ARTIFICIAL INTELLIGENCE TERM PROJECT REPORT II

Mehmet Onur Erboy – 15011007

1. Project Description

For this project, some classification algorithms are tried to implement for a prepared Kaggle dataset. This dataset contains credit card information about fraud and normal processes. Scope of the project, 3 algorithms are studied for this classification process : K Nearest Neighbour, Naive Bayes and Learning Vector Quantization.

The project has been developed with python programming language because of the developer experience background.

As a result of this project, these algorithms are compared and investigated for this dataset.

The details are described at further chapters.

2. The Solution and Development Process

Dataset Link : https://www.kaggle.com/mlg-ulb/creditcardfraud

Upon develeping this Project, firstly the dataset is observed. So, I am beginning with dataset's details. Dataset contains 31 attributes. The first attribute represent the indis for time. Next 28 column is related to credit card process. But this fields are transformed to different values with PCA algorithm because of the security of banking data. First column after these columns represent the amount of credit card process. The last attribute keeps the fraud information. This field contains "1.0" value for fraud and "0.0" for normal processes. In this dataset, there are 284.807 records and 492 of them are labeled as fraud.

Before the classification process, we have to seperate data as train data and test data. For this purpose I use Holdout method. With this method, I seperate 70 percentage of data as train data and all other data as test data. Implementing this seperation, there is a problem that there are very limited fraud data according to normal data. So, ordinary split operation cause that all train or test data contains only normal data. This is not healthy for classification. So, I keep normal data and fraud data on different variables. Then, I calculate the train dataset record number and test dataset number. These values also are calculated for normal data and fraud data. I generate a random value between 0 and normal data. Then, this obtained value of normal array is added to test data. Also, this indis is kept and it is not used next selected test normal data. This process is implemented on fraud data. Remaning normal and fraud data is adding to train data.

After this dataset prepare operation, algorithms are tried to implement. Naive Bayes algorithm is implemented firstly. In this algorithm, normally conditional probabilty is calculated for each columns

while we are observing the label column. But, our data is not categorical. Our data is numerical. So, the probability calculation could not direktly because of the continious range. So, the numerical field oriented naive bayes calculation is implemented for each row probability. They are calculated for each row and after they multiplied. This calculation must be impelemented for each label. So, the calculated dataset splitted according to label and the mean and standart deviation is calculated. After that, the probability values that obtained with the below formula are multiplied. Finally this value is multiplied with the probability of calculated label occurance. Used formula is here :

$$f(100000) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Normal Distribution 1000 value

Formula 1. Naive Bayes row based probability calculation for numerical attributes

The K Nearest Neighbour algorithm is a distance calculation based algortihm. Before the deciding the label of a data point, we have to obtain all distance value to train data field. After this process, we count the labels of k nearest neighbour data points that selected with minimum distance value. The most repetitive label of these neighbours is the label of this data point. In this dataset there are nearly 200 k train data point and nearly 85 k test data point. This means that 200.000 x 85.000 distance calculation, 85.000 times sorting 200.000 length distance array and 85.000 times obtaining the most k shortest distance of that length distance. All of these means too much calculation. And I could not shorter that. I could not give a result for this algorithm.

Learning Vector Quantization algorithm is very similar to K Nearest Neighbour algorithm. But, it takes some random values from train data and calculate. This method generate some vectors for calculation. According to this vectors, it tries to find nearest vector to test data point. After the implemented calculation with this data point and vector, calculated label is equal to this, test data point is rewarded with closing this vector. Otherwise it is punished as moving away from there. This reward and punishment rate is related to learning rate. Learning rate is high in first steps but it decreases on latest steps.

3. Numerical Success

I could only obtain for Naive Bayes results. They are represented on Table 1.

Real \ Predicted	Normal Data	Fraud Data
Normal Data	0,663413	0,334854
Fraud Data	0,000842	0,000889

Table 1. Naive Bayes Classification for Project Dataset

According to this table, the success of naive bayes could be said 67 percentage. Most of normal data is labelled as fraud data. This could be problematic in a banking sector because of false alarm.