Predicting clinical features based on RNA-protein correlations in cancer patients

Hannah Boekweg, Corbin Day, Caleb Lindgren
Department of Biology
Brigham Young University

Abstract

Your abstract should concisely answer the following three questions: 1) what problem are you addressing? 2) what approach are you taking to solve the problem? 3) what are your results?

1 Introduction

There are several molecular process that occur in order for proteins, which perform the function of cells, to be made. It begins with DNA (the instuctions for the cell), which gets transcribed to RNA (the messenger), which is then able to be translated to proteins. Proteins are often refered to as the 'workhorses' of the cell, and are the molecules that actually carry out the instructions encoded in the DNA. The correlation between RNA and protein has been previously investigated [10.1016/j.tibs.2014.10.010] [10.1371/journal.pone.0043432] [10.1039/d1mo00178g]. We can use RNA/protein correlations to help us predict clinical outcomes of cancer patients. Cancer patients have an abnormal RNA/protein correlation when compared to healthy patients. We will use these abnormal correaltions to prediect patient survival, recurrence status, histology, pathology, and measure of patient recovery.

2 Methods

2.1 Data generation

Inputs

Our original proteomics and transcriptomic data was obtained from the Cancer Proteomics Tumor Analysis Consortium (CPTAC) (TODO: CITE) using the `cptac` software package (TODO: CITE). Previous work (not yet published) calculated the change between tumor and normal RNA-protein correlation, or delta correlation, for each protein across all patients in each cancer type. These correlations were calculated using the Spearman method. Orthogonal residuals for tumor and normal for each patient in each protein's correlation plot were calculated using the `scipy.odr` module.

We took the delta correlations and generated several features from them for training our models. This required some adaptation, as the delta correlations look at one protein across all patients, while we were interested in looking at one patient at a time. We thus sought to summarize the most important delta correlation information for each patient. First, we found the 25 proteins whose delta correlations had the greatest absolute value across all cancer types. Our reasoning was the the proteins with the greatest delta correlation were more likely to be informative. Then, for each patient, we calculated the change between tumor and normal residuals of the patient for each of those 25 proteins and saved these protein tumor-normal residual differences for each patient. We also noted whether that patient was above or below the tumor and normal regression lines for each protein.

After calcuating these tumor-normal residual differences for the overall most important proteins in each patient, and the above or below regression line data for each protein in each patient, we calculated correlation between tumor and normal residuals for each patient in across all proteins. This used the Spearman method again, and provided with a single new feature for each patient, based on the delta correlation data.

Our final feature was not derived from the delta correlations. Instead, for each protein in each patient, we calculated the ratio of tumor RNA abundance to normal RNA abundance and the ratio of tumor protein abundance to normal protein abundance, then found the correlation for each patient between this RNA tumor-normal ratio and this protein tumor-normal ratio across all proteins. The goal was to generate a feature for each patient that reflected differences between RNA and protein data, but, unlike the delta correlation-derived data, did not depend on data across multiple patients, and thus could be calculated for a single new patient without needing to find how that data fit in with data from other patients.

Targets

Clinical data was obtained in the same manner as proteomics and transcriptomics data. For targets, we selected six categorical features that had lower numbers of missing values and had potential to be predictable based on our input features. The targets chose were survival status, recurrence status, and treatment success status of the patient at last follow-up, as well as histologic grade and type, and tumor stage.

2.2 Data cleaning

For boolean input features (the above or below regression line features), missing values were filled randomly with true and

false values in such a way that the imputed data would have the same proportion of true to false values as the original data. Missing values in numerical features were filled with the mean of the feature, and all numerical features were normalized using the formula $(x - min(x))/(max(x) - min(x))$.

2.3 Scoring

We used three scoring metrics: accuracy, precision, and recall. Accuracy is simply the proportion of samples that were correctly scored. Precision and recall are based on the confusion matrix for the scoring results. For a single class, precision is calculated as $\frac{TP}{TP+FP}$, where TP is the true positive rate and FP is the false positive rate. It tells us what proportion of samples assigned the given class were actually in that class. Recall for a single class is calculated as $\frac{TP}{TP+FN}$, where TP is the true positive rate and FN is the false negative rate. This tells us what proportion of all members of the given class were successfully detected as members of that class. Thus, if we have high precision we know that anything assigned the given class is probably actually in that class; if we have high recall, we know that anything that is actually in the given class was probably assigned to that class.

In our data, some targets have more than two output classes. In these cases, precision and recall are calculated separately for each output class, and then averaged across all output classes.

3 Results

3.1 Baseline accuracy

To establish a baseline accuracy for the targets, we trained the SciKit Learn multi-layer perceptron (MLP) using the default hyperparameters, except reducing the hidden layers to one layer of 16 nodes, supposing that this simple model would give us a good idea of the complexity of the problem. The cancer type input feature, as well as all targets, were one-hot encoded. Testing with 10-fold cross-validation yielded these baseline results for each target:

Target	Accuracy	Precision	Recall
recurrence status	0.7702	0.7733	0.7764
survival status	0.8067	0.8113	0.8159
histologic grade	0.3309	0.3371	0.3434
histologic type	0.1949	0.1949	0.1949
success last follow-up	0.2570	0.2585	0.2600
tumor stage	0.3578	0.3761	0.3945

Table 1: Baseline scores for each target.

3.2 Multi-layer perceptron

After establishing the baseline accuracy using an MLP, we initially tried to optimize all hyperparameters at once using the SciKit Learn randomized cross-validation search. This method draws random combinations of hyperparameters from user-provided distributions and tests them, returning the best combinations it finds. However, we found that it took too many iterations to happen upon good parameter combinations if all parameters were being randomly selected at once. So, we instead decided to only use the random search to optimize one hyperparameter at a time, holding all others constant. Using this method, we obtained the following results for each target:

3.3 k-nearest neighbors

We used sklearn's KNeighbors model. 10 cross fold validation was used and accuracy, precision, and recall scores were reported. We optimized the parameters using sklearn's RandomizedSearchCV.

3.4 Naive Bayes

4 Discussion