Feature Selection Algorithm Based on Association Rules Mining Method

Jianwen Xie, Jianhua Wu, Qingquan Qian
Department of Computer Science of Zhuhai College
Jinan University
Zhuhai, China
kenxie.4@163.com, tjhwu@jnu.edu.cn, qqq@home.swjtu.edu.cn

***Abstract*—This paper presents a novel feature selection algorithm based on the technique of mining association rules. The main idea of the proposed algorithm is to find the features that are closely correlative with the class attribute by association rules mining method. Experimental results on several real and artificial data sets demonstrate that the proposed feature selection algorithm is able to obtain a smaller and satisfactory feature subset when compared with other existing feature selection algorithms. It is a new feature selection algorithm with vast of application prospect and research value.**

Keywords-feature selection; machine learning; Apriori algorithm; association rules

I. Introduction

Feature selection is a common technique used in data preprocessing for pattern recognition, machine learning and data mining. It performs to remove redundant and noisy features from high-dimensional data sets and select a subset of relevant features for building robust learning models [1]. With a limited amount of training data, excessive amount of features may cause a significant slowdown in the learning process, and may increase the risk of the learned classifier to over-fit the training data because irrelevant or redundant features confuse learning algorithms [2]. Thus it can be seen that selection of well feature subset can bring great benefits to real application of inductive learning.

Feature selection is a classical problem in the field of statistics, and also an important topic in machine learning, which is mainly studied from the perspective of statistics and information processing, and generally refers the low-dimensional data sets and assumes that the features are independent on each other [3-5]. In the field of machine learning, feature selection has great significance in text categorization, data mining, bioinformatics, computer vision, information retrieval, and time series prediction. Due to the development of information acquirement and storage, the dimension and amount of data stored in databases for some real-world application gets high increasingly. The existing feature selection algorithms are severely challenged, and we urgently need some feature selection algorithms that adapt to mass data and also have good accuracy and running efficiency.

At presents, feature selection has attracted high attention of scholars in the field of machine learning. The main reasons can be summarized as following two points: (1) the performances of some learning algorithms are affected by irrelevant and redundant features. Some researches indicate: with the number of irrelevant features increasing, the amount of training data increases exponentially [6-7]. Consequently, feature selection not only reduces computational complexity and improves classification accuracy, but also helps finding easier algorithmic models; (2) mass data processing problems with high-dimension features appear continuously. The development of data mining brings an urgent demand on large-size data processing, such as information retrieval, gene analysis, etc. It is an experiential axiom that high-dimensional feature space is not suitable for machine learning. “Dimension disaster” or “combination explosion” is fatal for some learning algorithms. Therefore, feature selection is required for dimension reduction in case with mass data.

According to the mode of combining the learning algorithm, the existing feature selection strategies can be mainly categorized into three groups: embedded methods, filter methods and wrapper methods [8-10]. With respect to the structure of embedded method, feature selection algorithm as a component is embedded into the learning algorithm, such as some classification algorithms that are implemented by adding and eliminating features. The decision tree algorithm is representative among the embedded models, which selects the feature with the greatest potential classification ability in each node for dividing subspaces. Filter methods evaluate the goodness of the feature subset by using the intrinsic characteristic of the data. They are relatively computationally cheap, because they do not involve the learning algorithm. However, they also take the risk of selecting subsets of features which may not match the chosen learning algorithm [9]. The typical filter methods are ReliefF algorithm, chi-squared (χ^2) feature selection, information gain (IG) based feature selection, gain ratio (GR) based feature selection, symmetrical uncertainty (SU) based feature selection, etc. Wrappers methods use a search algorithm to search through the space of possible features and directly evaluate each subset by running a learning algorithm on the subset. As a result of wrapping the learning algorithm as evaluation tool, they generally outperform filter methods in terms of accuracy, but are computationally expensive and have a risk of over fitting to the learning algorithm [10]. The feature selections methods using genetic search or greedy hill climbing search are representative wrappers methods.

978-0-7695-3641-5/09 $25.00 © 2009 IEEE
DOI 10.1109/ICIS.2009.103

In this paper, we makes a great effort to apply association rules mining techniques to solve feature selection problems, and attempt to produce a small size feature subset that is acceptable for classification tasks.

