IDENTIFYING GENE EXPRESSION PATTERNS ASSOCIATED WITH INCREASED SURVIVAL TIME ACROSS CANCERS

BACKGROUND

Aside from predicting drug efficacy, genetic biomarkers also function as important tools for cancer treatment. For example, by examining gene expression differences in tumors from different breast cancer patients and comparing their survival, it may be possible to learn what genetic differences lead to better breast cancer prognosis. However, performing this analysis *across* different types of cancers will hopefully identify genetic patterns that contribute most to cancer patient survivorship in general. If multiple cancers can be shown to function similarly at a genetic level, then it may be possible to repurpose a drug therapy developed for one cancer to another.

Unfortunately, this is a complex task because humans have over 20,000 genes. Simultaneously examining expression in this many genes for thousands of patients proves to be quite challenging. Therefore, we will utilize machine learning approaches to handle this complexity.

In this project, we apply supervised machine learning methods to publicly available cancer gene expression data to identify genes expression patterns that are associated with increased survival across multiple cancers (if such patterns exist). In our analysis, we will create a meta-dataset of gene expression data from breast, ovarian, and uterine cancer tumors paired to corresponding clinical data with survival outcome. Our goal is to identify a subset of genes that are associated with increased survival time across multiple cancers.

METHODS

DATA COLLECTION

The example data used for this project is breast, ovarian, and uterine cancer data downloaded from The Cancer Genome Atlas (TCGA). TCGA is a large project aimed at cataloging genetic mutations that are responsible for a variety of cancers by sequencing tumors and measuring gene expression from patients in both tumor and normal tissue [1]. It involves

multiple research institutions and is managed by the National Cancer Institute (NCI) and the National Human Genome Research Institute (NHGRI) which are part of the National Institutes of Health (NIH).

Training examples downloaded from TCGA were combined to create a meta-dataset of gene expression values paired to corresponding clinical data. Our total dataset is comprised of 1,455 tumor samples and expression data from 16,115 genes [Figure 1]. Only samples that had gene expression data available were selected for our analysis. Furthermore, any genes whose expression values were "NA" were removed from the feature list. Gene expression values were normalized to a scale of -1 to 1.

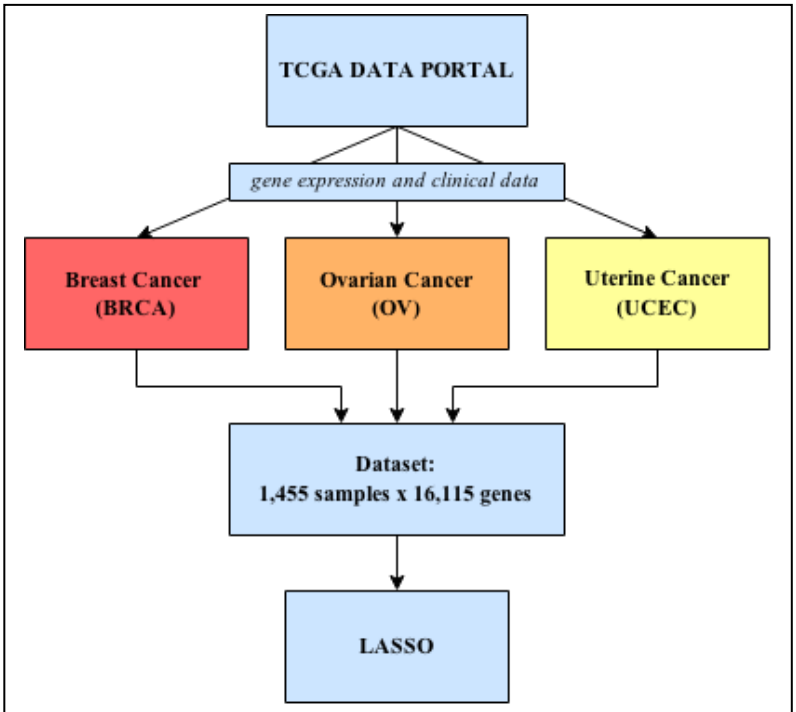

Figure 1: Schematic design for our study data.

Clinical information for each patient was determined by looking at time-to-event (survival analysis). Because different cancers can have widely different expected survival rates, we chose to define survival by examining each patient's time-to-event in relation to the mean survival time for the cancer type. If the patient survived longer than the mean survival time, their outcome was denoted +1 for better survival. Otherwise, the patient's outcome was denoted -1. In the combined dataset, 551 patients survived longer than the mean survival time, and 904 patients did not survive past the mean survival time. The mean survival times for each cancer type are listed in Table 1.

Table 1: Mean survival times per cancer type

Cancer Type	Mean Survival Time (Days)
Breast	1,106
Ovarian	1,032
Uterine	1,024

For algorithm implementation, the dataset was randomly split into two clusters: 2/3 used for training, 1/3 saved for final testing. To build the initial model, we used an even smaller subset of the original dataset to shorten run time and produce preliminary results for the milestone. Genes were filtered and selected by median absolute deviation; the value required being greater than .1. Selecting the subset this way increased the likelihood of choosing informative genes, because those features have more variation in expression values. This selection resulted in a set of 2,550 genes. Furthermore, the data subset was limited to a random sample of 50 patients.

LASSO Theory and Implementation

LASSO (Least Absolute Shrinkage and Selection Operator) is the primary method we have explored for this project. LASSO aims to build a model using only the most informative variables [4]. It reduces many of the feature coefficients to 0, which ultimately removes redundant features [5]. In this way, we will be able to select a subset of features which pinpoint the most informative genes for predicting increased survival time across cancers.

