Volume 6, Issue 1, January 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A study on Effects of Intrinsic Characteristics of Datasets on Classification Performance

Kaveri Sharma*, Abhilasha
CSE, GZSCCET, Bathinda, Punjab,
India

Abstract— The classification performance of standard machine learning classification algorithms is highly affected by some properties of the datasets like imbalance in the class distribution of non-linearly separable datasets. The problem of class imbalance becomes more severe in the cases of datasets with class overlap and when there is lack of density in the training set. In this paper the effect of intrinsic characteristics of datasets on the classification performance of various standard algorithms has been analyzed.

Keywords— Characteristics of datasets, Classification, Class imbalance, Class overlap, Lack of density

I. INTRODUCTION

Classification is the task of categorizing an instance into a class chosen from a set of classes. The machine learning algorithms that perform this task are called classifiers. The classifier is fed with training data, from which it forms a model for the classification of new observations based on its learning. It has been observed that the learning of a classifier is greatly affected by some intrinsic properties of the datasets on which it is trained. The imbalance of class distribution has been observed as a major factor for the deterioration of the performance of a classifier. But in recent works it has been observed that when other properties of datasets like class overlap and lack of density also combine with the problem of imbalance of class distribution, they pose serious problems for the classifiers.

In this paper, we review and investigate the correlation of various characteristics of datasets with the loss of the classification performance of classifiers trained on these datasets.

The rest of the paper is organized in the following manner: In section II, the class imbalance problem, the class overlap problem and the problem of lack of density of the minority class are discussed. In section III, some of the solutions to deal with the class imbalance problem are discussed. In section IV, some experiments, performed to analyze the problem, are discussed and finally the conclusions drawn are presented in section V.

II. CHARACTERISTICS OF DATASETS

A. Imbalance in Class Distribution

A dataset in which one class severely out-represents other can be considered as an imbalanced dataset. The class with relatively less number of instances in a dataset is called 'minority class' and the other class is 'majority class'. This is also known as between-class imbalance. Usually, the minority class represents the most important concept to be learned, and it is difficult to identify it since it might be associated with significant and exceptional cases, or because data acquisition of these examples is costly [1,2]. The examples of various domains in which imbalanced datasets can be observed are medical diagnosis of a rare disease, network intrusion detection, fraud detection in banking operations, failure in a manufacturing unit etc.

The relative imbalance exists in datasets where the minority class is not necessarily rare in its own right but rather relative to the majority class. For example, if a dataset of 1,00,000 samples has 100:1 between-class imbalance, it contains 1000 minority examples. If the sample space is doubled, and the distribution does not change, the number of minority examples becomes 2000. The minority class with 2000 examples is not rare in its own right but it is rare as compared to the majority class. Imbalance due to rare instances is representative of domains where minority class examples are very limited, i.e. where the target concept is rare. In this situation, the lack of representative data makes learning difficult [3]. This is called the situation of absolute rarity.

B. Class Overlap

The problem of class overlap appears when classes of a dataset are not linearly separable. Two classes are linearly separable when there exists a hyperplane that can divide the dataspace such that all instances of one class are on one side of the hyperplane and all instance of other class are on other side. If classes of a dataset are not linearly separable, then the classes of dataset do not form separate clusters but instead, they overlap in the data space. So the overlapping region contains data from more than one classes of the dataset. Prati et. Al. in [4], conducted various experiments on synthetic datasets by varying degree of class overlapping and class imbalance and they concluded that the degree of class overlapping and class imbalance have a strong correlation.

C. Lack of Density of Minority Class

The problem of lack of density arises when there is not enough data for the classifier to make generalized rules for classification [1]. This problem becomes severe in case of imbalanced datasets because in such case, if minority class samples are very rare, for example, in case of absolute rarity, then it becomes difficult for the classifier to distinguish them from the noise. In such case, the rules generated on the majority class samples seem to be more generalized to the classifier than the rules generated on minority class and thus, the rules generated on minority class are discarded, leading to high misclassification of minority class instances.

III. SOLUTIONS FOR CLASS IMBALANCE PROBLEM IN LITERATURE

To overcome the class imbalance problem, numerous solutions have been proposed. These solutions can be divided into two main categories:

- Data Level Approaches
- Algorithmic Level Approaches

A. Data Level Approaches

In the Data Level Approaches, the imbalanced dataset is preprocessed to make it balanced. Several data re-sampling techniques are there to change the class distribution in imbalanced datasets.