II. MINING ASSOCIATION RULES

A. Association analysis

Association analysis is a methodology that is useful for discovering interesting relationships hidden in large data set. It was initially applied to market basket data for finding relationships existing among the sales of the products which can help retailer identify new opportunities for cross-selling their products to customers. The development of association analysis can be traced back to AIS algorithm [11] proposed by R. Agrawal in 1993. AIS algorithm doesn't utilize the property of the frequent itemsets, which result in unnecessarily generating and counting too many candidate itemsets. Subsequently, R. Agrawal and R. Srikant introduced the property of the frequent itemsets and proposed Apriori algorithm [12], which is a main technique widely used in commerce at present. Under the influence of Apriori, many researchers were attracted into the field of research on mining association rules, and proposed many improved algorithms, e.g. Apriori-Hybrid algorithm (R. Srikant et al) [12], fuzzy association rule algorithm (C. M. Kuok et al) [13], FP-Growth algorithm (J. Han et al) [14] etc.

B. Related terms

1) Itemset and support count

Let $I=\{i_1, i_2,\ldots, i_d\}$ be the set of all items, and $T=\{t_1, t_2,\ldots, t_N\}$ be the most of all instances. Each instance t_i contains a subset of items chosen from I. In association analysis, a collection of zero or more items is termed an itemset. If an itemset contains k items, it is called a k-itemset. An important property of an itemset is its support count, which refers to the number of instances that contain a particular itemset. Mathematically, the support count, $\sigma(X)$, for an itemset X can be stated as follows:

$$\sigma(X) = |\{t_i \mid X \subseteq t_i, t_i \in T\}|,$$

where the symbol $|\cdot|$ denotes the number of elements in a set [15].

2) Support and confidence

An association rule is an implication expression of the form $X \rightarrow Y$, where X and Y are disjoint itemsets. The strength of an association rule can be measured in terms of its support and confidence [15]. Support determines how often a rule is applicable to a given data set, while confidence determines how frequently items in Y appear in instance that contains X. The formal definitions of these metrics are

$$\textit{support}\text{: } s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N},$$

$$\textit{confidence}\text{: } c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}.$$

3) Lift

Lift is a correlation measure that can be used to augment the support-confidence framework for association rules, and it is given as follows. The occurrence of itemset A is independent of the occurrence of itemset B if $P(A \cup B) = P(A)P(B)$; Otherwise, itemsets A and B are dependent and correlated as events. This definition can easily be extended to more than two itemsets. The lift between the occurrence of A and B can be measured by computing

$$\textit{lift}(A, B) = \frac{P(A \cup B)}{P(A)P(B)}.$$

If the $\textit{lift}(A, B)$ is less than 1, then the occurrence of A is negatively correlated with the occurrence of B. If the $\textit{lift}(A, B)$ is greater than 1, then A and B are positively correlated, meaning that the occurrence of one implies the occurrence of the other. If the $\textit{lift}(A, B)$ is equal to 1, then A and B are independent and there is no correlation between them [15].

C. Apriori algorithm

Apriori algorithm is the first association rules mining algorithm that pioneered the use of support-based pruning to systematically control the exponential growth of candidate itemsets. It divides the procedure of mining association rules into two steps:

The first step is to iteratively find out all frequent itemsets whose supports are not less than the user-defined threshold. The pseudocode for the frequent itemsets generation step of the Apriori algorithm is shown in Fig. 1. Let C_k denote the set of candidate k-itemsets and F_k denote the set of frequent k-itemsets. The frequent itemsets generation algorithm has two important characteristics: (1) It is a level-wise algorithm; i.e., mapped to the lattice structure, it traverses the itemset lattice one level at a time, from frequent 1-itemsets to the maximum size of frequent itemsets; (2) It uses a generate-and-test strategy for finding frequent itemsets. At each iteration, new candidate itemsets are generated from the frequent itemsets found in the previous iteration. The support for each candidate is then counted and tested against the minimum support threshold [16].

