Project

The task of binary classification using so-called lazy method was applied to dataset with early stage diabetes[1] cases. All attributes except age were binary. I performed binarization of this attribute using thresholds found at distribution of this attribute and one-hot-encoder.

I decided not to use notations as classified as 'unknown' or 'contradictory' because it's not what we want to get in real life.

As suggested, I used three features during the decision-making process – classification:

1. $Aggr_i|g' \cap g_i^+| = \frac{\sum_i|g' \cap g_i^+|}{|G^+|}$ – power of intersection
2. $Aggr_i|(g' \cap g_i^+)^+| = \frac{\sum_i|(g' \cap g_i^+)^+|}{|G^+|}$ – support of intersection
3. $Aggr_i|(g' \cap g_i^+)^-| = \frac{\sum_i|(g' \cap g_i^+)^-|}{|G^-|}$ – confidence of intersection

The idea of the first algorithm (power) is the following: to classify object from test group I calculate the intersection with all objects from train positive class and train negative class and normalize this values on the size of corresponding class. The final decision of label class (1) assignment is made in accordance with the following rule $power_{positive} - power_{negative} > threshold.$ For parameters tuning I used KFold cross-validation. The highest accuracy score is achieved with threshold=-0.05.

The idea of the second algorithm (support) is the following: to classify object from test group I intersect tested object with each object from positive (negative) class and for this intersection I find how many objects from this positive (negative) class also contain this intersection. The final decision of label class (1) assignment is made on comparison of rate of support of tested object in each class in accordance with the following rule $support_{positive} - support_{negative} > threshold.$ For parameters tuning I used KFold cross-validation. The highest accuracy score is achieved with threshold=-0.1.

The idea of the third algorithm (confidence) is the following: to classify object from test group I intersect tested object with each object from positive (negative) class and for this intersection I find how many objects from the contrary class also contain this intersection. The smaller this rate the better for predicting class as positive, so that The final decision of label class (1) assignment is made according to the following rule $confidence_{negative} - confidence_{positive} > threshold.$. For parameters tuning I used KFold cross-validation. The highest accuracy score is achieved with threshold= -0.015.

So to assign label of positive class all classification rules let the calculated rate for positive class objects be slightly lower than the calculated rate for negative class objects. This means that intersection with objects from positive class is more important than with objects from negative class. This statement looks quite realistic in case of predicting illnesses.

The results as basic metrics for all classification rules are presented in the table below:

[1] https://archive.ics.uci.edu/ml/datasets/Early+stage+diabetes+risk+prediction+dataset.

accuracy	precision	recall
power (threshold=-0.05)		
0.846	0.947	0.792
support (threshold=-0.1)		
0.861	0.873	0.905
confidence (threshold=-0.015)		
0.938	0.996	0.902

The last algorithm (confidence with threshold=-0.015) outperform not only in accuracy, but also in False negative rate (confusion matrix is calculated in .ipynb file), which is the lowest among all algorithms ($FNR = \frac{FN}{FN+TP} = \frac{3}{3+68} = 0.05$). This is crucial in medicine – not diagnosing a disease when you actually have it, is a serious mistake.

The last algorithm perform slightly worse than SVM method with accuracy=0.96, precision = 0.97, recall = 0.96, fnr = 0.03.