Evaluating Chronic Cystic Fibrosis Severity Using Artificial Neural Networks

J. Redfield[1], D. Palmer[2], D. J. Conrad[3], and P. Salamon[1]
[1] Department of Mathematics and Statistics, [2] Department of Biology,
San Diego State University, San Diego CA
[3] UC San Diego Division of Pulmonary & Critical Care Medicine, La Jolla, CA

Abstract - *We present an ensemble of artificial neural networks to predict the severity of chronic cystic fibrosis within an individual by comparing against fifty patients ranked ordinally by increasing disease severity. The neural networks were programmed using Matlab and trained to minimize the sum-of-squared-error between the networks' rankings and the fifty actual rankings. The training data was subjectively but professionally ranked by one of us (DJC). Variables were chosen from the data collected in the UCSD cystic fibrosis clinic to use as input parameters for the networks. The resulting data matrix was then used by an ensemble of neural networks in training, validation and testing to ultimately produce a prediction of the chronic severity of cystic fibrosis within a patient. Between any two patients, the networks were able to correctly identify the more severe case 86% of the time. The goal was to capture Dr. Conrad's severity index, implied by the ranked set provided, to use on patients outside of that data set in assessing their chronic disease severity. The resulting neural networks have the potential to provide a useful diagnostic tool for physicians treating CF patients.*

Keywords: Machine learning, neural networks, cystic fibrosis

1 Introduction

We set out to develop an ensemble of artificial neural networks (ANNs) to predict chronic disease severity within cystic fibrosis patients as an experienced pulmonary physician would. ANNs are a form of machine learning algorithm based on the functionality and structure of a biological neural network, as observed in the brain. Used to observe complex trends and patterns in a set of data, they are capable of applying sets of non-linear equations to inputs to achieve a desired outcome. These equations can be reproduced to apply to further data, which gives rise to the predictive capabilities that this project utilizes.

2 Materials and Methods

The data were collected from patients cared for in the UCSD Adult Cystic Fibrosis Clinic (ACFC). Fifty patients were selected and ordinally ranked by Dr. Douglas J. Conrad, director at the ACFC, in order of increasing disease severity ranking 1 to 50 as a training cache for the ANNs. The 50 patients and their corresponding 14 variables were compiled as a matrix, along with their actual rankings, and imported to Matlab. Such variables included results from lung function tests (FEV1, FVC, FEV1/FVC), physical descriptions (age, height, weight, gender, BMI) and longitudinal regression values based on FEV1 vs. time graphs (m, b, r^2, se_m, se_b). For each patient, only the best FEV1 value from the previous year was considered.

2.1 Programming the ANN

To obtain the artificial neural networks, a progression of training, validating and testing steps were taken to develop the ability to predict with an acceptable amount of error. In training, each network is supplied with a set of data as inputs, and through a series of equations, returns an answer. Once the tested outputs of the network accurately reflect the answers provided in the training data, the ANNs can then be used to classify future data.

For an ANN to follow the trends in cystic fibrosis data, the variables were run through a series of equations, deemed "layers." Four matrices of random numbers were generated, two representing weights and two being biases. Let *w1* and *w2* denote the weight matrices, *b1* and *b2* the biases, and *in* represent the vector of one patient's inputs. The layout of a single-hidden layer ANN is as follows:

$$hidden\ layer = squash((in \times w1) - b1) \quad (1)$$

$$CF\ severity = (hidden\ layer) \times w2 - b2 \quad (2)$$

$$squash(x) = \frac{1}{1+e^{-x}} \quad (3)$$

Once the severity has been run through the above layers for each patient, the ANN returns its predicted severity, and the error is calculated between its prediction and the actual answer. The weights and biases are adjusted using Matlab's fminsearch optimization function, and after each adjustment, the squared error was calculated between the ANN outputs and the actual provided patient severity. Fminsearch is set to repeat these adjustments until a minimum in the total calculated error is found. However, if the entire set of 50 patients were to be used in training a network, there would be no unknowns upon which to test its accuracy. For this reason, only thirty of the fifty patients, or 3/5 of the original data set, were randomly selected and used to train each ANN.

The remaining 20 patients are randomized and split evenly into validation and testing groups. In validation, the purpose is to halt the fminsearch function once a network begins to over-train. Between the fminsearch iterations of the training set, the squared error is calculated and recorded for the ten validation patients. Once the weights and biases have become overly specific for the training set, the validation error will halt the network training (Figure 2). The test set for the network is recorded along with the adjusted parameters.

