Budgeted Transcript Discovery: A Framework For Joint Exploration And Validation Studies

Sheehan Khan and Russell Greiner
Department of Computing Science
University of Alberta
Edmonton, Canada
{sheehank,rgreiner}@ualberta.ca

***Abstract*—This paper presents the *budgeted transcript discovery problem* (BTD): deciding how to spend a given research budget collecting data, using a combination of microarrays and PCRs, to discover which transcripts are differentially expressed with respect to a given phenotype. We present algorithms that address this task by sequentially analyzing the data collected so far, to decide which data would be most informative to collect next. We provide empirical studies that demonstrate their effectiveness.**

Keywords*—*Microarray, feature selection, association studies

I. Introduction

A microarray observes the expression levels of all of the genes and transcripts of a sample – *i.e.*, the entire transcriptome.[1] Many transcriptomics researchers use such microarrays for exploratory studies to determine which genes are "biomarkers", *i.e.*, are individually associated with some specific biologically interesting phenotype. Unfortunately, different microarray studies, investigating the same phenotype, often produce very different results [1] – even if the same tissue samples are arrayed by different labs [2].

The conflict between studies can be attributed to both statistical issues arising from (1) analyzing tens of thousands of genes with only a handful of observations [3], and (2) methodological decisions such as choice of summary statistics and criterion for deciding differential expresion [4]. To see that issue (1) is not going away: note that the number of microarray datasets submitted to the GEO microarray database [5] has been steadily increasing, with approximately 10,000 last year; Figure 1[top]. However, Figure 1[bottom] shows that the number of microarrays used per dataset has not increased over time: essentially all have less than 100 microarrays and the majority have only 10–12. This trend suggests the statistical issues of microarray studies will remain.

To alleviate these issues, some researchers will perform a follow up validation study using PCR (polymerase chain reaction) to confirm that the genes implicated by the microarray study truly are differentially expressed [6]. However, there is yet to be a consensus as to which genes require confirmation, what confirmation will be, nor how it will be achieved [7], [8].

[1]Below, we will just use "genes", with the understanding that this term also includes transcripts.

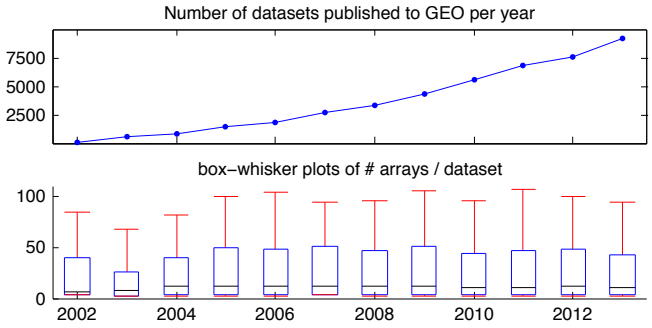

Fig. 1. [top] Summary of the dataset submitted to GEO per year. [bottom] Box and whisker plots for the number of microarrays per dataset.

This leads to the *Budgeted Transcript Discovery (BTD)* problem. The goal of BTD is to provide a framework within which microarray and PCR studies can be combined to objectively identify the genes associated with a phenotype. Thus, we address the aforementioned methodological issues, and present a solution to the previously open problem of defining a strategy for validating microarray studies with follow up confirmation via PCR.

We begin by formally defining our BTD problem in Section I-A. Section I-B presents approaches to similar problems from statistics and computing science, which we will later use as building blocks for our BTD algorithms in Section II. Section III compares these algorithms experimentally.

