Documentation

November 30, 2019

Data Pre-processing

The aim of this report is to analyze the Kaggle stroke data-set, to obtain a predictive model and to gain some insight in the most responsible causes that produce stroke. In general there are 11 features that are present in the data set that involve both categorical and continuous data. The figure below shows the correlation that are seen at first sight of the data.

	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
id	1.000000	0.005445	0.000131	0.002157	0.013075	0.000043	0.008955	-0.000966	0.020976	0.013226	0.004082	0.002238
gender	0.005445	1.000000	0.040785	0.037431	0.097549	0.024852	0.010247	0.004814	0.053496	0.019278	-0.013613	0.013689
age	0.000131	0.040785	1.000000	0.259528	0.251819	0.546996	0.018850	0.004044	0.230682	0.101619	-0.147968	0.159838
hypertension	0.002157	0.037431	0.259528	1.000000	0.114957	0.133258	0.019309	-0.004427	0.154702	0.120730	-0.030094	0.073310
heart_disease	0.013075	0.097549	0.251819	0.114957	1.000000	0.098229	0.034466	-0.000583	0.139449	0.018562	-0.035660	0.107007
ever_married	0.000043	0.024852	0.546996	0.133258	0.098229	1.000000	-0.067305	0.004990	0.120161	0.139949	-0.053049	0.051666
work_type	0.008955	0.010247	0.018850	0.019309	0.034466	-0.067305	1.000000	-0.010796	0.008315	-0.066278	-0.030810	0.025708
Residence_type	-0.000966	0.004814	0.004044	-0.004427	-0.000583	0.004990	-0.010796	1.000000	-0.001361	-0.002944	0.010095	0.002064
avg_glucose_level	0.020976	0.053496	0.230682	0.154702	0.139449	0.120161	0.008315	-0.001361	1.000000	0.167699	-0.035164	0.077206
bmi	0.013226	0.019278	0.101619	0.120730	0.018562	0.139949	-0.066278	-0.002944	0.167699	1.000000	-0.035568	-0.006950
smoking_status	0.004082	-0.013613	-0.147968	-0.030094	-0.035660	-0.053049	-0.030810	0.010095	-0.035164	-0.035568	1.000000	-0.023068
stroke	0.002238	0.013689	0.159838	0.073310	0.107007	0.051666	0.025708	0.002064	0.077206	-0.006950	-0.023068	1.000000

Figure 1: Correlation table

The first issue to notice is that the data set contains a lot of missing values. As shown below.

```
[ ]  df.isnull().sum()/len(df)*100 #percentage of missing data for each feature

id                   0.000000
gender               0.000000
age                  0.000000
hypertension         0.000000
heart_disease        0.000000
ever_married         0.000000
work_type            0.000000
Residence_type       0.000000
avg_glucose_level    0.000000
bmi                  3.368664
smoking_status      30.626728
stroke               0.000000
dtype: float64
```

Figure 2: Missing data percentage

The issue was dealt with by imputing the data. More specifically, the bmi variable only had a small portion of its data missing and hence it was reasonable to approximate with just the mean value. In addition the smoking status was also problematic but almost 30% of its

data missing, such an issue would require the use of an unsupervised learning algorithm in order to fill in the values (Random forest, with an EM algorithm optimization).However in the interest of time the data were decided to be ignored instead.

Another issue to notice is the imbalance of data that exists in regards to having stroke (783 data points) and not having a stroke (42617). In order to deal with this the SMOTE library was used in order to balance the data set, which effectively produces the additional data set by considering the convex combination of the actual points.

Prediction and feature selection

Two different models where investigated. The first one being logistic regression along with L1 norm regularization (in order to avoid over-fitting and to analyze the surviving coefficients.) The second one was a random forest algorithm. In general for both algorithm a k cross validation algorithm was employed on a validation set separately from a test set in order to find the best performing hyper-parameter over a grid of values. To elaborate, the logistic regression method is faster but less accurate and with the help of the L1 norm along with the value of coefficients it can be investigated which features go to zero. For the case of the random forest the hyper-parameter in question was the number of trees to be used for the rest of the variables the default values where used. Notice that the correlations reflect the values of the l1 norm coefficients intuitively since

Algorithm	Validation accuracy	Test accuracy	Hyper-parameter value
Logistic Regression	75.7%	75.7%	10
Random Forest	99.9%	87.7%	200

Table 1: Results of the algorithms

Feature importance is shown below:

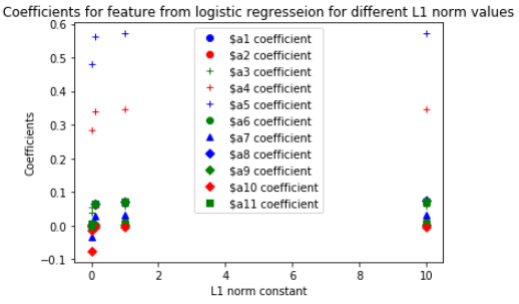

Figure 3: Feature importance for regularization

According to Random Forest the feature importance correspondingly as follows: a1=0.14917898, a2=0.01799764, a3=0.34528338, a4=0.02407382, a5=0.03259769, a6=0.02040119, a7=0.03189783, a8=0.01965301, a9=0.17401725, a10=0.15204961,a11=0.03284961. Therefore Random Forest seem to capture higher order correlations as well