Developing and Accelerating Predictive Models for Predicting Relapse of Pediatric Oncology patients using Smart Cyberinfrastructure

Mauricio Ferrato, Brad Atmiller, Erin Crowgey, Karl Franke, Sunita Chandrasekaran
Affiliation: University of Delaware, Nemours/Alfred I. duPont Hospital for Children

Goal: To apply ML/AI algorithms to clinical and genomic data of nearly 200 Leukemia patients to predict their relapse potential.

Dataset: The TARGET (Therapeutically Applicable Research To Generate Effective Treatments) from National Cancer Institute (NCI) initiative provides comprehensive molecular characterization to the greater research community. TARGET has applied this approach to multiple types of cancer so far. Our focus is on Acute Lymphoblastic Leukemia (ALL), a cancer of white blood cells. One half of our data is each patient's Clinical Annotations; each patient's demographic data (like age and race) and medical data (like White Blood Cell count and Minimal Residual Disease) throughout their treatment. The other half is their genetic profile. One major benefit of using TARGET genetic data is that it is already pre-processed and clustered into gene groups with levels of expression for each. Instead of low-level gene data and nucleotides, we get to work with a spreadsheet that has 20,000 gene groups for each of the 200 patients.

Project Focus: Our project focuses on using both clinical and genetic data of Acute Lymphoblastic Leukemia (ALL) patients to determine molecular changes that drive childhood cancers. We aim to find what mix of feature selection techniques and classification models provide the most accurate and generalizable results.

Approach: The first important step for any data science project is **pre-processing,** getting the data into a form that machine learning algorithms will recognize and can process. The primary data structure we use to hold our data is DataFrames from the Python Data Analysis Library (pandas). From there we clean up the data by removing columns that are missing too many values, fill in the rest of the absent value (using averages or placeholders), split categorical variables into a binary representation, and scale numeric columns to a consistent range.

The second step is **feature selection**. One of the major pieces of our research is feature selection; how to best determine which features will be most useful in predicting relapse. The most accurate learning algorithms are those which are fed high-quality data. One possible formalization is that a good feature set contains features which are highly correlated with the class, yet uncorrelated with each other. Specifically determining which genes out of the 20,000 will be most useful is quite difficult. There are a few different routes. One route is performing statistical analysis to find which variables are most closely correlated with relapse. Chi-squared statistic, correlation, information gain, Symmetrical uncertainty, ReliefF statistic, and a host of other forms of analysis are possible. One of the main analyses we've used is the ratio of expression for each gene between relapse and non-relapse patients. If we take the datascience route, there are several techniques we could employ to perform feature selection. Techniques could be Market Basket Analysis, PCA and autoencoder among others.

The third step is **classification.** Once we have handled missing data and selected features, we to determine the type of classier to use to arrive at predictive and accurate models for the data under study. We will explore artificial neural networks, and ensemble models based on random splits, bootstrapping and cross validation. We will gather accuracy, specificity and sensitivity metrics and analyze how each of the models arrive at these metrics. If the above ensemble algorithms are not useful for the dataset under study, we use another ensemble method called the gradient boosting algorithm for the given dataset. Gradient boosting is one of these ensemble techniques that uses weak learners, like small decision trees called decision stumps, to learn in an iterative process. While using these ensemble approaches, we want to be cautious and observant that the model does not pick a random sample every time for multiple iterations. We also need to explore the applicability of our findings on a larger dataset or dataset that is different to the one under study.

Research Challenges: As we can see, there are more than a handful of techniques one could use to perform feature selection or classification. Such an exploration is also very closely tied to the input dataset and the volume of the same, often termed as "Big Data" . In order to arrive at an acceptable % for accuracy, sensitivity and specificity (ROC curve for example) by the community, we are often bound to employing permutations and combinations of techniques. However, this is a very time-consuming approach. We need **modular workflows**.
Secondly, we clearly need **better software** to be able to use ML/AI solutions. We need interoperability between frameworks such that programmers are not re-inventing the wheel when they have to move between architectures. Given for example Habana relies on a base architecture for both training and inference but optimizes the designs for different workload as opposed to another route where a company manufactures fundamentally different architectures for training and inference. The challenge is how do you target these architectures for the case under study – building models for predictive oncology?
Thirdly, while disruptive hardware is fantastic for exploration, IMHO the community has not come to a consensus on the architectures we need for training and inference. Like mentioned above, sometimes the base architecture is the same and sometimes they are not. A better analysis or study of which is a better route would be very useful. To that end, we need ML/DL benchmark suite like SPEC CPU/HPG. MLPerf is something we have been referring to, however how comprehensive is the suite?

Workforce Development and Training: Jupyter notebooks are one way to go that can be adopted by the next-generation workforce in order to use the smart cyberinfrastructure without needing to get into the nitty-gritties of the workflow itself.

Tools and Techniques: We need "modular" workflows that can walk researchers through techniques and strategies employed for a given dataset. Scripts are OK but they are not helpful when the scientists have to test more than a handful of tools and techniques before arriving at the right combinations of tools for a given dataset.