A. Formal Model

We assume we have a set of genes $\mathcal{G} = \{g_1, \ldots, g_N\}$ and wish to find the subset that are relevant (*i.e.*, "biomarkers") w.r.t. some binary phenotype of interest – *e.g.*, corresponding to biopsies from breast cancer tumors, some of which responded to a treatment (labeling that instance as "+"), or not ("−"). We also have a set of instances, each with the associated +/− label. Furthermore, we are able to use both microarrays and PCRs to observe the values of these genes, from any instance. The relevance of gene g_i is assessed based on the effect size of its expression values, $\Delta_i^{(\chi)} = \frac{\left|\mu_{i,+}^{(\chi)} - \mu_{i,-}^{(\chi)}\right|}{\sigma_i^{(\chi)}}$ for each type of observation χ (either $\chi = M$ for microarray or $\chi = P$ for PCR), where $\mu_{i,+}^{(\chi)}$ (resp., $\mu_{i,-}^{(\chi)}$) is the mean expression value

for gene i for the $+$instances (resp., $-$instances), and we also assume a common standard deviation $\sigma_i^{(\chi)}$.

Naturally features will have different effect sizes for the two observation types, with $\Delta_i^{(P)} > \Delta_i^{(M)}$, as PCR is more accurate than microarray. We define the set of relevant genes to be $R = \{g_i : \Delta_i^{(\chi)} \geq \Delta_{min}^{(\chi)}\}$, for the specified minimum values $\Delta_{min}^{(M)}$ and $\Delta_{min}^{(P)}$.

Definition [*Budgeted transcript discovery problem*] Given a total experimentation budget B, collect data using a series of microarrays (each of which produces an observation of the expression of each gene, for a specified individual, at the cost C_M) and PCR's (each of which produces an observation of the expression of a single specified gene, for a specified individual, at cost 1). The data will be collected sequentially, in that the results from each observation may be used to select which observation to perform later. After spending the budget B collecting information, the goal is to return the relevant set of genes $R \subset \mathcal{G}$.

We assume that algorithms for this problem know the sensitivities ($\Delta_{min}^{(M)}$, $\Delta_{min}^{(P)}$) for the technologies used. They also know (a reasonable estimate of) the number of relevant genes $|R|$ that they may use to tune their parameters.[2] We evaluate an algorithm based on its F1 score,

$$F_1(\text{TP}, \text{FP}, |R|) \quad = \quad \frac{2\,\text{TP}}{|R| \;+\; \text{TP} \;+\; \text{FP}}\,, \qquad (1)$$

where TP and FP denote the counts of relevant and irrelevant genes in the returned set, respectively. □

We chose the F1 measure for two reasons. (1) It has the desired property of rewarding the discovery of relevant genes and penalizing both false positives and false negatives. (2) This measure is good at penalizing degenerate behaviour, which is relevant as, in most situations, there will be relatively few relevant genes, $|R| \ll |\mathcal{G}|$.

B. Related Problems

The problem of testing if a *single* gene is relevant, corresponds to the sequential hypothesis testing problem, of collecting data until one of two hypotheses, $H0$ versus $H1$, can be decided at a pre-fixed confidence. The sequential probability ratio test (SPRT) [9] solves this problem optimally in the sense that provably no other algorithm can make the decision with the same confidence and collect less data in expectation. Unfortunately no theoretical work has been done to analyze the case of running N SPRTs with a common budget – *i.e.*, in the event that the budget is insufficient for all SPRTs to terminate, it is unclear how to divide the budget across the genes.

The m-best arm identification problem [10]–[12] shares much of the problem structure with our BTD problem. An algorithm for m-best arm identification is given N distributions (arms), with the goal of finding the m-best arms, *i.e.*, the distributions with the largest mean values. There is an obvious mapping of this problem to our BTD, but a key difference is that we evaluate algorithms based on the F1 score, while m-best arm algorithms are constructed to solve the problem *exactly* with a high confidence – *i.e.*, an algorithm only gets a point if it correctly returns exactly the m-best arms; finding 99 of the top 100 arms in a set of 1000 is as bad as finding nothing at all.

Note also that neither the SPRT nor the m-best arm identification tasks deal with the challenge of switching between collecting data using microarrays to using PCRs, with their differing cost and accuracy models.