Re-sampling techniques can be divided into following categories:

1) Oversampling Methods

In over-sampling methods the examples of minority class are increased by either replicating the instances of minority class or by creating new instances from the existing ones. Some popular methods of over-sampling are:

*a) Random Over-sampling:*In Random over-sampling, the minority class examples are replicated i.e. the exact copies of minority class examples are made to make them equal to the majority class instances. Random over-sampling has two main problems:

- It will increase the likelihood of occurring over-fitting, since it makes exact copies of the minority class examples. This results in a model with its decision regions closer and specific to the minority class examples.
- Over-sampling makes the learning process more time consuming if the original dataset is already fairly large but imbalanced [5].

b) SMOTE: SMOTE stands for Synthetic Minority Over-sampling Technique. The minority class is over-sampled by taking each minority class sample and introducing synthetic examples along the line segments joining any of the minority class nearest neighbors. The synthetic examples are created by taking the difference between sample under consideration and its nearest neighbor and multiplying this distance by a random number between 0 and 1 and then adding it to the sample under consideration [6].

c) Borderline-SMOTE: It was purposed by Han et al (2005) that the examples at the borderline and the examples close to the borderline are at more risk of misclassification than the other examples. So the importance of the borderline examples in the classification process is more than other examples that are far from the borderline. This technique is different from SMOTE as it strengthens and oversamples only the borderline examples of the minority class [7].

2)Under-Sampling Methods:

In under-sampling methods the examples of majority class are decreased by removing the instances of majority class to achieve the balance. Some of the methods of under-sampling are:

a) Random under-sampling: In random under-sampling, the instances of majority class are randomly removed i.e. the choice of which examples to be removed is taken randomly. This could result in loss of information leading to degradation of the classifier.

b) Heuristic under-sampling: There are various methods of systematic under-sampling that have been proposed in literature to minimize the information loss during under-sampling.

- Hart in 1968[8] proposed a technique called Condensed Nearest Neighbor Rule (CNN) for finding a consistent subset of examples. Using this method a subset of training examples with all minority examples and one majority example is formed. Then the examples from the original training set are classified using k nearest neighbor algorithm with k=1. If any training example is misclassified, then it is added to the subset. This method stops when all examples from the training set are classified. This main disadvantage of this method is that it is sensitive to noise. If there are noisy samples in the training data, they will get added to the subset, thus increasing the size of the dataset and affecting the generalization ability of the classifier [9].
- Another method is the use of the concept of tomek links in under-sampling. Two points a and b, belonging to different classes and having distance d(a,b) between them, are said to form a tomek link if there does not exist any point c such that distance $d(a,c)<d(a,b)$ or $d(c,b)<d(a,b)$. The examples forming tomek link are either the borderline examples or noise. For the data cleaning purpose, both examples are removed and for under-sampling only the example belonging to the majority class is removed [10].
- Kubat et al. proposed the technique of one-sided selection [11] for under-sampling. In this technique the Condensed Nearest Neighbor Rule is used to form a subset of examples and after that Tomek links are detected and majority class examples belonging to tomek links are removed.
- Jorma Laurikkala proposed the technique called Neighborhood Cleaning Rule [12]. In this technique, 3 nearest neighbors of each example are detected. If the example belong to majority class and 2 or more of its 3 nearest

neighbors are of minority class, then the example is removed. If the example belongs to minority class and two or more of its nearest neighbors are of majority class, then the neighbors belonging to majority class are removed.

- K-medoid is an unsupervised clustering algorithm. It could be used in under-sampling by forming that many clusters in the majority class as there are examples of minority class. After forming the clusters only those examples that form the median of clusters are selected and rest of the majority examples are removed [13].

3)Hybrid Methods :In the hybrid re-sampling methods both under-sampling and over-sampling techniques are applied on imbalanced data to make it balanced. This method balances the disadvantages of over-sampling and under-sampling to some extent especially in cases where the imbalance ratio is very high and the cases where the class clusters are not well defined. Other than random over-sampling and random under-sampling, other advanced re-sampling techniques may also be combined. Batista et al.(2009) proposed a technique of using SMOTE + Tomek links[10]. In this technique, First SMOTE is applied to oversample the minority class and then Tomek links method is applied to remove the example to make clear clusters.

