|  |
| --- |
|  |
| **Predicting Gene Expression using Classification Models on TCGA Data** |
|  |

**Michael LeeJonathan Wolf**

Department of Computer Science Department of Computer Science

University of Washington University of Washington

Seattle, WA 98105 Seattle, WA 98105

*leem42@uw.edu* *wolfj4@cs.uw.edu*

**Abstract**

Gene expression refers to the process by which a functional gene is produced by the information from a gene. This process is ubiquitous in all organisms and understanding it better is all the more pertinent in a cancer data set. In this paper we predict gene expression for cancer afflicted patients using regression models.

**1 Introduction**

The Cancer Genome Atlas (TCGA) is a comprehensive study profiling thirty-three types of cancer. The goal of the study is to better understand the molecular basis for cancer. This study has generated over two petabytes of data publically available for researchers to analyze. This wealth of data has prompted many computational methods and made headway in finding features of cancer.

**1.1 Purpose**

The goal of this project is to predict gene expression for cancer afflicted patients. Making accurate predictions would provide potential for clinical application.

**2 Methods**

Our methods can be split into two areas of involvement. The first being data retrieval using SQL via Google BigQuery. The second area being our machine learning methods we chose to analyze our data.

**2.1 Data Retrieval**

We constructed our data tables by using SQL to combine elements of several tables on the Institute for Systems Biology (ISB) data set held in a Google BigQuerry table. First, we selected 50 random genes and 500 patients from four cancer types. Then, we joined the clinical data for each patient with the bio-specimen information of each patient’s tumor type. Biospecimen data included the following: average percentage of lymphocytes infiltrated, the percentage of normal/tumor cells of the tumor, and others. The clinical data we used was the patient BMI, age at pathological diagnosis, and the stage of the cancer type. Of the features listed the one of principal interest was the nearest mutation for a specific patient relative to the gene in question. The rational can be seen in figure (1).

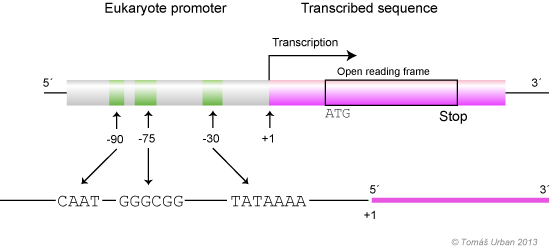


Figure 1: The promoter region is shown in full, examining proximal mutations is motivated by the many elements with which a mutation could disrupt transcription. **[1]**

We observe that mutations near by the promoter sequence of a gene, in particular the transcription start site and transcript stop site could drastically alter expression levels.

**2.2 Machine Learning Algorithms**

Using a regression was a natural choice at the beginning of our study. The values of the patients were continuous and this model makes predictions accordingly. We chose to examine lasso regression, ridge regression, and linear regression. These three models were tried with variations of parameters.

The second model we used in our study was a decision tree. The motivation for employing this model lay in its capability to make broader predictions than a regression model. Broader meaning that we sought to classify a patient as having high gene expression or low gene expression. We chose the barrier between high expression and low expression as the median value because it represented a split between half the data points. Our data had continuous features so our implementation used optimal thresholds of feature. The implementation we used was Iterative Dichotomiser 3 (ID3) with pseudocode provided at source [2].

The last model we considered was an AdaBoost framework. This model was chosen because it is another classifier that we could compare to the result of our decision tree model. Our framework used depth one decision trees for its base learners. We also normalized our weights for each data point after each iteration of the model. Further, to optimize the number of iterations trained over the data we used cross validation techniques. In particular, we implemented a 5-fold cross validation to pick the value of T such that it minimized error.

**3 Results**

**3.1 Regression Results**

Shown below is the results of our regression model and its sum squared error over the training data. This model predicted poorly as it had a squared error that we were able to outperform by having predicting values at random. We believe that the model performed poorly because the features we used lacked predictive power. The amount of predictive power of each feature will be explored later with principal component analysis.

**3.2 Decision Tree Results**

Our decision tree performed better than our regression.

**3.3 AdaBoost Results**

**3.4 Revisiting Decision Tree Using PCA**

**4 Conclusion**

**5 Discussion**

**Acknowledgments**

**References**

[1] http://web2.mendelu.cz/af\_291\_projekty2/vseo/print.php?page=307&typ=html