[2]This assumption is plausible as the biologists involved in the study typically know the difficulty of the study without knowing the outcome *a priori*.

II. BTD Algorithms

This section presents several algorithms for the BTD problem. Each algorithm is allowed to perform any sequence of microarrays and PCRs within the budget constraint. An algorithm may use any observed values it wishes in order to decide which observation to collect next. To reduce the complexity of the algorithms, we force them to collect data from both classes equally – *i.e.*, when collecting data, the algorithm may either use two PCRs to observe the expression value of a specified gene g_i from both a $+$instance and for a $-$instance, or it may use two microrrays to observe the expression values of all genes from a $+$instance and from a $-$instance.

Keeping with common practice in bioinformatics, we construct our algorithms under the assumption that gene expression values for each gene (for each category $+$ vs $-$) are drawn from a normal distribution [13]–[15]. For compactness in the algorithm descriptions, we assume that each algorithm keeps track of all sufficient statistics required to compute sample estimates of effect sizes, $\hat{\Delta}_i^{(M)}$ and $\hat{\Delta}_i^{(P)}$, and the log-likelihood ratio,

$$\Lambda_i \quad = \quad \log\left(\frac{P\left(i \in R \mid \hat{\Delta}_i^{(P)}\right)}{P\left(i \notin R \mid \hat{\Delta}_i^{(P)}\right)}\right).$$

Note we only use the PCR data when computing the log-likelihood ratios for reasons that will be come apparent in Section II-B when we present our mSPRT algorithm.

A. Round Robin (RR & RR+RR)

We begin by considering naive algorithms that make no attempt to exploit the sequential nature of the problem. The first algorithm, RR, spends all the budget collecting microarrays then applies a simple threshold decision for relevance. The second algorithm, RR+RR, spends part of the budget collecting microarrays to eliminate the obviously irrelevant genes and then spends the remaining budget to collect an equal amount of PCR on the remaining genes. We call these algorithms RR and RR+RR, in reference to the round robin nature of the data collection.

We tune the parameters for these algorithms by observing that if $\hat{\Delta}$ is computed from n samples, then $\hat{\Delta}\sqrt{n/2}$ follows a non-central t-distribution. Using the appropriate values of Δ in Eqn 2, it is straightforward to tune the parameters to maximize the expected F1 score (Eqn 1) of RR and RR+RR. For microarrays, we use $\Delta = \frac{1}{2}\Delta_{min}^{(M)}$ for the irrelevant genes and $\Delta = \frac{3}{2}\Delta_{min}^{(M)}$ for the relevant genes.

$$\nu = 2(n-1) \quad , \quad \lambda = \Delta\sqrt{n/2} \tag{2}$$
$$P\left(\hat{\Delta} < \tau \mid \nu, \lambda\right) = T_{\nu,\lambda}\left(\tau\sqrt{\frac{2}{n}}\right) - T_{\nu,\lambda}\left(-\tau\sqrt{\frac{2}{n}}\right)$$

where $T_{\nu,\lambda}(\cdot)$ is the CDF for the non-central t-distribution with ν degrees of freedom and non-centrality parameter λ.

Algorithm 1 RR % uses τ [Eqn 1, 2]

collect $\lfloor B/C_M \rfloor$ microarrays
Return $\hat{R} = \left\{g_i : \hat{\Delta}_i^{(M)} > \tau\right\}$

Algorithm 2 RR+RR % uses n, τ_1, τ_2 [Eqn 1, 2]

collect n microarrays
$uncertain = \{g_i : \hat{\Delta}_i^{(M)} > \tau_1\}$ and $h = \frac{\text{budget left}}{|uncertain|}$
for $g_i \in uncertain$ **do**
    collect h observations of gene i
Return $\hat{R} = \left\{g_i : \hat{\Delta}_i^{(P)} > \tau_2 \text{ AND } g_i \in uncertain\right\}$

B. Modified Sequential Probability Ratio Test (mSPRT)

While RR and RR+RR are quick and easy solutions, notice that they do not model the sequential nature of the problem. An obvious improvement to RR+RR would be to run a SPRT for each gene in the uncertain set, with the additional criteria that we terminate the SPRT for each gene if it collects as many observations as RR+RR. Thus, we can ensure that all decisions are made with the same confidence as RR+RR but we potentially spend less budget. The remaining budget can then be used to refine decisions about genes that are closest to the decision boundary.

