PharmaSUG 2019 - Paper ST-325

Machine Learning Approaches to Identify Rare Diseases

Xuan(Kate) Sun and Ruohan Wang, Ultragenyx Pharmaceutical Inc. Novato, CA

Abstract

Rare diseases are very difficult to identify and diagnose than other diseases, since there are not enough data and experts in rare diseases. Better availability of patient data and improvement in machine learning algorithms empower us to tackle this problem computationally. In this paper, we adapt state of the art machine learning algorithms to make this classification, such as K-nearest neighbors, Support Vector Machine, Neural Networks and Naive Bayes. We find that using these machine learning methods, we can identify people with rare diseases with low misclassification rate.

I. Introduction

A. Challenge of Rare Diseases

The challenges of rare diseases are much greater than other diseases. The solutions for rare diseases are not distinguished as other diseases since there might be few experts in rare diseases. The limitation will cause diagnosis and treatment delay while it may not have corresponding medication for those rare diseases.

There are approximately 7,000 rare diseases and disorders. 30 million people, which is 10% of total population in the United Stated, are living with rare diseases. Such diseases usually have a genetic basis, 80% often affecting patients early in childhood, and are frequently progressive, disabling and life threatening in nature. These characteristics can have a devastating psychological impact on families of children suffering from these diseases.

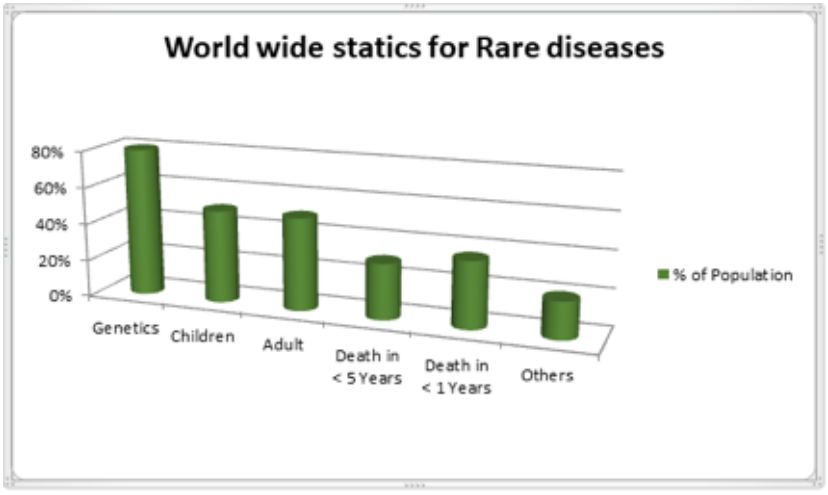

Figure 1: World wide statics for Rare diseases

B. Machine Learning

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to effectively perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as ”training data”, in order to make predictions or decisions without being explicitly programmed to perform the task.[1]

C. Machine Learning in Pharma and Medicine

Machine learning has been widely used in pharma and medicine and it could generate a value of up to $100B annually, based on better decision-making, optimized innovation, improved efficiency of research/clinical trials, and new tool creation for physicians, consumers, insurers, and regulators.

Disease identification and diagnosis of ailments is at the forefront of ML research in medicine. According to a 2015 report issued by Pharmaceutical Research and Manufacturers of America[2], more than 800 medicines and vaccines to treat cancer were in trial. In an interview with Bloomberg Technology, Knight Institute Researcher Jeff Tyner stated that while this is exciting, it also presents the challenge of finding ways to work with all the resulting data. That is where the idea of a biologist working with information scientists and computationalists is so important, said Tyner[3].

II. Machine Learning Models

The machine learning classification algorithms we have considered in our study are K Nearest Neighbors (KNN), Support Vector Machines (SVM), Neural Networks (NN) and Naive Bayes.

K-nearest neighbors (KNN) is an instance based learning algorithm that stores all the training instances and classifies the new cases based on the distance functions. K-NN works on the principle of classifying the new test case by a majority vote, with the test case being assigned to the class most common amongst its K nearest neighbors. Each point in the neighborhood may be assigned equal importance or given some sort of weight such as distance (larger the distance smaller the weight/importance).

To determine which of the K instances in the training dataset are most similar to a new input a distance measure is used. For real-valued input variables, the most popular distance measure is Euclidean distance.

Euclidean distance is calculated as the square root of the sum of the squared differences between a new point (x) and an existing point (x_i) across all input attributes j.

$$Euclidean\ Distance(x, x_i) = \sqrt{\sum (x_j - x_{ij})^2} \tag{1}$$

Other popular distance measures include Hamming Distance, Manhattan Distance, Minkowski Distance, Tanimoto, Jaccard, Mahalanobis and cosine distance. You can choose the best distance metric based on the properties of your data. If you are unsure, you can experiment with different distance metrics and different values of K together and see which mix results in the most accurate models.