B. Algorithmic Level Approaches

In the algorithmic level approaches, the existing algorithms are adapted to make them appropriate for handling imbalanced dataset. This can be done by various methods such as including misclassification cost in the learning process, shifting decision threshold towards the minority class, recognition based learning etc.

1)Cost sensitive learning: In certain domains, the cost of misclassification of different classes is not equal. For example, in medical domain, if a positive instance (i.e. a person with disease) is misclassified as a negative instance (as a healthy person), then the cost of misclassification is much higher than the negative instance being misclassified as positive, as it could lead to severe consequences. In cost sensitive learning a misclassification cost factor is used to reduce the misclassification of the class of interest. The cost sensitive learning can be divided into two categories:

a) Some algorithms, like decision tree could include the cost factor in the model building [14].

b) The algorithms that could not take cost function directly at model building, could be made cost sensitive by associating meta-cost with them.

The main drawback of this approach is that it requires that the costs of misclassification should be known in prior, which is usually a difficult factor to evaluate. Also it could lead to over-fitting in case where the misclassification costs are very different [15].

2) One-class learning: In one class learning, the classifier is trained only on the minority class in the absence of the majority class instances. The majority class instances are treated as noise. The classification is done by using a similarity threshold that classifies new instances having patterns similar to the learnt minority class as inliers (minority class) and others as noise (majority class). The similarity threshold acts as the decision boundary between classes. If its value is too high, it will result in the misclassification of minority class samples as majority class and if it is too low, it will lead to misclassification of majority class instances to minority class [15]. This method could be applied only on some of the machine learning algorithms like SVM.

IV. EXPERIMENTS

We performed various experiments to evaluate the effects of dataset characteristics on performance of five standard classifiers namely decision tree, neural network, SVM(Support Vector Machine), Naïve Bayes and k-NN (k-Nearest Neighbors with k=5)with their default settings in rapidminer. These experiments are performed on various datasets available on UCI repository [16].

The number of samples of each class in training and test set for each of the dataset are summarized in TableI along with the characteristics of the dataset. For the Iris Dataset 5 fold cross validation is used to evaluate the performance.

TABLE I.DESCRIPTION OF DATASETS

Dataset	**No. of samples**				***Dataset Characteristics***
	Training Set		***Test Set***		
	classI	***classII***	***classI***	***classII***	
Shuttle	85	1194	38	512	Class Imbalance
Iris	50	50	5 Fold CV		Class Overlap
Pima	189	349	79	151	Class Imbalance and Class Overlap
Semi-conductor	76	924	28	539	Class Imbalance, Class Overlap and Lack of density

A. Shuttle dataset

The Shuttle Dataset is imbalanced with more examples of majority class than the minority class. But in this dataset, there is no class overlap. The two classes are linearly separable as shown in Fig.1

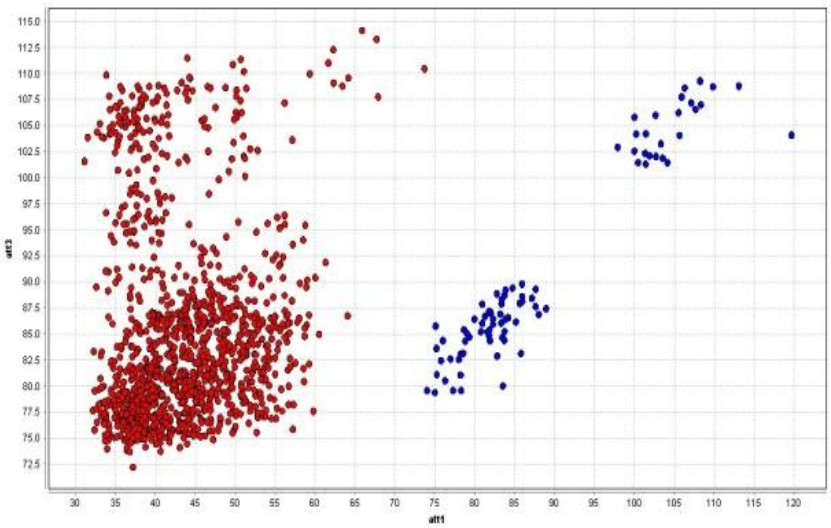

Fig.1 Shuttle Dataset with class imbalance but no overlap

The experiments conducted on this dataset with the algorithms-Decision Tree, Neural Network, SVM, Naïve- Bayes and K-NN resulted in 100% F-Measure. This shows that if the classes are linearly separable then the performance of a classifier is not affected by class imbalance.