To set the parameters for this modified SPRT (mSPRT) algorithm, we compute the true and false positive probabilities from RR+RR, denoted p_{TP} and p_{FP} respectively, and use them to set the bounds,

$$\gamma_0 = \log\left(\frac{1-p_{TP}}{1-p_{FP}}\right) \text{ and } \gamma_1 = \log\left(\frac{p_{TP}}{p_{FP}}\right) . \tag{3}$$

C. Lower Upper Confidence Bounds (LUCB)

While it is easy to motivate mSPRT, as it definitely beats RR+RR, it may also be limited in that it only seeks to beat RR+RR. An algorithm could potentially do better by spending less budget on the initial SPRT phase and more budget on the latter greedy phase. To construct such an algorithm, we modify the LUCB1 algorithm for the best m-arm identification problem [11] to create our mLUCB. This algorithm operates by splitting the genes in the uncertain set based on their estimates of $\hat{\Delta}^{(P)}$, above or below threshold α, and then collects observations of the genes that are close to the decision boundary. To prevent the algorithm from spending too much effort on the hard-to-classify genes close the boundary, we use optimistic confidence bounds to augment the estimates. Confidence bounds are controlled by the C parameter; where large C encourages the algorithm to behave more like RR+RR and smaller C cause the algorithm to behave more greedily.

Algorithm 3 mSPRT % uses n, τ_1 [Eqn 1, 2]; γ_0, γ_1 [Eqn 3]

collect n microarrays
$uncertain = \{g_i : \hat{\Delta}_i^{(M)} > \tau_1\}$ and $h = \frac{\text{budget left}}{|uncertain|}$
for $g_i \in uncertain$ **do**
    $m_i = 0$ and $\Lambda_i = 0$
    while $\gamma_0 < \Lambda_i < \gamma_1$ AND $m_i < h$ **do**
        collect an observation of gene g_i
while budget left **do**
    $i = \arg\min_{i \in uncertain} |\Lambda_i - \gamma_1|$
    collect an observation of gene g_i
Return $\hat{R} = \{g_i : \Lambda_i > \gamma_1 \text{ AND } g_i \in uncertain\}$

Algorithm 4 mLUCB(α, C) % uses n, τ_1 [Eqn 1, 2]

collect n microarrays
$uncertain = \{g_i : \hat{\Delta}_i^{(M)} > \tau_1\}$
for $g_i \in uncertain$ **do**
    $m_i = 2$
    collect two observations of gene g_i
$t = 0$
while budget left **do**
    $t = t + 1$
    for $g_i \in uncertain$ **do**
        $b_i = \sqrt{\frac{C}{m_i}\log(t)}$
    $i = \arg\min_i \left\{\hat{\Delta}_i^{(P)} + b_i : \hat{\Delta}_i^{(P)} < \alpha\right\}$
    $j = \arg\min_j \left\{\hat{\Delta}_j^{(P)} - b_j : \hat{\Delta}_j^{(P)} > \alpha\right\}$
    collect an observation of gene g_i, and gene g_j
    $m_i = m_i + 1$ and $m_j = m_j + 1$
Return $\hat{R} = \left\{g_i : \hat{\Delta}_i^{(P)} > \alpha \text{ AND } g_i \in uncertain\right\}$

III. Experiments

One way to compare the effectiveness of our various algorithms is by running them on some historical datasets, which include both microarray data for a set of instances, and also PCR data over *all* of the genes for those instances. Unfortunately, there are no such datasets, as it is prohibitively expensive to collect that much PCR data.

