Phoenix Lens Technical Progress Report

This document contains the guidelines on the expected technical progress report of our project **Phoenix Lens**.

1. Introduction

Phoenix Lens is a **data driven predictive model** been developed to aid **USC Keck School of Medicine during surgical procedure**.

2. Problem definition

Predict the requirement of allogeneic blood during the process of surgery. Assisting the surgeon to limit the allogeneic blood requirement during the surgery. We will be delivering a web GUI (Graphical User Interface) where the system based on the following input parameters:

- Age
- Surgical Procedure
- Surgical Specialty
- SN_BM_Pre_OP_INR
- SN_BM_Pre_OP_Platelet_Count
- SN_BM_PRBC_Ordered
- Results Before Surgery (Hemoglobin)

The system will produce the following columns as the output:

- SN_BM_Red_Blood_Cells (Numerical Attribute)
- Allogeneic Blood Transfusion (Categorical Attribute)

3. The learning system

a. The target function

1) SN_BM_Red_Blood_Cells (In terms of inputs)

We are using Linear Regression to predict the column SN_BM_Red_Blood_Cells which is the actual usage of Red Blood Cells during the surgery in case the allogenic blood transfusion has occurred.

2) Allogenic Blood Transfusion (Required or not)

We are using Neural Network in order to classify the particular case for requirement of allogeneic blood transfusion.

b. Representation

Using linear regression model to get the relation between the following inputs and predicted column. During our development, we are designing a model that is basically using the column SURG_PROCEDURE and SURG_SPECIALITY to get a better predicative model. We are grouping the based on MULTIVALUE and obtaining the correlation inside each group among the inputs and outputs.

Using three-layer neural network based on the input parameters and sigmoid activation function we are developing an algorithm to classify the input surgical case into the requirement classes namely Yes or No.

c. System structure

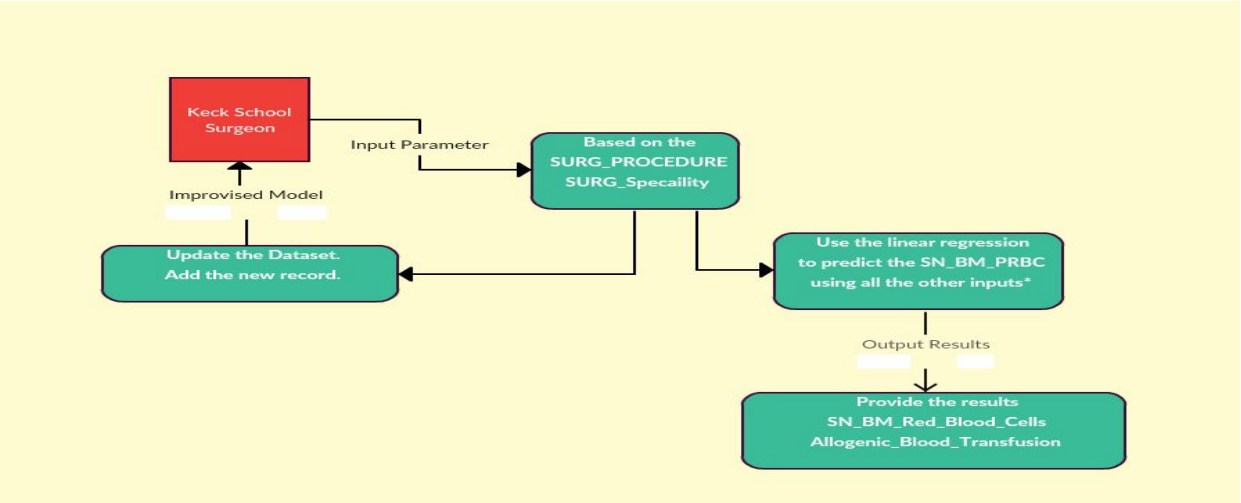

Figure 1

d. The learning algorithm

Following is the description of the algorithm we are developing for our system:

1) Load the dataset into pandas dataframe
   Dataset = pandas.read_csv('path_file')
2) Drop the irrelevant columns based on the data analysis.
3) Develop a linear regression model using the scikit library of python
   Target_variable = SN_BM_Red_Blood_Cell
   Dependent_variables = Age, Surgical Procedure, Surgical Specialty, SN_BM_Pre_OP_INR, SN_BM_Pre_OP_Platelet_Count, SN_BM_PRBC_Ordered, Results Before Surgery (Hemoglobin).
4) Develop a neural network model for classification into classes of Allogeneic blood transfusion
5) Test the developed model using the data set
6) Calculate the efficiency of the model

e. Improvements/modifications

As per our discussion with the TA we are working to use age as well to classify and get better results along with surgical procedure and surgical speciality.

4. Experimental Evaluation/tests /Results

a. The data sets

Using the business model description, we cleaned the dataset provided by the stakeholder. This cleaned dataset became our train data for the model.
Size of the data - 27217 rows where columns will be the input variables.
Train data Size: 60 percent of the total dataset.
Test data Size: 40 percent of the total dataset.

b. Learning

Linear Regression is very sensitive to Outliers. For example, as pointed during one of the class session SN_BM_PRBC_Ordered = 0 still there are instances where SN_BM_Red Blood Cells >0
which is clearly an outlier. Similarly, SN_BM_PRBC_Ordered > SN_BM_Red_Blood_Cells

but during data analysis we came across instances where it is not followed claiming an outlier.

Allogeneic Blood Transfusion
Classify the instance using neural network, which involves three layers and sigmoid function for activation.

c. Model Evaluation

1) Root Mean Squared Error (RMSE).
2) Cross Validation.
3) R-square.

5. Other learning systems

Different Models considered.

1. Polynomial Regression: A regression equation is a polynomial regression equation if the power of independent variable is more than 1.
2. Stepwise Regression : This form of regression is used when we deal with multiple independent variables.

6. Comparing learning systems

Both algorithms are really fast. There isn't much to distinguish them in terms of run-time.

- Logistic regression will work better if there's a single decision boundary, not necessarily parallel to the axis.
- Decision trees can be applied to situations where there's not just one underlying decision boundary, but many, and will work best if the class labels roughly lie in hyper-rectangular regions.
- Logistic regression is intrinsically simple, it has low variance and so is less prone to overfitting. Decision trees can be scaled up to be very complex, are are more liable to over-fit. Pruning is applied to avoid this.

7. Concluding Remarks & Future work

Future Works -> Includes ‘age’ in the model to improvise the correlation between the inputs and outputs.

8. Appendix

Reference

1)http://stats.stackexchange.com/questions/43538/difference-between-logistic-regression-and-neural-networks
2)https://www.r-bloggers.com/selecting-the-number-of-neurons-in-the-hidden-layer-of-a-neural-network/
3) https://www.analyticsvidhya.com
4)https://www.cs.upc.edu/~marias/teaching/ml/1regression.pdf
5)http://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html

Gantt Chart
https://drive.google.com/a/usc.edu/file/d/0B-rnx_dKiVzuZThsSy1IaUNiZDg/view?usp=sharing