Machine Learning Engineer Nanodegree

Capstone Proposal

Vitor R. Machado

August 29st, 2017

Domain Background

Cancer is a group of diseases involving abnormal cell growth, being one of the most complex group of diseases in the world. There are several universities and research centers searching for the best treatment for cancer. In Brazil, the state of Sao Paulo has a group of hospitals and treatment centers called ICESP known as the best option for cancer treatment in Brazil (read more in this link - in portuguese). These care centers provide indicators of the quality of their treatment based on their records (read more in this link, page 6), however, this network of hospitals does not have a method to predict the likelihood of mortality of future patients. Machine Learning was used once to solve a similar problem related to cancer. Machine Learning algorithms were developed to predict the mortality of patients which started chemotherapy (red more in this link) and to predict if patients with cancer would die or not (read more in this link).

Problem Statement

The ICESP is a network of hospitals of a third world country, therefore it does not posses the most advanced equipments for cancer treatment. Moreover, the most advanced treatments for cancer are not available in this country yet. Furthermore, the ICESP does not have a method to predict if a given patient will survive or not given a combination of treatments. This is a classic classification problem for Machine Learning. A Machine Learning model would receive all the data about the patient (for example, age, type of cancer, treatment to be performed and so on) and would output if this patient will survive or not the treatment.

Datasets and Inputs

The ICESP has several records of their patients. Their dataset is very detailed with a lot of information and can be downloaded through this link. Their dataset has 735237 records of patients from 2000 to 2017. Since this dataset has several columns, each column in this dataset is described in an external PDF file (columns_definition.pdf). The original description of each column is available in this link, in portuguese. This dataset contains data such as the age and sex of patients, the treatment they received (Immunotherapy, hormone therapy, chemotherapy, radiotherapy, surgery), the type and stage of the cancer, and if the patient was cured, died or left the hospital before the end of the treatment. This dataset is extremely relevant for the project since it is possible to make classifications using Machine Learning through this dataset. Learning more about this dataset would allow Machine Learning algorithms to make predictions about new patients and their chances of surviving. There are 4 output classes in this dataset (1 - the patient left the hospital alive with cancer, 2 - the patient left the hospital alive without cancer, 3 - the patient died from cancer, 4 - the patient died from other cause). The records are distributed in these 4 classes in this proportion:

1	95955	13.05%
2	339986	46.24%
3	214900	29.23%
4	84396	11.48%

The idea of the proposed model is to predict whether a patient will survive or not at the end of the treatment. For this, the classes 1 and 2 will be unified to form the surviving group as well as the classes 3 and 4. So the data will be splitted in 59.29% (survivors) and 40.71%. This numbers shows that the two output classes (survive, die) are balanced. To train and test the algorithm, 80% and 20% of the data will be used respectively. The data will be shuffled before being splitted, to guarantee that neither training or testing data will be unbalanced (since that it is possible that the classes 3 and 4, for instance, are in the final of the data).

Solution Statement

My main goal is to build a Machine Learning model to help the ICESP to make predictions of the likelihood of life or death for every future patient of this network of hospitals. This model would help ICESP to increase the efficiency and rate of success of their treatments. The features of the data will receive the necessary normalization and transformation (machine learning techniques to standardize the data) before being sent to train and test the model. Also, the redundant or unnecessary features will be removed.

Benchmark Model

For making predictions of the likelihood of life and death, a Random Forest Classifier is applicable. I've chosen Random Forest because of its versatility and performance. Random Forest performs well in data with many features, and are considered a fast algorithm in training, testing and predicting. The benchmark consists of comparing the predicted classes of my random forest model in the testing set with the true labels of this test set, to check the accuracy of the model and if it is able to make future predictions in the true world. To evaluate if the chosen model is the best choice for the test, another pre-defined model will be built, using K Nearest Neighbors. The accuracy of both models will be compared.

Evaluation Metrics

Since the output classes are balanced, the accuracy will be used as evaluation metric for the model. Accuracy is the number of correct predictions made divided by the total number of predictions made, multiplied by 100 to turn it into a percentage. Accuracy measures the overall performance of the model in predicting new values. To make the model more reliable, another metric will be used since accuracy alone is a weak evaluation metric (read more about the accuracy paradox). The F-1 score Along with the accuracy, the metric F-beta score will be used. The F-beta score is the weighted harmonic mean of precision and recall.

Project Design

Firstly, prior to using the records of the dataset to train and test the model, some preprocessing is required:

- Categorical features will be transformed into dummy features;
- Numerical features such as age, days between first appointment and last information about the patient will be normalized to range between 0 and 1 to facilitate both training and testing;
- Features that are supposed to be irrelevant or redundant (such as the state of residence of the patient, age range and patient's education degree) will be removed.
- Missing values in some columns will be replaced with 0's if the values of the column are categorical, and in cases where the values in the column are continuous, the mean will replace the missing values;
- The column ULTINFO which has the values 1, 2, 3 and 4, will be used to produce the output labels 0 and 1 (died and lived, respectively). The values 3 and 4 will be transformed into 0 and 1 and 2 into 1.
- Finally, the processed data will be split into 80% and 20%, for training and testing, respectively.

The random forest model will be build along with GridSearchCV in order to find the best estimators for this model. The parameters n_estimators (the number of trees in the forest), max_depth (the maximum depth of the tree) and min_samples_split (The minimum number of samples required to split an internal node) will be used in the GridSearchCV to find the best model for the random forest. The optimized model returned by the GridSearchCV will be trained using the training set and tested using the testing set. The accuracy and the f-beta score for this model will be displayed. Finally, a default KNN model will be trained and tested with the same data, and the accuracy and f-beta score for this model will be displayed as well for comparison and benchmarking.

The following Python libraries will be used for this task:

- Numpy
- Pandas
- SKlearn
-