The second step is to construct association rules that satisfy the user-defined minimum confidence by using frequent itemsets. Suppose one of the frequent itemsets is F_k, $F_k = \{i_1, i_2,\ldots,i_k\}$, association rules with this itemsets can be generated in the following way: the first rule is $\{i_1, i_2,\ldots,i_{k-1}\} \Rightarrow \{i_k\}$, by checking the confidence this rule can be determined as interesting or not. Then other rules are generated by deleting the last items in the antecedent and inserting it to the consequence, further the confidences of the new rules are checked to determine the interestingness of them. Those processes iterated until the antecedent becomes empty [15]. The pseudocode of the rules generation step in Apriori is shown in Fig. 2.

```
Algorithm: Frequent itemsets generation in Apriori algorithm
Method:
k=1;
F_k={i|i ∈ I ∧ σ({i})≥N×minsup}; //Find all frequent 1-itemsets
Repeat
    k=k+1;
    C_k=candidates generated from F_{k-1};
    For each instance t∈T do
    C_t=subset(C_k, t); //Identify all candidates that belong to t
        For each candidate itemset c∈C_t do
        σ(c)= σ(c)+1;    //Increment support count
        End for
    End for
    F_k={c|c∈C_k∧σ(c)≥N×minsup}; //Extract the frequent k-itemsets
Until F_k=∅;
Result=∪F_k;
```

Figure 1. Pseudocode of frequent itemsets generation step in Apriori

```
Algorithm: Rules generation in Apriori algorithm
Method:
For each frequent k-itemset f_k, k⩾2 do
    H_1={i|i∈f_k}      //1-item consequents of the rule.
    call ap-genrules(f_k, H_1)
End for

Procedure ap-genrules(f_k, H_m)
k=|f_k|               //size of frequent itemset
m=|H_m|               //size of rule consequent
If k>m+1 then
H_{m+1}=m+1-item consequents generated from H_m
    For each h_{m+1}∈H_{m+1} do
        conf=σ(f_k)/ σ(f_k−h_{m+1})
        If conf≥minconf then
            output: the rule (f_k−h_{m+1})→h_{m+1}
        Else
            delete h_{m+1} from H_{m+1}
        End if
    End for
    call ap-genrules(f_k, H_{m+1})
End if
```

Figure 2. Pseudocode of rules generation step in Apriori

III. Feature Selection Algorithm Based on Association Rules

Association analysis is used for discovering the relationship of things by mining association rules in large size database or data warehouse. This paper proposes a new feature selection method that integrating the theory of association analysis in data mining. Its idea is to find the features that are closely correlative with the class attribute by mining strong association rules with its consequence as class attribute in the training data set.

This algorithm mainly includes three phases: generating association rules set, constructing feature set, and testing feature set. In the first stage, we use Apriori algorithm to generate all the association rules whose consequence is class attribute and lift is greater than 1. While applying Apriori, the parameter support and confidence are determined by user. The second stage is to pick rules circularly from the rules set by a certain strategy, as well as adding the attribute that appears in the antecedent of the picked rule into the feature set. The final feature set will be the result of feature selection. The last stage is a testing procedure in which all selected features are evaluated by learning algorithms. In this paper, we use C4.5 [17] classification accuracy as a measure to evaluate the goodness of the feature subset. The main steps of the proposed feature selection algorithm are described detailedly as follows.

Procedure of feature selection based on association rules:

Step 0: Discretize the numerical attributes in the training set D;

Step 1: Generate an association-rule set R by employing *Apriori* algorithm to the training set D;

Step 2: Clean up the feature set F;

Step 3: Select all the rules with consequences as the class attribute from the rule set R, then construct a new rule set R_{class} using these rules;

Step 4: Calculate lift of each rule in R_{class}, and then obtain a new rule set R'_{class} by deleting the rules whose lifts are less than 1.0;

Step 5: Sort R'_{class} on their length in ascending order first, then on their confidence in descending order, and last on support in descending order;

Step 6: Check the loop termination condition: If satisfied, output the feature subset and exit. Otherwise, execute the procedure: pop the first rule r out of R'_{class}, and then add all attributes that appear in the antecedent of r to F;

Step 7: Remove the instances covered by r from D;

Step 8: Recalculate the confidence and support of the left rules according to the processed training set D;

Step 9: Sort R'_{class} on confidence in descending order first, then on support in descending order. Jump to Step 6.

