 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.

[41] 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.

[42] J. Kim, D. Shin, and Y. Lee, "Effect of local background intensities in the normalization of cDNA microarray data with a skewed expression profiles." *Exp Mol Med.*, vol. 34, no. 3, pp. 224–32, 2002.

[43] D. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society*, vol. 34, no. 2, pp. 187–220, 1972.

[44] S. Pounds and S. Morris, "Estimating the occurrence of false positives and false negatives in microarray studies by approximating and partitioning the empirical distribution of p-values." *Bioinformatics*, vol. 19, pp. 1236–42, 2003.

[45] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press*, 1967, pp. 281–297.

[46] M. Zweig and G. Campbell, "Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine." *Clin Chem.*, vol. 39, no. 4, pp. 561–77, 1993.

[47] L. Chin, W. Hahn, G. Getz, and M. Meyerson, "Making sense of cancer genomic data," *Genes and Development*, vol. 25, pp. 534–555, 2011.