International Journal of Pure and Applied Mathematics
Volume 119 No. 12 2018, 15977-15988
ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue

ijpam.eu

A study of classification algorithms using Rapidminer

Dr.J.Arunadevi[1], S.Ramya[2], M.Ramesh Raja[3]

[1] Assistant Professor, PG Department of Computer Science and Research Center, R.D.Govt. Arts college, Sivaganga, Tamilnadu

[2] M.Phil Research Scholar, PG Department of Computer Science and Research Center, R.D.Govt. Arts college, Sivaganga, Tamilnadu

[3] Ph.D Research Scholar, PG Department of Computer Science and Research Center, R.D.Govt. Arts college, Sivaganga, Tamilnadu

Abstract:

Classification is an important task in the day to day life. In this paper we have analyzed the performance of various classifiers like K-nearest neighbor, Naïve bayes, generalized liner model, Gradient boosted trees, deep learning with H2O. The classifiers are checked against four synthesized datasets. This experiment is carried out in the Rapidminer tool. In the observation of the results Deep learning with H2O outperforms the other classifiers in most of the case. The results are clearly discussed.

Keywords: K-nearest neighbor, Naïve bayes, Generalized liner model, Gradient boosted trees, Deep learning, Rapidminer

1. Introduction

Classification in the data mining functionalities is an inevitable job by which the data could categorize by the class labels previously known. Due to this reason this is said to be supervised learning. The jobs related to data classification are enormous in the day to day life. Classification is the task in the data mining knowledge discovery process by which the data under consideration could be grouped under the known class labels. This task could be

performed in two phases. The phases are training and testing. In the training phase the job to be done is to train the classifier with the training sample set of the data. In the testing phase the untrained sample set is exploited against the knowledge gathered by the training set.

There are number of methods employed for the classification task in the dataset. This paper is written to analyze the various methods used for the classification purpose and then the strength and weakness of the same. The research in the classification is still an open avenue to find the better classification algorithm. Here this paper interrogates about the various methodologies adopted for classification and their uniqueness.

Section two talks about the literature review on the classification task and the various methods employed for it. Section three says about the algorithms used for the comparison of the classification tasks. Section four leads to know about the experimental environment. Section five details the results obtained and give the discussion on it.

2. Literature review

Classification is the task used to predict the class label of the dataset which is discrete or nominal. This section reviews the various algorithms available for the classification task in the literature.

2.1 Bayesian classification

Bayesian classification is based on the bayes' theorem. Bayes' theorem gets the probability of the event based on the knowledge already known about the conditions related to the event [1]. Bayes' theorem is stated mathematically as

$$P(H/X)= P(X/H)P(H) / P(X) \text{ ---- } 1$$

Where H and X are the events and $P(X) \neq 0$

P(H/X) – the likelihood of event H occurring given that X is true

P(X/H) - the likelihood of event X occurring given that H is true

P(H) and P(X) are probabilities of observing H and X independently of each other.

2.2 Nearest Neighborhood classification

It is a non-parametric method used for classification and regression. In this class of classification technique the new data sample is classified by calculating the distance to the nearest training case [2]. This type of classifier use all the patterns in the training set to classify a test pattern [3]. This model is less expensive in training but more expensive in the testing phase.

2.3 Linear models

This type of classifiers is used to do the classification based on the linear combination of the features of the dataset. This linear model classifier is mainly used for dataset with many features [4]. In non linear classifiers the data is mapped to a high dimensional space but in linear models directly work in the data.

2.4 Boosting

Boosting is the technique used for the improvement of the performance of the models. It consists of a family of algorithms. The main aim is to convert a weak learner to strong learners. There are many types of boosting available they are adaptive boosting (AdaBoost), Gradient tree boosting , XGBoost etc. It is a flexible non linear regression procedure.

2.5 Deep learning

Deep learning is the sub class of machine learning which mimics the brain. It could also otherwise call as large neural network. But the concept is not knotted with the neural networks alone. Since the usage of deep learning is well explained thorough neural networks it is mostly understood that it is the large neural network.

3. Algorithms used for comparison

There are five algorithms used for the comparison purpose in this paper. They are as follows

3.1 Naïve bayes classifier (NB)

This algorithm is a special case based on the bayes' classification. This algorithm is written based on the probability models which integrate strong independent assumptions. This model is easy for the construction of the classifier and it is suitable for large database too since the basic idea is the independent assumption of the features.

3.1.1 Pseudo code

BEGIN

Convert the given dataset into frequency table

Create likelihood table by finding the probability

Find the posterior probability for each class

Class with maximum probability is the class predicted

END

3.1.2 Advantages

- Relatively easy to build

- Performs well in multiclass classification

3.1.3 Disadvantages

- In case of zero frequency we need a smoothing technique
- In case of independent features it is a rare case in real time dataset

3.2 K Nearest Neighbor (KNN)

KNN is also an efficient and simple classifier to build. It is non parametric and instance based classifier. Non parametric means it doesn't take any pre assumption and thus it doesn't lead to the danger of in correct modeling of the data distribution. This classifier doesn't build any model but learn from the memory of the training phase.