Fig.3 shows pseudocode for the feature selection algorithm based on association rules and its main related procedures. The function apriori (line 3) is to apply standard Apriori algorithm we mention above to generate the association rules. To ensure the success of rules generation, a procedure discretize (line 2) is needed to perform the act of making numerical attributes discrete before executing Apriori. The procedure sort1 (line 9) and sort2 (line 18) are two multiple key sorting, the former is to sort rules on length in ascending order first, then on confidence in descending order, and last on support in descending order; the latter is to sort rules on confidence in descending order first, then on support in descending order. The function pop (line 13) performs the act that getting the top rule out of the rule set. Procedure extract_features_in_antecedent (line 14) executes returning a feature set that includes all attributes that appear in the antecedent of the rule in parameter list. Function filer_train_set (line 16) performs simple deletion work that canceling all instances covered by the transferred rule, and return the deleted instances set to main program. Procedure reset (line 17) is a key step that correcting the support and confidence of the remainder rules in rule set by recalculating them according to the new training set after executing procedure filer_train_set.

```
Algorithm: feature selection based on association rules
Input:
    D, training set;
    minsup, minimum support threshold;
    minconf, minimum confidence threshold;
    C, class attribute;
    cyclenum, cycle number threshold.
Output:
    Result, feature subset.
Method:
    $result=\varnothing$;
    discretize(D);   // Discretize numerical attributes
    R=apriori(D, minsup, minconf);   //Apriori algorithm
    For each rule $r \in R$ do
        If $lift(r)<1 \wedge consequence(r)!=C$ then
            delete r from R;
        End if
    End for
    sort1(R);   //on length in ascending order first, then on confidence in descending order, and last on support in descending order.
    For (int i=1; i<=cyclenum; i++) do
        If $R==\varnothing$ then break
        Else
            r=pop(R);
            F=extract_features_in_antecedent(r);
            $result=\cup F$;
            $D_{delete}$=filer_train_set(r, D);
            reset($D_{delete}$, R);
            sort2(R); //on confidence in descending order first, then on support in descending order.
        End Else
        End If
    End For
    output result;
Procedure reset($D_{delete}$, R)
    N=NumOfTrainData;  //NumOfTrainData is the number of instances before filtering
    For each instance $s \in D_{delete}$ do
        $N=N-1$;
        For each rule $r:(A \to B) \in R$ do
            If r cover s then
                $\sigma(A \to B)= \sigma(A \to B)-1$;
                $\sigma(A)= \sigma(A)-1$;
            Else If $A \subseteq s$ then $\sigma(A)= \sigma(A)-1$;
            End If
            End If
            $sup(r)= \sigma(A)/N$;
            $conf(r)= \sigma(r)/ \sigma(A)$;
        End for
    End for
```

Figure 3. Pseudocode of feature selection algorithm based on association rules

IV. EXPERIMENTS AND ANALYSIS

As we can see, the cycle number is a significant factor in feature selection algorithm based on association rules to control the loop when to stop, which directly affects the final size of the feature subset. Too great cycle number would generate a large size feature subset, while too few may result in that representative features are not enough to reflect the original feature set. Thus, an appropriate cycle number parameter is required. According to some experimentation, we make a simple conclusion regarding the cycle number: Generally, a considerable result can be obtained when the cycle number is defined as 3 to 6. In this paper, for the training sets whose feature number is less than 10, the cycle number is set as 3 in order to guarantee the reduction in the size of feature subset; for those whose feature number is greater than or equal to 10, the cycle number is defined as 6 to gain well integrated effect.

This section is to check the validity and advantage of the proposed feature selection algorithm by experiments and compare it with other algorithms. The algorithms to be compared with would be genetic search based feature selection, ReliefF algorithm, chi-squared feature selection, information gain based feature selection, gain ratio based feature selection and symmetrical uncertainty based feature selection. In the experiments, we use two kinds of data sets: artificial and real data sets. The 10 real data sets come from UCI machine learning repository [18], which is a collection of databases that are used for the empirical analysis of machine learning algorithms. The selected data sets possess various feature dimensions ranging from ones to tens, include different attribute types (numerical attributes only, nominal attributes only and mixed type) and some of them even have missing data. We used 2 types of artificial data sets. One is waveform-40 [18] constructed by Breiman, each class of which is generated from a combination of 2 of 3 "base" waves, and the latter added 19 attributes of which are all noise attributes with mean 0 and variance 1. The other is data sets generated by a preliminary artificial data generation program [19] coded by Isabelle Guyon for a linear 2 class classification problem and its detailed data generation rule can be found in the website of NIPS 2001 workshop on variable and feature selection. In this paper, we generated 2 such data sets and named them artif-1 and artif-2.