We instead used a realistic synthetic testbed, using the microarray data from GSE41726[3] [16], involving 134 microarrays over 41,000 genes. Keeping with our modelling assumptions, we set normal distributions for each class for each gene such that for each gene $\Delta_i^{(M)}$ matches the observed value in the dataset. To model that PCR is more accurate that microarrays, we then set $\Delta_i^{(P)} = 2\Delta_i^{(M)}$ for all genes.

For the experiment we define the relevant genes to be $R = \{g_i : \Delta_i^{(M)} \geq 1\}$, which corresponds to 141 genes. We set the budget to correspond to 20 microarrays, $B = 20C_M$. We will consider the effect of different microarray vs PCR costs by scaling the C_M parameter.

We tune parameters for RR, RR+RR, and mSPRT, using Eqn 1 and 2 and enumerating all possible outcomes, under the assumption that $|R| = 150$. To tune mLUCB, we consider

[3] http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE41726

the simplified case where $\Delta^{(M)} = 1$ and $\Delta^{(P)} = 2$ for all relevant genes, and $\Delta^{(M)} = \Delta^{(P)} = 0$ for the irrelevant genes. Figure 2[left] shows the comparison in terms of the expected F1 score. We can see that mLUCB performs the best when C_M is low. However, as C_M increases, mSPRT does significantly better. Figure 2[right] shows the precision (*a.k.a.* true positive rate) of the algorithms, which makes it clear that the RR, RR+RR, and mLUCB algorithms try to achieve a good F1 score by returning a large number of genes in the hope of selecting a few of the relevant ones by chance, while mSPRT tries to achieve a good F1 score by returning relatively few genes, but ones that (it thinks) are relevant. We believe that mSPRT's behaviour is closest to what bioinformaticians will want.

To explain the slight dip in mSPRT's performance, at around $C_M \approx 600$: This dip occurs because the increased C_M allows mSPRT to perform more PCRs, meaning it explored some genes with $\Delta_i^{(P)} \approx \Delta_{min}^{(P)}$ and thus suffered some false positives. However as C_M further increased, mSPRT then had sufficient budget to correct those false positives.

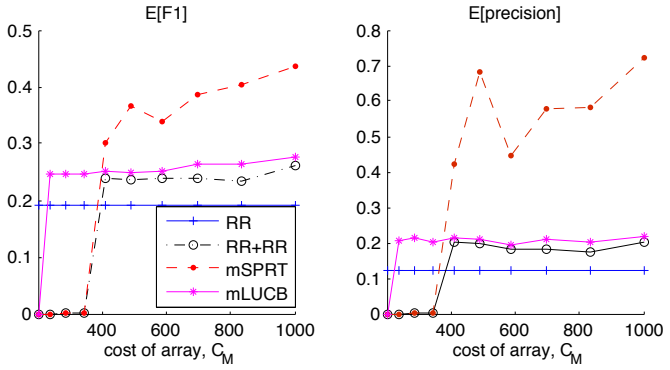

Fig. 2. Comparison of the algorithms; [left] F1 score, [right] precision.

IV. Conclusion

Future Work: For this work we considered algorithms that had access to a reasonable estimates of $|R|$. A natural extension would be to design a meta-algorithm that can estimate $|R|$ while collecting the initial microarrays, which it could then use to tune the parameters on the fly, and decide whether it should run mSPRT or mLUCB.

Our algorithms implicitly incorporate the naive-bayes assumption: that the expression values for the genes are independent, given the class label. We considered extending the algorithms to model interactions between the genes, as we know that gene expressions are correlated with one other. However, this would involve learning a correlation structure for all pairs (or worse, all r-tuples) of features; due to the large number of features, this approach would be further plagued with statistical issues. We believe it is better to develop algorithms for BTD under the naive bayes assumption as they cannot be mislead by believing an incorrect correlation structure.