Support Vector Machine (SVM) performs classification by finding the hyperplane that maximizes the margin between the two classes. The instances that define the hyperplane are called support vectors. The linear SVM classifier is shown in Figure. 2. H_1 does not separate the classes. H_2 does, but only with a small margin. H_3 separates them with the maximal margin. So H_3 is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum-margin classifier. SVM is high performing machine learning classifier due to the fact that if the data is linearly separable, SVM would produce a hyperplane that completely separates the vectors into two classes. However, in real life scenario perfect separation may not be possible. In such a case we can use different kernel function to project the data point into a different feature space where data may be linearly separable. Kernel function can be viewed as feature extraction techniques which take the existing features as input, performs the transformation and output a new feature space.

Neural Networks (NN) Neural networks are a set of algorithms, modeled loosely after the human brain, that are designed to recognize patterns. They interpret sensory data through a kind of machine perception, labeling or clustering raw input.

Neural networks help us cluster and classify. You can think of them as a clustering and classification layer on top of the data you store and manage. They help to group unlabeled data according to similarities among the example inputs, and they classify data when they have a labeled dataset to train on.

The learning problem in neural networks is formulated in terms of the minimization of a loss function, f. This function is in general, composed of an error and a regularization terms. The error term evaluates

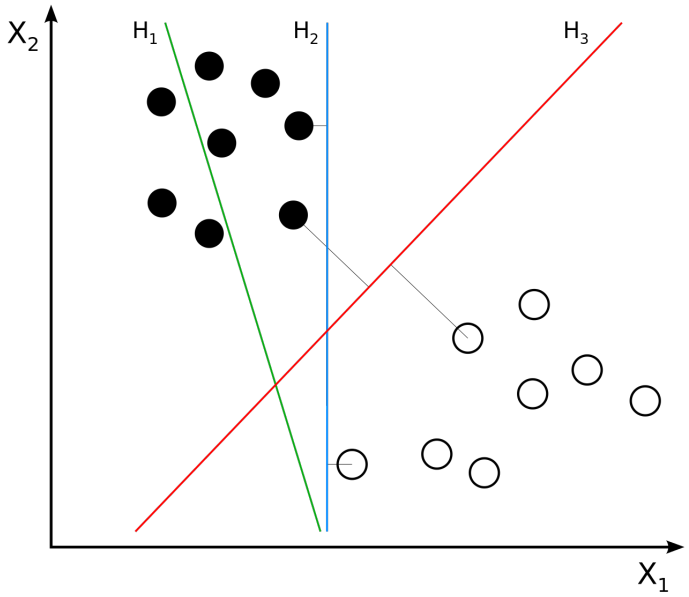

Figure 2: Linear SVM Classifier

how a neural network fits the data set. On the other hand, the regularization term is used to prevent overfitting, by controlling the effective complexity of the neural network.[4]

A typical Neural Net is shown in Figure. 3 with input layer, hidden layer and output layer. We can choose the number of hidden layers and the number of neurons in each hidden layer in the Neural Net.

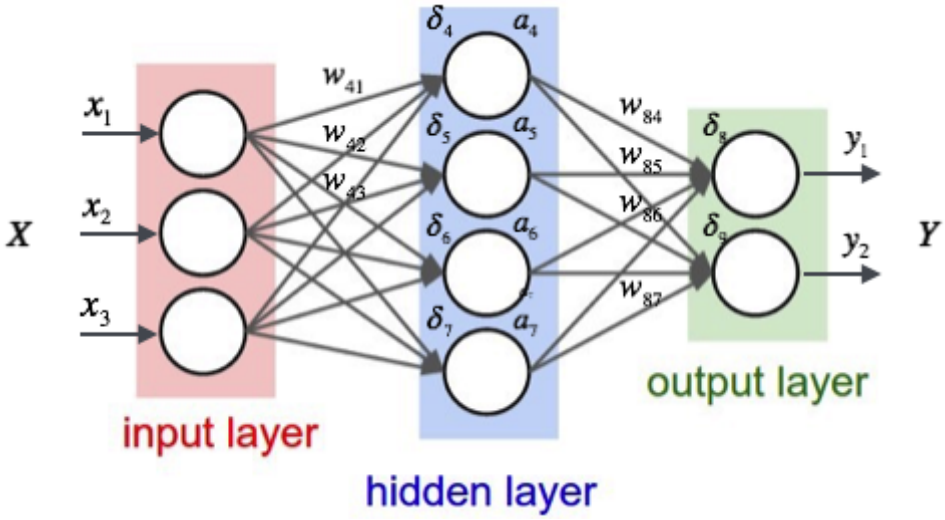

Figure 3: Neural Net Example

Naive Bayes (NB) algorithm is based on the Bayes theorem with assumption that the features are independent of each other. NB model is easy to develop with no parameter estimation. Despite its simplicity, NB model is widely used and outperforms other sophisticated machine learning algorithms such as SVM. NB provides a method of calculating posterior probability of class given the prior probability and learning the likelihood of features given the class.