B. Iris Dataset

The iris dataset originally has 3 classes. Out of which class 'Iris Setosa' is linearly separable from the other two classes- 'Iris Versicolour' and 'Iris Virginica'.To conduct the experiment on non linearly separable classes, only 'Iris Versicolour'(labeled as 0) and 'Iris Virginica'(labeled as 1) are considered. So in this experiment both the classes in the dataset are balanced with equal number of samples but they are non-linearly separable as shown in Fig.2 The results of various algorithms on Iris dataset are summarized in TableII.

The K-NN algorithm outperformed other algorithms with F-measure 96.09%. This shows that if dataset is balanced but not linearly separable, then the performance of distance-based classifiers is not affected by class overlap.

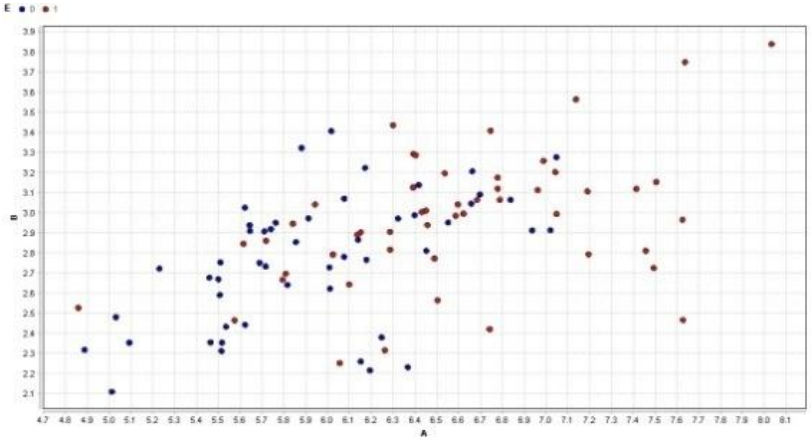

Fig.2 Iris Dataset without class imbalance and with class overlap

TABLE II. F-MEASURE FOR IRIS DATASET

Algorithm	***F-Measure (%)***
Decision Tree	89
Neural Network	95.23
SVM	95.04
Naïve Bayes	94
K-NN	96.09

C. Pima Dataset

The Pima dataset is an imbalanced dataset with class overlap but there is no problem of lack of density as there are 189 examples of minority class and 349 examples of majority class. This dataset is normalized before performing any experiments. The scatter plot of this dataset is shown in Fig.3.

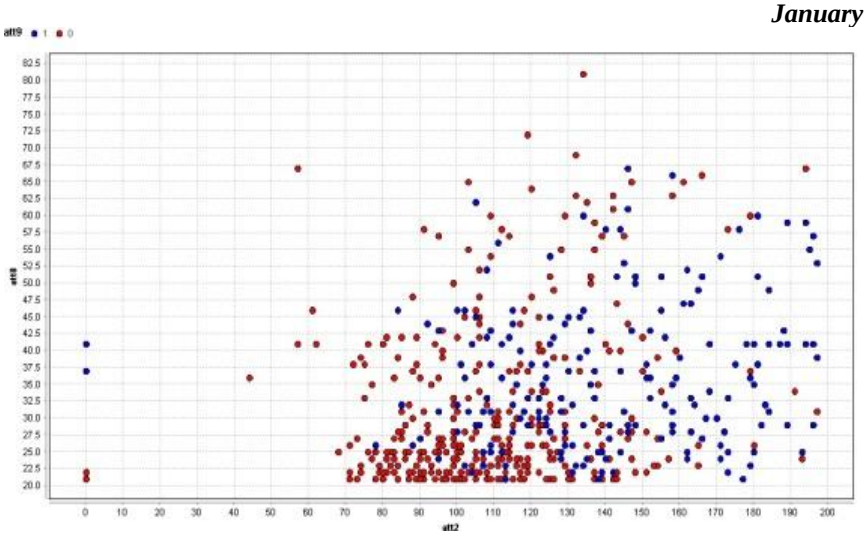

Fig.3 Pima Dataset with class imbalance and class overlap without lack of density