Contributions: This paper has defined and analyzed the budgeted transcript discovery (BTD) problem – *i.e.*, how to effectively and efficiently identify which genes are differentially expressed wrt a phenotype. One benefit of the BTD framework is that it circumvents the ambiguous methodological choices that are present in traditional microarray studies, such as how to decide when a gene is differentially expressed. Another advantage of the BTD problem is that it provides a solution to the irreproducibility of microarray studies due to statistical reasons, as it provides a principled manner to confirm results with PCR.

We have presented several well-motivated approaches to solving the BTD problem. If microarrays are affordable relative to PCRs (in terms of the number of PCR experiments possible for the cost of an microarray C_M), then we found that our mLUCB algorithm provides the best solution. However, if the microarray/PCR cost ratio is high (≥ 400), then our mSPRT algorithm is able to leverage the additional PCR observations to be much more precise about the genes it returns, and thus performs significantly better.

Acknowledgment

The authors thank Mitacs, Metabolomic Technologies Inc., NSERC and Alberta Innovates Technology Fund, for helping to fund this research.

References

[1] S. Michiels, S. Koscielny, and C. Hill, "Prediction of cancer outcome with microarrays a multiple random validation strategy," *Lancet*, vol. 365, pp. 488–492, 2005.

[2] J. P. Ioannidis *et al.*, "Repeatability of published microarray gene expression analyses." *Nat. Genet.*, vol. 41, no. 2, pp. 149–55, 2009.

[3] L. Ein-Dor, O. Zuk, and E. Domany, "Thousands of samples are needed to generate a robust gene list for predicting outcome in cancer," *Proc. Natl. Acad. Sci.*, vol. 103, no. 15, pp. 5923–5928, 2006.

[4] A.-L. Boulesteix and M. Slawski, "Stability and aggregation of ranked gene lists." *Brief. Bioinform.*, vol. 10, no. 5, pp. 556–68, 2009.

[5] R. Edgar, M. Domrachev, and A. E. Lash, "Gene Expression Omnibus: NCBI gene expression and hybridization array data repository." *Nucleic Acids Res.*, no. 1, pp. 207–10.

[6] N. A. Campbell and J. B. Reece, *Biology*, 8th ed. Benjamin Cummings, 2008.

[7] J. C. Rockett and G. M. Hellmann, "Confirming microarray data–is it really necessary?" *Genomics*, vol. 83, no. 4, pp. 541–9, 2004.

[8] D. B. Allison *et al.*, "Microarray data analysis: from disarray to consolidation and consensus." *Nat. Rev. Genet.*, vol. 7, no. 1, pp. 55–65, 2006.

[9] A. Wald, "Sequential Tests of Statistical Hypotheses," *Ann. Math. Stat.*, vol. 16, no. 2, pp. 117–186, 1945.

[10] E. Even-Dar, S. Mannor, and Y. Mansour, "Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems," *J. Mach. Learn.*, vol. 7, pp. 1079–1105, 2006.

[11] S. Kalyanakrishnan *et al.*, "PAC Subset Selection in Stochastic Multi-armed Bandits," *ICML*, 2012.

[12] S. Bubek, T. Wang, and N. Viswanathan, "Multiple Identifications in Multi-Armed Bandits," in *ICML*, 2013.

[13] P. Baldi and a. D. Long, "A Bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes." *Bioinformatics*, vol. 17, no. 6, pp. 509–19, 2001.

[14] B. Efron *et al.*, "Empirical Bayes Analysis of a Microarray Experiment," *J. Am. Stat. Assoc.*, vol. 96, no. 456, pp. 1151–1160, 2001.

[15] G. K. Smyth, "Linear models and empirical bayes methods for assessing differential expression in microarray experiments." *Stat. Appl. Genet. Mol. Biol.*, vol. 3: Article3, 2004.

[16] C. Stretch *et al.*, "Effects of sample size on differential gene expression, rank order and prediction accuracy of a gene signature." *PLoS One*, no. 6, p. e65380.