We executed various feature selection algorithms on each data set and applied C4.5 algorithm for classification after feature selection. The record of the size of selected subsets and the C4.5 classification accuracy based on 10-fold cross validation are shown in Tab. I and Tab. II respectively. The results of experiments indicates that feature selection algorithm based on association rules can be used to deal with feature selection problem, and it is able to reduce the number of selected features significantly and produces a little improvement in the classification accuracy. It even uses only about one-third of the features required by other feature selection methods to arrive at the similar classification accuracy. The classification accuracies obtained by each feature selection method considered are almost equal and close. But with regard to the number of features selected by each algorithm, it is obvious that effect of reduction achieved by the proposed method is superior to others. In a word, compared with 6 methods on 13 various data sets, feature selection algorithm based on association rules obviously present its advantage in the great reduction of the number of subset and guarantee the classification accuracy acceptable, which can offer an efficient preprocess for the data mining, pattern recognition and machine learning on the data sets.

TABLE I. C4.5 CLASSIFICATION ACCURACY OF VARIOUS FEATURE SELECTION ALGORITHMS (%)

Data set	**Raw**	**Association Rules**	**Genetic Search**	**ReliefF**	**Chi-square**	**Information Gain**	**Gain Ratio**	**Symmetrical Uncertainty**
waveform-40	68	69.3333	73.6667	68.3333	70.6667	70.6667	70.3333	70.6667
artif-1	77	84	88.5	79.5	79.5	79.5	81	79.5
artif-2	75	75.3333	72	77	77.3333	77	77.3333	77
ionosphere	91.453	91.7379	85.4701	90.8832	91.1681	91.1681	90.8832	91.1681
vote	96.3218	95.1724	96.3218	96.3218	96.3218	96.3218	96.3218	96.3218
wine	93.8202	92.1348	94.382	93.8202	93.8202	93.8202	93.8202	93.8202
horse-clonic	81.2709	81.2709	86.6221	81.2709	81.6054	80.9365	80.9365	80.9365
zoo	92.0792	88	93.0693	92.0792	92.0792	92.0792	92.0792	92.0792
breast cancer	94.5637	95.279	95.7082	94.5637	94.5637	94.5637	94.5637	94.5637
monk-1	82.2581	95.9677	95.9677	95.9677	82.2581	82.2581	82.2581	82.258
monk-2	56.213	57.3964	56.213	56.213	56.213	56.213	56.213	56.213
monk-3	93.4426	93.4426	93.4426	93.4426	93.4426	93.4426	93.4426	93.4426
mushroom	100	99.1137	100	100	100	100	100	100
Average	84.7248	86.014	87.0280	86.1074	85.3056	85.2285	85.3219	85.2285

TABLE II. NUMBER OF FEATURES OF THE SUBSET GENERATED BY VARIOUS FEATURE SELECTION ALGORITHMS

Data set	**Raw**	**Association Rules**	**Genetic Search**	**ReliefF**	**Chi-square**	**Information Gain**	**Gain Ratio**	**Symmetrical Uncertainty**
waveform-40	40	8	19	32	19	19	19	19
artif-1	17	6	5	15	9	9	9	9
artif-2	35	5	16	27	9	9	9	9
ionosphere	34	4	13	33	33	33	33	33
vote	16	8	7	16	16	16	16	16
wine	13	4	5	13	13	13	13	13
horse-clonic	27	5	6	25	22	21	21	21
zoo	17	4	7	17	17	17	17	17
breast cancer	9	7	6	9	9	9	9	9
monk-1	7	3	3	3	7	7	7	7
monk-2	7	5	6	6	7	7	7	7
monk-3	7	3	2	3	7	7	7	7
mushroom	22	6	7	21	21	21	21	21
Average	19.31	5.2	7.85	16.92	14.54	14.46	14.46	14.46

V. CONCLUSIONS AND FUTURE WORK

Reducing redundant or irrelevant features can improve classification performance in most of cases and decrease cost of classification. In this work, we propose a new feature selection algorithm based on association rules. We designed an adaptive feature selection strategy embedding Apriori algorithm for finding association rules, which can discover the features that are related to the class attribute according to the theory of association analysis. The experimental results indicate that the feature selection algorithm based on association rules can yield a significant reduction in the number of features required for classification algorithm and simultaneously keep classification accuracy acceptable. That is, there are potential advantages of using the techniques of mining association rules to implement feature selection.

