Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 5, No. 3, September 2017, pp. 275~284
ISSN: 2089-3272, DOI: 10.11591/ijeei.v5i3.299

An Heterogeneous Population-Based Genetic Algorithm for Data Clustering

Amina Bedboudi[*1], Cherif Bouras[2], Mohamed Tahar Kimour[3]
[1,3] Departement of Computer Science, University of Badji Mokhtar - Annaba, Algeria
[2] Departement of Mathematics, University of Badji Mokhtar - Annaba, Algeria
e-mail: bedboudi.amina@hotmail.fr

Abstract

As a primary data mining method for knowledge discovery, clustering is a technique of classifying a dataset into groups of similar objects. The most popular method for data clustering K-means suffers from the drawbacks of requiring the number of clusters and their initial centers, which should be provided by the user. In the literature, several methods have proposed in a form of k-means variants, genetic algorithms, or combinations between them for calculating the number of clusters and finding proper clusters centers. However, none of these solutions has provided satisfactory results and determining the number of clusters and the initial centers are still the main challenge in clustering processes. In this paper we present an approach to automatically generate such parameters to achieve optimal clusters using a modified genetic algorithm operating on varied individual structures and using a new crossover operator. Experimental results show that our modified genetic algorithm is a better efficient alternative to the existing approaches

Keywords*: data mining, clustering, K-means, genetic algorithms, crossover*

1. Introduction

As one of the most important tasks of spatial data mining, cluster analysis has been widely used in several domains, such as biology, system engineering and social sciences, in order to identify natural groups in large amounts of data [1-3]. A clustering algorithm assigns a large number of data points to a smaller number of groups (or clusters) such that data points in the same group share the same properties (similar) while, in different groups, they are dissimilar [4]. Data modeling puts clustering in a historical perspective rooted in mathematics, statistics, and numerical analysis [5]. Data mining adds to clustering the complications of very large datasets with very many attributes of different types. This imposes unique computational requirements on relevant clustering algorithms.

A variety of algorithms have recently emerged that meet these requirements and were successfully applied to real-life data mining problems. Especially, popular methods such as K-means, genetic algorithms and their variants are the most notable ones for data clustering [1-9], [14-21]. Their advantages lie in the simplicity, efficiency and ease convergenc, however, most of them suffer from the drawbacks of requiring the number of clusters and their initial centers, but also from the poor clustering quality [9].

In this paper we present an approach to automatically generate such parameters, while improving the speed and accuracy of the algorithm. It is based on an appropriate data structure and process that handles heterogeneous populations in a modified genetic algorithm that operates on varied individual structures .Generally, using parameters such as the crossover and mutation probabilities that adapt to the evolution of the algorithm is a good choice, since higher diversity in the population can be achieved, preventing the algorithm to fall in local minima.

Moreover, to accelerate the genetic algorithm process and increase the individual diversity of the initial population, we generate that initial population in two manners: the first subpopulation is obtained with a deterministic way and the second one with a random way. Increasing the population diversity will allow to achieve better quality.

The rest of the article is structured as follows: section 2 gives an analysis of two data clustering techniques; k-means and genetic algorithms. Section 3 details our data clustering

Received March 25, 2017; Revised July 10, 2017; Accepted August 11, 2017

approach by describing the proposed structure of chromosomes and genetic operators. Section 4 presents the related works and a comparaison with our aproposed approach. Section 5 describes the experimental results and discusses the strength of our approach. Finally, we draw a conclusion and define some future works in section 6.

2. Clustering

Clustering can be thought of as data partitioning or segmenting into homogeneous groups. The clustering is usually accomplished by determining the similarity among the data on predefined attributes. The most similar data are grouped into clusters [2-5], [9].

There is a close relationship between clustering techniques and many other disciplines. Clustering has always been used in statistics and science [7]. Typical applications include speech and character recognition. viewed as a density estimation problem, machine learning clustering algorithms were applied to image segmentation and computer vision [10-13]. In the literature, various methods have been used to handle the clustering problem. The K-means [4-5] [17] is considered one of the major algorithms widely used in clustering [1]. Genetic algorithms [6] are also used in clustering, either in a separated way or combined with-the k-means algorithm [9].