III. Machine Learning Example

A. *Data prepocessing*

We applied different machine learning methods on an example data set about a rare disease. The data set includes the values of 400 instances and 25 attributes. Since the data set contains missing values, I

replaced the missing values with the mean of that variable data calculated without missing values. And we can get the new data set which is a full rank 400*25 matrix.

I divided the data set to two parts: training data set and testing data set where training set contains 200/400 data and testing set contains the rest data 200/400. Since this data set has plenty of missing values, I use mean value of numerous variables to represent the missing values. The original dimension of training and testing data is around half dimension of the raw data. After I replaced the missing numerous variable, the dimension changed to 178*25 and 171*25 which is much better than the original data set.

B. Cross Validation

Cross-validation is a statistical method used to estimate the skill of machine learning models.

It is commonly used in applied machine learning to compare and select a model for a given predictive modeling problem because it is easy to understand, easy to implement, and results in skill estimates that generally have a lower bias than other methods.

The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation. When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as $k = 10$ becoming 10-fold cross-validation.

The general procedure is as follows:

1) Shuffle the dataset randomly.
2) Split the dataset into k groups.
3) For each unique group:
    a) Take the group as a hold out or test data set.
    b) Take the remaining groups as a training data set.
    c) Fit a model on the training set and evaluate it on the test set.
    d) Retain the evaluation score.
4) Summarize the skill of the model using the average of model evaluation scores.
5) Pick the model M_i with the lowest estimated generalization error, and retrain that model on the entire training set.

C. KNN Method

If we do not define k value and set k a range, in this case I use $[2, 15]$, we can use cross-validation sampling method to get the best estimator of k which gives us the least error rate. When summarizing the result, we can get the best performance: = 0.02875817. The error rate of confusion matrix is around 0.03. For further calculation, we can get the misclassification rate of KNN method:

$$\text{Misclassification rate of KNN} = 0.03$$

D. SVM Method

Using SVM (Support vector machine) method to calculate the prediction values and error, when we use radial kernel in this case, we can get Error estimation of svm using 10-fold cross validation: 0.01111111. when we use polynomial kernel, the error rate increased to 0.01176471. So we use radial kernel as the comparison method.

$$\text{Misclassification rate of SVM} = 0.011111$$

E. Neural Networks Method

Using Neural Networks Method to calculate the prediction values and error, we use size = 8, Neural Network with $178 - 8 - 2$, in this case. (After omitting the missing nominal variables, the dimension of training data changed to $178 * 25$). The error rate of confusion matrix is around 0.0075.

$$\text{Misclassification rate of Neural Networks} = 0.0075$$

F. NaiveBayes Method

Using NaiveBayes Method to calculate the prediction values and error, we can get the confusion matrix:

Table I: Confusion matrix for NaiveBayes

y test vs. NaiveBayes pred	Predict=0	Predict=1
True=0	118	3
True=1	0	50

So, the error rate of confusion matrix is 0.0875. For further calculation, we can get the misclassification rate of NaiveBayes method:

$$\text{Misclassification rate of NaiveBayes} = 0.0875$$

The summary table is as below

Table II: Summary Table

Methods	KNN	SVM	NN	NaiveBayes
Misclassification rate	0.03	0.0111	0.0075	0.0875

IV. Conclusions

According to the summary table of those misclassification rates, we can see that the neural network has the least error rate. So, we can choose Neural Networks method to calculate the prediction model.

References

[1] Machine Learning Wikipedia, *https://en.wikipedia.org/wiki/Machine_learning*.

[2] Medicines In Development For Cancer. *http://phrma-docs.phrma.org/sites/default/files/pdf/oncology-report-2015.pdf*.

[3] Microsoft Develops AI to Help Cancer Doctors Find the Right Treatments. *https://www.bloomberg.com/news/articles/2016-09-20/microsoft-develops-ai-to-help-cancer-doctors-find-the-right-treatments*.

[4] Swingler K. Applying neural networks: a practical guide[M]. *Morgan Kaufmann*, 1996.

Disclaimer: This paper is using simulation data, it represents the options of the authors and not meant to represent the opinions from the company.

Contact Information

Your comments and questions are valued and encouraged. Contact the author:

Name: Xuan(Kate) Sun
Enterprise: Untragenyx Pharmaceutical Inc.
Address: 60 Leveroni Ct, Novato, CA 94949
Work Phone: 415-483-8974
Email: xsun@ultragenyx.com

Name: Ruohan Wang
Enterprise: Untragenyx Pharmaceutical Inc.
Address: 60 Leveroni Ct, Novato, CA 94949
Work Phone: 415-483-8810