3.2.1 Pseudo code

BEGIN

Calculate the distance between the points in the dataset

Arrange the distances in the ascending order

Take the first K distances from the arranged list of distances

Find the K points with the K distances

Return the majority to the class

Repeat until all the points in the dataset are classified

END

3.2.2 Advantages

- Performs well for large dataset
- Robust to noisy data

3.2.3 Disadvantages

- There is a need for determination of optimal value for K
- Computation cost is high

3.3 Generalized liner model (GLM)

GLM is the special case of linear models. GLM extends the linear models to a response variable by a link function and it also takes care of the weight of the variance of each

measurement to be the relative function of its predicted value[5]. It consists of three components namely probability distribution, linear predictor and link function.

3.3.1 Pseudo code

Let Y_i be the independent random response variable with n observations

Let $g(\mu_i)=\eta_i$ be the differentiable transformation of the expected value of y_i

Let $\eta_i = \mathbf{x}'_1\beta$ be the link function

Let z_i be the working dependent variable and w_i iterative weights

BEGIN

For an initial rough estimate of the parameters β, use the estimate to generate fitted values $\mu_i = g^{-1}(\eta_i)$

$z_i=\eta_i +(yi-\mu_i)d\eta_i / d\mu_i,$

Calculate W_i

Calculate X_i *based on the calculated* W_i *to get the new estimate of* β

Repeat until the β changes less than the predetermined value

END

3.3.2 Advantages

- Flexibility in modeling
- The model is fitted for the optimal properties of the estimator

3.3.3 Disadvantages

- Linear function will give linear estimator only
- Responses must be independent

3.4 Gradient boosted trees

Gradient boosting consists of three components they are loss function which is to be optimized, a weak learner and an additive model to make the weak leaner to minimize the loss function.

3.4.1 Pseudo code

BEGIN

Initialize the list of weak learners to forest

For each epoch

Update the weight of the examples for the weak learners

Estimate the new weak classifier

Computer weight of new weak classifier

Add the pair (new weak classifier , weight of new weak classifier) to the forest

Return the forest

End for

END

3.4.2 Advantages

- Boosting algorithm is generic one irrespective of change in the loss function
- It can use the sub samples of the training for the new tree creation

3.4.3 Disadvantages

- The prediction of each tree are added sequentially which can slow down the learning
- Proper tuning of parameters is essential for better results

3.5 Deep learning

Here we use a multi layer feed forward neural network which uses back propagation for building the deep learning environment.

3.5.1Pseudo code

BEGIN

Initialize the neural network with the inputs and corresponding weights

For each hidden layer

The activation function is calculated and the weights are initialized

End For

The weights are updated in each hidden layer as per the gradient descendent of the output layer

Repeat until the error is minimized to the minimum level

END

3.5.2 Advantages

- It gives the better accuracy
- It is versatile in nature

3.5.3 Disadvantages

- Time consuming
- Complex in nature

4. Experimental environment

The experiment is carried out with the Rapid miner tool. RapidMiner is a software platform developed by the company of the same name that provides an integrated environment for machine learning, data mining, text mining, predictive analytics and business analytics. It is used for business and commercial applications as well as for research, education, training, rapid prototyping, and application development and supports all steps of the data mining process including data preparation, results visualization, validation and optimization. RapidMiner is developed on an open core model, with the RapidMiner Basic Edition available for download under the AGPL license [6].

4.1. Dataset used

There are four dataset has been used for this experiment which are available in Rapidminer.

Table 4.1: Dataset description

Name of the Dataset	No. of Attributes	No. of Instances
Deals	4	1000
Ripley set	3	250
Sonar	61	208
Weighting	7	500

4.2 Validation used

The test bed has been created in the Rapidminer miner tool. The dataset is loaded and the classification operators are acted on it. The Cross validation operator is used for validating the learner.

The Cross Validation operator is a nested operator. This operator has two sub processes, one for training and another for testing. The trained model is followed by testing. The performance is measured during the second phase.

4.3 Performance Metrics used

The performance metrics used for the experiment is given below

4.3.1 Accuracy:

Accuracy is how close a measured value is to the true value. It expresses the correctness of a measurement and determined by absolute and comparative way.

$$Accuracy = \frac{Sum\ of\ true\ positives + Sum\ of\ true\ negatives}{Total\ population}$$

4.3.2 Classification error

Relative number of misclassified examples or in other words percentage of incorrect predictions.

$$Classification\ Error = \frac{Sum\ of\ false\ positives + Sum\ of\ false\ negatives}{Total\ population}$$

4.3.3 Kappa

The Kappa statistic (or value) is a metric that compares an Observed Accuracy with an Expected Accuracy (random chance).