2.1. K-means

K-means first randomly select the k objects; each object initially represents a cluster center. For each remaining object according to its distance from the center of each cluster, assign it to the nearest cluster. Afterward, each cluster center is replaced by the average value on the respective cluster. This process is repeated, until k centers do not change. K-means algorithm attempts to provide a partition of the given data such that the centroids of provided classes are the minima of the following objective function [8-9], [13], [20] :

$$\min_{(\mu_{1,\dots}\mu_{k})} \sum_{h} \sum_{x \mathrm{I} x_{h}} \|x - \mu_{h}\|$$

The typical algorithm is as follows [1]

1. Select k objects as initial centers;
2. Assign each data object to the closest center;
3. Recalculate the centers of each cluster;
4. Repeat steps 2 and 3 until centers do not change;

Due to its practice properties, K-means algorithm is considered as the most popular technique to cluster information. Its main advantages are: i) easy to implement, ii) the comparison is conducted only between the observations and the center of classes, and iii) it detects and isolates the outliers.

However, the algorithm exhibits some drawbacks, especially on the fact that it can be really slow since in each step the distance between each point to each cluster has to be calculated, which can be really expensive in the presence of a large dataset. Moreover, it is sensitive to the initial cluster centers that are provided by the user as well as the k number of clusters.

2.2. Genetic algorithms

Genetic algorithms [6] are robust, stochastic optimization algorithms, used to solve a wide variety of problems. They were developed with the goal of better understanding natural processes such as adaptation, and it belongs to a type of search techniques that mimic the principles of natural selection to develop solutions of large optimization problems. A genetic algorithm finds the optimal value for a particular objective function depending on the problem to be solved. The standard approach to an optimization problem begins by designing an objective function that can model the problem's objectives while incorporating any constraints.

A genetic algorithm consists of: Chromosomal Representation, initial population, fitness evaluation, selection and reproduction (i.e., crossover and mutation). GA operates by maintaining and manipulating a population of potential solutions called chromosomes. Each chromosome has an associated fitness value which is a qualitative measure of the goodness of the solution encoded in it.

In a genetic algorithm process, we determine firstly an initial solution, composed of a set of chromosomes (initial population), and iteratively apply reproduction operators(selection, crossover and mutation) until achieving a certain quality parameter or a predefined number of iterations. A use of a fitness function guides the stochastic selection of chromosomes which are then used to generate new candidate solutions through crossover and mutation.

Therefore, basic operations are: i) Crossover, which generates new chromosomes by combining sections of two or more selected parents, ii) mutation, which acts by randomly selecting genes which are then altered; thereby preventing suboptimal solutions from persisting and increases diversity in the population.

There are three main types of selection methods: fitness proportionate selection, ranking method and tournament selection.

In tournament selection [14], individuals are selected randomly from the population, based on the fitness function. Tournaments are often held between pairs of individuals, although larger tournaments can be used. Simple outline of a genetic algorithm is shown in Figure 1.

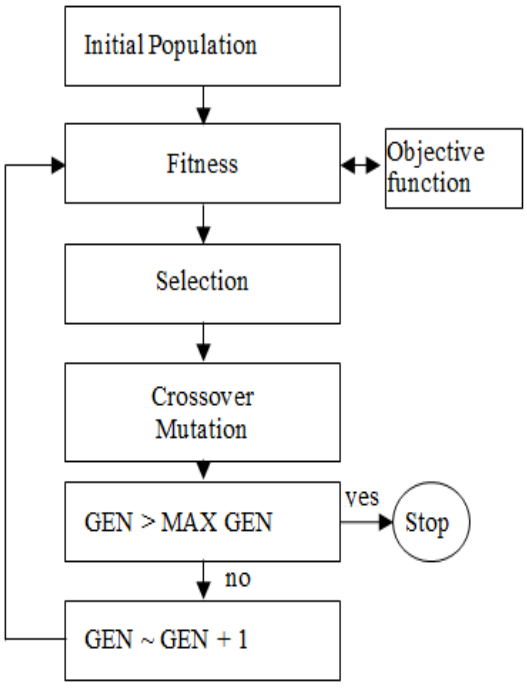

Figure 1. Outline of the genetic algorithm.

Genetic algorithms are characterized by attributes such as objective function, encoding of the input data, crossover, mutation, and population size.

(1) Objective function. It is used to assign each individual in the population a fitness value; an individual with a higher fitness represents a better solution to the problem than an individual with a lower fitness value;
(2) Encoding. Genetic algorithms operate on an en-coding of the problem's input data (which represent in-dependent variables for the objective function);
(3) Elitism. This is a way to ensure that the highly fit-ting chromosomes are not lost and copied to the new population. Elitism has been found to be very important to the performance of genetic algorithms;
(4) Crossover. It is a procedure in which a highly fit-ting chromosome is given an opportunity to reproduce by exchanging pieces of its genetic information with other highly fitting chromosomes;
(5) Mutation. This is often applied after crossover by randomly altering some genes to individual parents;
(6) Population size. It is the number of individuals in a population. The larger the population size, the better the chance that an optimal solution will be found.

