Logistic Regression

In this assignment, Breast Cancer Wisconsin Data Set (named “breast-cancer-wisconsin.data”) was used to train a Logistic regression model that was implemented using python’s scikit learn LogisticRegression library to predict the severity of a breast cancer. Ten features describing the characteristics of a cell nuclei present in an image taken from a fine needle aspirate (FNA) of a breast mass was used to predict whether the tumor was benign or malignant. Name of the ten features are Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal, Mitoses.

First the data set was loaded into a Panda’s data frame and given appropriate column names for the features since original data was not labeled. This was just to introduce more readability to the loaded data. Original data set was observed to have 16 missing values- represented by ‘?’ out of the total of 698 records. In the first phase of the algorithm, those records with missing values were dropped from the data frame by replacing ‘?’ values with np.nan value and then by dropping nan values using Panda’s dropna method on row-basis of the data frame. Although it is not directly related for the classification, seaborn’s countplot method was used to visualize the distribution of the count of feature values that were associated with resulting target classes. Class 2 was representing benign and class 4 was representing malignant cancers. It can be seen that some values of the selected feature was totally falling on a specific class (Ex:- clump thickness value 10 totally falling on class 4), while some feature values falling on both the classes (Ex:- clump thickness value 5).

In order to train the model, data set was split using test to training ratio similar to the research paper as directed. For this purpose, sklearn.linear_model’s train_test_split method was used. Sklearn’s LogisticRegression module was used as the model with ‘lbfgs’ as the optimization algorithm that uses L2 or no penalty in solving. Split data set contained two data sets as training and tests each having a feature set with associated class set. Then the logistic regression model was trained using the training data set of the features and the corresponding classes. This model is then used to predict the class labels of the above test data set. After observing the predictions, it can be seen there was no predictions associated with class-4 in spite there were 88 instances of class-4 in test data set. classification_report method was used to generate a summary of precision, recall, f1 score and support for each of the class labels and can be observed that class 2 was having a 63% precision and 100% recall (all the class 2 instance were predicted) with 151 support while class 4 was having 0 precision and 0 recall with 88 support. confusion_matrix method was used to generate confusion matrix for above predicted and actual results and claim to be have 151 true negatives, 0 false positives, 88 false negatives, and 0 true positives. Overall accuracy of the model was 63.1%.

On next phase above same process was followed again by replacing the missing values on the original data set with ‘1’ and it was observed that new predictions of the model contained classes of both 2 and 4. Classification report also resulted better results with 98% precision, 92% recall, 95% f1 score and 164 support on class 2, and 86% precision, 96% recall, 91% f1 score and 80 support on class 4. New confusion matrix depicted a lesser number of false cases such that 151 true negatives, 77 true positives, 13 false positives and 3 false negatives. Overall accuracy of the model had been significantly increased to 93%. This experiment clearly showed the importance of not ignoring data samples with missing values and rather using those in model training with a better imputation of data. It helps to keep the sample data size intact and therefore the model gets more chances to train.