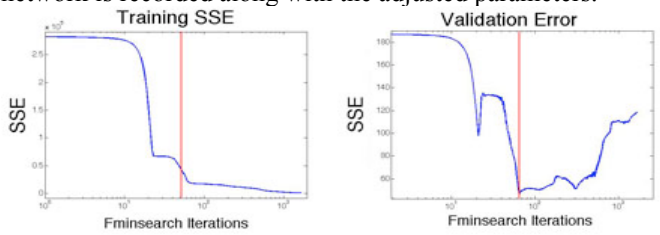

Figure 2. Training and Validation Errors for an ANN

2.2 Testing

A set of twenty networks' parameters and test sets was compiled. Each patient defined to be in a test set was run through the layers using the corresponding network's weights and biases, and averaged with the other participating networks. Thus, each ANN only "voted" on the severity of inputs that were not used in its training or validation processes. The averaged output consists of fifty patients, to be compared against the actual severities provided on the original fifty-patient datasheet.

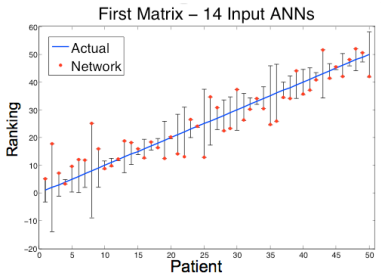

Figure 3. Voting of 14-input ANNs

2.3 Subsequent Inputs

Following the testing of the original networks, the importance of the 14 training variables was determined. Using the computational program R and the randomForest toolbox, FVC, FEV_1, and obstruction ratio were found to hold the strongest predictive power for CF severity. A new ensemble of 50 ANNs were trained and tested using only the FVC, FEV_1, and obstruction ratio as training features (See figure 5A).

A new dataset was then assembled including several new inputs. Multiproduct and powerproduct attempt to place emphasis on age, and were generated from the equations:

$$multiproduct = Age \times FEV1\% \quad (4)$$

$$powerproduct = FEV1\% \times e^{\frac{Age}{10}} \quad (5)$$

Other variables included the Brasfield score and its components. The randomForest toolbox predicted multiproduct, FEV1, FVC, powerproduct, obstruction ratio, and the overall Brasfield score as the most important variables of the new set (Table 1). An ensemble of ANNs were trained from these six variables and tested for their performance. For each of the 3 sets of inputs, the R^2 values were calculated (see Table 2). The ranking accuracy was defined by the ability of the ANNs to identify the more severe case of CF for any two patients.

First Inputs	Node Purity	Second Inputs	Node Purity
FVC	3018.8	Multiproduct	2308.5
FEV1	1960.8	FEV1	1217.0
Obstruction Ratio	1086.6	FVC	960.5
Age	616.3	PowerProduct	832.5
BMI	593.3	Obstruction Ratio	622.3
Weight	452.0	Brasfield	563.9
m	381.0	Cystic	481.3
r^2	377.9	Age	456.1
b	299.5	BMI	293.1
Patient ID	290.1	Overall	254.1
Height	258.6	Patient ID	160.1
sey	219.2	Height	145.4
seb	201.0	Linear	131.4
sem	155.7	Exp	73.1
Gender	21.5	Gender	16.5
		LgLS	13.2

Table 1. Sets of Inputs : The Initial and the Modified

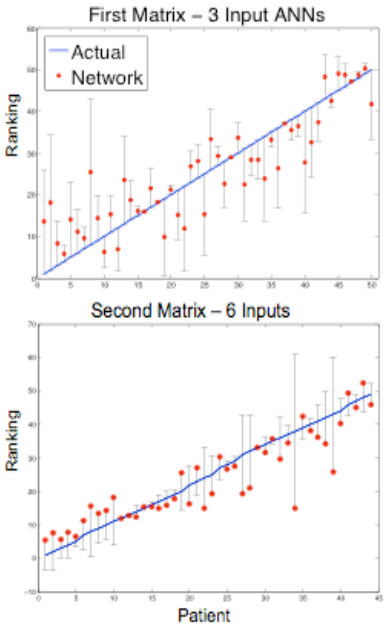

Figure 5. Voting of 3-Input and 6-Input ANNs

Matrix	Inputs	Train Time (min)	R^2	Ranking Accuracy (%)
1	14	360	.8261	86.67
1	3	10	.7845	85.14
2	6	30	.8109	88.48

Table 2. ANN Voting Results

3 Conclusions

The ensembles of neural networks were all able to train from the provided inputs and accurately vote upon unseen CF patients, as shown in Table 2. The variables FEV1, FVC, and obstruction ratio appeared to hold the greatest ability to train the ANNs from the original list of inputs. Of the second list of inputs, the Brasfield Index, multiproduct, and powerproduct were also found to be useful in ANN training. Future directions include expanding the parameters used, comparing the accuracies of multiple networks using the different inputs, obtaining rankings from other CF experts, and integrating the developed neural networks into a comprehensive GUI interface application being developed by the SDSU Cystic Fibrosis Group.

This work was made possible by NSF grant UBM 0827278 to A.M. Segall and P. Salamon. We thank the SDSU UBM and Cystic Fibrosis Groups for their guidance and many helpful discussions.