$$Kappa = \frac{Accuracy - Random\ Accuracy}{1 - Random\ Accuracy}$$

Where accuracy is simply the sum of true positive and true negatives, divided by the total number of items

$$Accuracy = \frac{Sum\ of\ true\ positives + Sum\ of\ true\ negatives}{Total\ population}$$

Random Accuracy is defined as the sum of the products of reference likelihood and result likelihood for each class. That is

$$Random\ Accuracy = \frac{a + b + c + d}{(\text{Total Population})^2}$$

Where $a = sum\ of\ true\ negative + sum\ of\ false\ positive$

$$b = sum\ of\ true\ negative + sum\ of\ false\ negative$$

$$c = sum\ of\ false\ negative + sum\ of\ true\ positive$$

$$d = sum\ of\ faslse\ positive + sum\ of\ true\ positive$$

4.3.4 Weighted mean recall

The weighted mean of all per class recall measurements. It is calculated through class recalls for individual classes.

$$Recall = \frac{Sum\ of\ true\ positives}{Sum\ of\ true\ positives + Sum\ of\ false\ negatives}$$

4.3.5 Weighted mean precision

The weighted mean of all per class precision measurements. It is calculated through class precisions for individual classes

$$Precision = \frac{Sum\ of\ true\ positives}{Sum\ of\ true\ positives + Sum\ of\ false\ positives}$$

5. Results and discussions

Five different classifiers are tested against four datasets based on five parameters. The results are tabulated as below.

Table 5.1 Results from deals dataset

Dataset used	**Deals**				
Measures	Classifiers used				
	Naïve	K-NN	GLM	GBT	Deep learning
Accuracy	91	97.33	99.67	91.67	99.33
Error	9	2.67	0.33	8.33	0.67
Kappa	0.82	0.947	0.993	0.75	0.987
W_M recall	91	97.35	99.66	83.33	99.28
W_M precision	91	97.35	99.67	95	99.39

Table 5.2 Results from Ripley set dataset

Dataset used	**Ripley Set**				
Measures	Classifiers used				
	Naïve	K-NN	GLM	GBT	Deep learning
Accuracy	85.33	84	86.67	85.33	88
Error	14.67	16	13.33	14.67	12
Kappa	0.707	0.682	0.733	0.707	0.759
W_M recall	85.36	84.4	86.5	85.47	87.93
W_M precision	85.31	85.41	86.7	85.47	88.04

Table 5.3 Results from Sonar dataset

Dataset used	**Sonar**				
Measures	Classifiers used				
	Naïve	K-NN	GLM	GBT	Deep learning
Accuracy	72.58	77.42	72.58	80.65	82.26
Error	27.42	22.58	27.42	19.35	17.74
Kappa	0.455	0.546	0.449	0.614	0.644
W_M recall	72.92	77.19	72.4	80.83	82.19
W_M precision	73.78	77.88	72.75	81.09	82.29

Table 5.4 Results from weighting dataset

Dataset used	**Weighting**				
Measures	Classifiers used				
	Naïve	K-NN	GLM	GBT	Deep learning
Accuracy	94.67	90	98.67	92.67	99.33
Error	5.33	10	1.33	7.33	0.67
Kappa	0.893	0.8	0.973	0.853	0.986
W_M	94.7	90.12	98.67	92.71	99.24

recall					
W_M precision	94.67	90.27	98.67	92.69	99.41

From the above results obtained it is evident that the deep learning based classifier outperforms the other classifiers in most of the cases.

6. Conclusion

Classification is the functionality mostly used in the day to day life and it is an inevitable one in the data mining task. Most of the real world problem is supervised in nature. The real time applications are demanding for the better classifiers. In this paper the researchers study the various classification techniques such as Bayesian based, nearest neighbor based, Generalized linear model based, Boosting based and with deep learning also. The five classifiers are tested with four synthetic dataset and five parameters are considered. In most of the cases the Deep learning with the back propagation based classifier outperforms other classifiers considered. This encourages the research based on the deep learning for this task. In future we can concentrate on the text based classifiers and the image based classification tasks.

References

[1] https://en.wikipedia.org/wiki/Bayes%27_theorem

[2] http://www.robots.ox.ac.uk/~dclaus/digits/neighbour.htm

[3] P. Keerthana, B.G. Geetha, P. Kanmani, "Crustose Using Shape Features And Color Histogram With K Nearest Neighbor Classifiers", International Journal of Innovations in Scientific and Engineering Research (IJISER), Vol.4, No.9, pp.199-203,2017.

[3] Murty M.N., Devi V.S. (2011) Nearest Neighbour Based Classifiers. In: Pattern Recognition. Undergraduate Topics in Computer Science, vol 0. Springer, London

[4] Yuan, Guo-Xun & Ho, C.-H & Lin, Chih-Jen. (2012). Recent Advances of Large-Scale Linear Classification. Proceedings of the IEEE. 100. 2584-2603.

[5] Madsen, Henrik; Thyregod, Poul (2011). Introduction to General and Generalized Linear Models. Chapman & Hall/CRC.

[6] https://rapidminer.com/