To combat the class imbalance problem, some re-sampling techniques are also used in experiments like random-oversampling, random under-sampling and hybrid re-sampling. One class learning is employed using One class SVM in python. The results of various algorithms on Pima dataset are summarized in TableIII.

TABLE III. F-MEASURE FOR PIMA DATASET

Algorithm	***F-Measure (%)***			
	No-resampling	***Over-sampling***	***Under-sampling***	***Hybrid sampling***
Decision Tree	54.55	61.32	62.44	51.20
Neural Network	63.49	64.56	67.05	61.97
SVM	62.22	67.97	69.18	70.89
Naïve Bayes	20	67.48	67.90	67.90
K-NN	63.01	67.86	67.80	68.64
One-Class SVM	62.65			

The experiments conducted on this dataset show that the performance of classification algorithms is affected by the presence of class imbalance and class overlap together. There is an improvement in performance when some re-sampling techniques are applied. The SVM (Support Vector Machines) algorithm gave best performance after hybrid random under-sampling and random over-sampling. For Naïve Bayes algorithm there is a significant increase in F-Measure after re-sampling.

D. Semiconductor Dataset

The Semiconductor dataset is a high dimensional dataset with 591 features and the problems of class imbalance and class overlap. Also there is the problem of absolute rarity as the number of examples belonging to the fail class (class 1) is very less as compared to the pass class (class -1) due to nature of the dataset. So in this dataset there are all three cases i.e. class imbalance, lack of density and class overlap. To reduce the dimensionality of the dataset, feature selection techniques of weight with PCA (Principal Component Analysis) and brute force(with w-SimpleCART) are used in rapidminer, and two features are selected to perform the experiments. To experiment with one-class learning, one class SVM is used in python. The dataset is normalized before performing experiments. The scatter plot of this dataset is shown in Fig. 4. The results of various experiments performed on this dataset are summarized in TableIV.

It is evident from the results of experiments on this dataset that the presence of class overlap and class imbalance with lack of density of minority class severely degrades the performance of classifiers trained on this data. When no re-sampling technique is used, no standard classifier was able to recognize any minority class sample. This means that the minority class was completely ignored by the classifiers during the learning process. After re-sampling, performance improvement is observed. The K-NN algorithm performed best after random over-sampling resulting in 12.95% value of F-Measure.

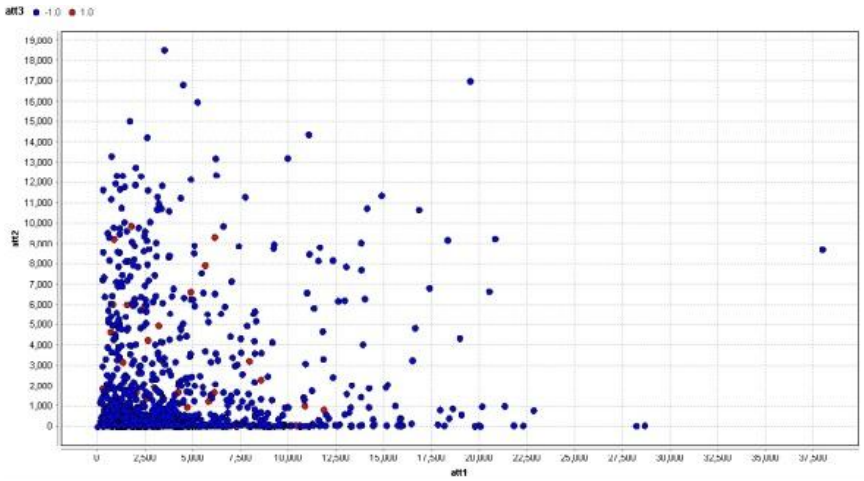

Fig.4 Semiconductor Dataset with class imbalance, class overlap and lack of density of minority class.