This work shows a novel approach to dealing with feature selection. Though it presents a remarkable advantage in reducing the feature number, due to using Apriori algorithm directly to mining association rules, the time complexity of the algorithm is quite high. To reduce the computational complexity and improve its running efficiency, our future work will be focused on simplifying present way to generate association rules with consequence as class attribute. One solution worthy of considering is to filter out the rules whose consequence is non-class attribute inside the Apriori algorithm. In addition, to improve the classification accuracy, our further work will be a research on wrapper feature selection algorithm based on association rules, which directly use the classification algorithm to evaluate the feature subsets inside the feature selection algorithm.

ACKNOWLEDGEMENTS

The authors thank the anonymous reviewers for their valuable comments and suggestions that helped us for improving our work. This work is funded by the National Natural Science Foundation of China under Grant 50878188.

REFERENCES

[1] K. Fukunaga. Introduction to Statistical Pattern Recognition. Academic Press, San Deigo, California, 1990.

[2] L. Yu, H. Liu. Efficiently handling feature redundancy in high-dimensional data, in: Proceedings of The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-03), Washington, DC, August, 2003, pp. 685-690.

[3] Lewis P M. The characteristic selection problem in recognition system. IRE Transaction on Information Theory, 1962, 8, pp.171-178.

[4] Kittler J. Feature set search algorithms. Pattern Recognition and Signal Processing. 1978, 41-60.

[5] Cover T M. The best two independent measurements are not the two best. IEEE Transactions on System, Man and Cybernetics, 1974, 4(1), pp.116-117.

[6] Langley P. Selection of relevant features in machine learning. Proceedings of the AAAI Fall Symposium on Relevance. Menlo Park, CA.:AAAI Press, 1994,140-144.

[7] Jain A k, Zongker D. Feature selection: evaluation, application, and small sample performance. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(2), pp.153-158.

[8] M. Dash and H. Liu. Feature Selection for Classification. Intelligent Data Analysis, 1997, Vol. 1, No. 3, pp.131-156.

[9] Zexuan Zhu, Yew-Soon Ong, Manoranjan Dash. Wrapper-filter feature selection algorithm using a memetic framework. IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics: a publication of the IEEE Systems, Man, and Cybernetics Society 2007; 37(1): 70-6.

[10] R. Kohavi and G. H. John. Wrapper for Feature Subset Selection. Artificial Intelligence, vol. 97, no. 1-2, pp.273-324, 1997.

[11] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp.207-216, Washington D.C., May 1993.

[12] Agrawal R, Srikant R. Fast Algorithms for Mining Association Rules. VLDB. Sep 12-15 1994, Chile, pp.487-499.

[13] C.M. Kuok, A. Fu, M.H. Wong. Mining Fuzzy Association Rules in Databases. ACM SIGMOD Record, Volume 27, Number 1, March 1998, pp.41-46.

[14] Jiawei Han, Jian Pei, Yiwen Yin, Mining frequent patterns without candidate generation, Proceedings of the 2000 ACM SIGMOD international conference on Management of data, p.1-12, May 15-18, 2000, Dallas, Texas, United States.

[15] Jiawei Han, Micheline Kamber. Data Mining: Concepts and Techniques, Second Edition. Morgan Kaufmann, 2006.

[16] Pang-Ning Tan, Michael Steinbach, Vipin Kumar. Introduction to data mining. Addison Wesley Longman, 2006.

[17] Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.

[18] Asuncion, A. & Newman, D.J. (2007). UCI Machine Learning Repository [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science.

[19] Isabelle Guyon. Preliminary artificial data generation program for a linear 2 class classification problem [http://clopinet.com/isabelle/Projects/NIPS2001/#dataset].