Our methodology for implementing LASSO follows closely the algorithms described in *Stochastic Methods for l1-regularized Loss Minimization* [6]. For the ith sample, (i=1, …, n), let $x^{(i)} = [x^1, \ldots, x^m]$, where $x^{(i)} \varepsilon [-1,1]$, be the *m* x 1 gene expression profile vector and $y^{(i)} = [y_1, \ldots, y_m]$, where $y \varepsilon \{-1,+1\}$, be the *m* x 1 survival data vector. Let $\theta^T = [\theta_1, \ldots, \theta_n]^T$ be the *n* x 1 vector of quantitative weights assigned to each feature. Thus, our problem takes the following form:

$$\min_{\theta \varepsilon \boldsymbol{R}^d} \frac{1}{m} \sum_{i=1}^{m} [L(< \theta^T, x^{(i)} >), y^{(i)}] + \lambda ||\theta||_1$$

In our problem, $\lambda > 0$ is the regularization parameter and will be optimized using cross validation. $L: \boldsymbol{R}^d * Y \rightarrow [0, \infty)$ is a non-negative loss function. We will use stochastic coordinate descent which is considered to be a good method for large scale loss minimization [6].

The error loss function for our problem using LASSO is as follows:

$$e(\theta) = \sum_{i=1}^{m} \ln[1 + \exp(-y^{(i)}\theta^T x^{(i)})]$$

The logistic error function, which we will minimize over θ_j, is:

$$\left(\frac{\partial E}{\partial \theta_j}\right) = \sum_{i}^{m} -\frac{y^{(i)}}{e^{y^{(i)}\theta^T x} + 1} x_j^{(i)}$$

RESULTS

Currently, we have implemented a working prototype of LASSO in Matlab. Cross-validation results (n=5) are shown in Figure 2. Lambda values ranged from .0010 to .0450. The step size (α) used for coordinate descent is a scalar value set to 4.

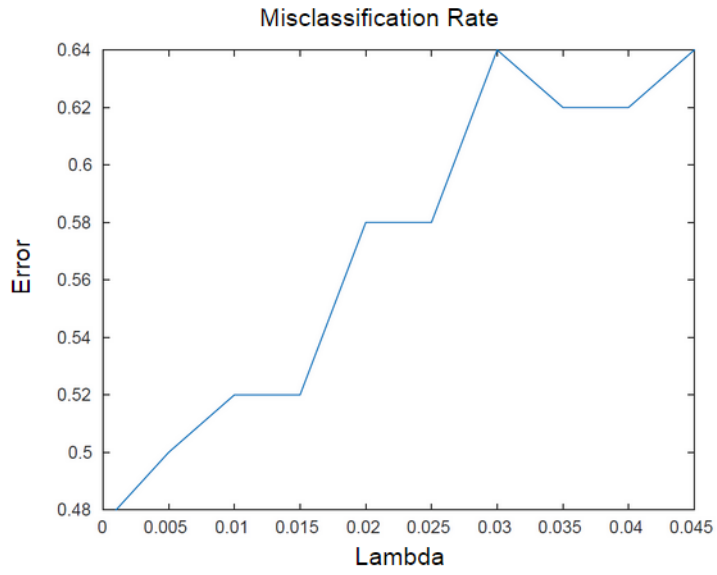

Figure 2: Cross validation results of the misclassification rate on the subset of data.

DISCUSSION AND FUTURE DIRECTIONS

We have been able to meet our milestone goal of having a preliminary working prototype of LASSO coded in Matlab. However, the current run-time is much slower than anticipated, even on the small subset of the data. Our current code uses coordinate descent. Implementing stochastic coordinate descent should significantly reduce run time. In addition, we will vectorize the algorithm where possible to remove excessive for loops.

Apparent from Figure 2, our current model for LASSO does not perform very well. The misclassification rate ranges from 0.48 to 0.64, which means the model is only classifying about half of the examples correctly; our algorithm is essentially non-informative. This will hopefully be remedied when we are able to run the entire dataset to find a better set of predictive genes and also analyze misclassification rate across a wider range of values for λ. Additionally, the cross-validation results should be run with n=10. Another part of our algorithm that could be improved is altering the step size (α) in the stochastic coordinate descent. This value is currently set to an arbitrary suggested value of 4 [6]. Cross validation could be run to determine its optimum value.

Because of the extremely poor performance of our current algorithm, our next steps will include a run of data through logistic regression to define a baseline for the potential performance of our prediction. Because this method will use all features to predict classification, we will examine the magnitude of the feature weights to determine if there are any patterns we will be able to discern through LASSO. If we are able to determine a subset of predictive genes, our final step will be to perform a literature search to validate the implications of our project by examining gene ontologies for biological significance [3]. We will also consider gene enrichment analysis. To do this, we will perform a statistical test which will determine if there is overrepresentation of closely related genes based on functionality (i.e. genes co-dependent in the same pathway, redundant genes).

REFERENCES

1. Cancer Genome Atlas Network (2008). Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*, **455**(7216), 1061-1068.
2. Cancer Genome Atlas Research Network, Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C. & Stuart, J.M. (2013), The Cancer Genome Atlas Pan-Cancer analysis project, *Nature genetics*, **45**(10), 1113-1120.
3. The Gene Ontology Consortium. (2005). Gene ontology: tool for the unification of biology. *Nature Genetics*,**25**(1):25-9.
4. Liang, Y., Liu, C., Luan, X.-Z., Leung, K.-S., Chan, T.-M., Xu, Z.-B., and Zhang, H. (2013). Sparse logistic regression with a L1/2 penalty for gene selection in cancer classification. *BMC bioinformatics*, **14**(1), 198.
5. Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc B.*, **58**(1), 267-288.
6. Shalev-Shwartz, S., Tewari A. (2011). Stochastic Methods for l1-regularized Loss Minimization. Journal of Machine Learning Research, **12**, 1865-1892.