TABLE IV. F-MEASURE FOR SEMICONDUCTOR DATASET

Algorithm	***F-Measure (%)***			
	No-resampling	***Over-sampling***	***Under-sampling***	***Hybrid sampling***
Decision Tree	0	8.81	9.24	5.51
Neural Network	0	9.90	7.90	9.63
SVM	0	7.16	6.40	6.73
Naïve Bayes	0	7.23	7.35	7.43
K-NN	0	12.95	7.35	10.77
One-Class	3.88			

V. CONCLUSIONS

In this paper we conducted various experiments on datasets having different characteristics like imbalance in class distribution, class overlap and lack of density to study their impact on the performance of standard machine learning classifiers. It has been observed that the performance of many standard machine learning classifiers is not affected when these characteristics are present in isolation. But serious performance degradation occurs when these characteristics are present together. Also, it is seen that use of Data Level Approaches like re-sampling, resulted in better classification performance than use of Algorithmic Level Technique like one-class SVM in case of complex datasets. However, the impact of various feature selection techniques, employed in pre-processing, on classification of such datasets is left as future scope.

.

REFERENCES

[1] López, Victoria, Alberto Fernández, Salvador García, Vasile Palade, and Francisco Herrera. "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics." Information Sciences 250 (2013): 113-141.

[2] G.M. Weiss, Mining with rare cases, in: O. Maimon, L. Rokach (Eds.), The Data Mining and Knowledge Discovery Handbook, Springer, 2005, pp. 765–776.

[3] He, Haibo, and Edwardo Garcia. "Learning from imbalanced data." Knowledge and Data Engineering, IEEE Transactions on 21, no. 9 (2009): 1263-1284.

[4] Prati, Ronaldo C., Gustavo EAPA Batista, and Maria Carolina Monard. "Class imbalances versus class overlapping: an analysis of a learning system behavior." In *MICAI 2004: Advances in Artificial Intelligence*, pp. 312-321. Springer Berlin Heidelberg, 2004.

[5] Guo, Xinjian, Yilong Yin, Cailing Dong, Gongping Yang, and Guangtong Zhou. "On the class imbalance problem." In Natural Computation, 2008. ICNC'08. Fourth International Conference on, vol. 4, pp. 192-201. IEEE, 2008.

[6] Chawla, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research (2002): 321-357.

[7] Han, Hui, Wen-Yuan Wang, and Bing-Huan Mao. "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning." In Advances in intelligent computing, pp. 878-887. Springer Berlin Heidelberg, 2005.

[8] Peter E. Hart. The condensed nearest neighbor rule. IEEE Transactions on Information Theory, 14(3):515-516, 1968.

[9] Wilson, D. Randall, and Tony R. Martinez. "Reduction techniques for instance-based learning algorithms." Machine learning 38, no. 3 (2000): 257-286.

[10] Batista, Gustavo EAPA, Ronaldo C. Prati, and Maria Carolina Monard. "A study of the behavior of several methods for balancing machine learning training data." ACM Sigkdd Explorations Newsletter 6, no. 1 (2004): 20-29.

[11] Kubat, Miroslav, and Stan Matwin. "Addressing the curse of imbalanced training sets: one-sided selection." In ICML, vol. 97, pp. 179-186. 1997.

[12] Laurikkala, Jorma. Improving identification of difficult small classes by balancing class distribution. Springer Berlin Heidelberg, 2001.

[13] Dubey, Rashmi, Jiayu Zhou, Yalin Wang, Paul M. Thompson, Jieping Ye, and Alzheimer's Disease Neuroimaging Initiative. "Analysis of sampling techniques for imbalanced data: An n= 648 ADNI study." NeuroImage 87 (2014): 220-241.

[14] Ling, Charles X., Qiang Yang, Jianning Wang, and Shichao Zhang. "Decision trees with minimal costs." In Proceedings of the twenty-first international conference on Machine learning, p. 69. ACM, 2004.

[15] Phung, Son Lam, Abdesselam Bouzerdoum, and Giang Hoang Nguyen. "Learning pattern classification tasks with imbalanced data sets." (2009): 193.

[16] Asuncion A, Newman D (2007) UCI machine learning repository. http://archive.ics.uci.edu/ml/datasets.html