Genetic algorithms iterate a fixed number of times. Since the function's upper bound (the maximum fitness value possible for an individual) may not be known or cannot be reached, we must limit the number of generations in order to guarantee the termination of the search process. This may result in a suboptimal solution. Moreover, combined approaches of genetic algorithms with other clustering techniques are still not satisfying on some user requirements especially for the accuracy and execution time

3. The Proposed Approach

The aim of our approach is to give an efficient clustering process leading to better enhanced clusters' quality and execution time using a varied structure-based genetic algorithm with a new crossover operator. Tthe input of such modified genetic algorithm process is a dataset. The genetic process automatically generates the number of clusters with their initial centroids

3.1. Chromosome representation

Chromosome representation is a problem encoding performed at the most important steps in using genetic algorithm to solve a problem. To encode both the number of clusters and their centers, we propose a chromosome structure that applies a real coded genetic algorithm to the clustering problem, where crossover and mutation operators are applied directly to real parameter values.

The use of real parameter values in the GA representation has a number of advantages over binary coding. The efficiency of the GA is increased as there is no need to convert the solution variables to the binary type, less memory is required, there is no loss in precision by discretization to binary or other values, and there is greater freedom to use different genetic operators. In this work, the existing population is processed in the form of several subpopulations with different size. In other words, the number of genes in every cluster differs.

A chromosome in a subpopulations may contain 2 genes, 3 genes to K_{max} genes, where $K_{max}$is the maximum number of clusters. These genes hold corresponding information about the cluster centers. Here, each gene in the chromosome represents cluster center, and number K of genes in a chromosome stands for number of clusters. The value of K is assumed to lie in the interval [$2; K_{max}$]. Thus, each solution i is a fixed-length string represented by cluster centers c_{ij} ; j = 2,…,K_{max}. Then, solution i of the population is represented via a vector as follows:

c_{i1}	…	c_{ij}	…	c_{iKmax}

Figure.2. An example of chromosomes as defined by our approach.

3.2. Fitness function

The fitness function *f* is a measure of profit we need during optimization. It is an objective function that is used to summarize how close a given solution is to achieving the aims and has an important effect on success of a genetic algorithm. Fitness is proportional to the utility or ability of individual which the chromosome represents. Measure of fitness helps in evolving good solutions and implementing natural selection. In this work, the fitness of a chromosome is computed using Mean Square Error (MSE) [18]. The MSE calculate the error between the clusters.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_{i-}y_i)^2 \quad (1)$$

$$Fitness = \frac{1}{MSE}$$

After the definitions of the fitness function the different parameters of the genetic algorithm operators are fixed.

3.3. Generation of the Initial Population

To accelerate the genetic algorithm process and increase the individual diversity of the initial population, we generate that initial population in two phases. The first phase consists in a deterministic producing manner, and the second one consists in a randomly generation. We proceed such generation so to obtain 20% of the initial population from the first phase and 80% from the second phase. his phase, we deterministically generate the first sub-population that constitutes 20% the entire population. In From the input dataset, we produce a sorted dataset. Then, the sorted dataset is divided into k equal segments and the means, modal, and min-max values (see Algorithm1) of each segment are determined and considered asinitial cluster centers. In doing so, we increase the diversity of individuals leading to enhanced genetic reproduction.

Algorithm1: generate20%population

- Sort the datset;
- For (k=2, k<=Kmax-1, k++){
  - Divide the sorted data int k equal segments: {Sk};
  - Modal step: take the modal value of each segment k as a center Ck;
  - Means step: take the means value of each segment k as a center Ck;
  - Min-Max step:

```
For (i=0, i<k, k++){
        Calculate c = (max-min)/2
        Consider c as a ith center
```

3.4. Reproduction

Genetic Algorithms aids to look for the best solution among a number of possible solutions throw reproduction. Reproduction can be implemented in an algorithmic form, based on reproduction operators, suited to the encoding scheme.The objective of the reproduction operators is to ensure diversity in the population, such that the fittest solutions can be derived through the evolutionary process. Such evolution process is composed of crossover and mutation operations.

During such process, invalid chromosomes may be produced. In the proposed chromosome representation, repetition of genes produces invalid chromosomes because a data point cannot be center point of more than one cluster. So to be able to detect production of invalid chromosomes, we sort the chromosome genes in ascending order. In doing so, we aim to achieve small length, fast detection of invalid chromosomes, and faster crossover and mutation operations. Figure 3. illustrates our crossover process using a mask.

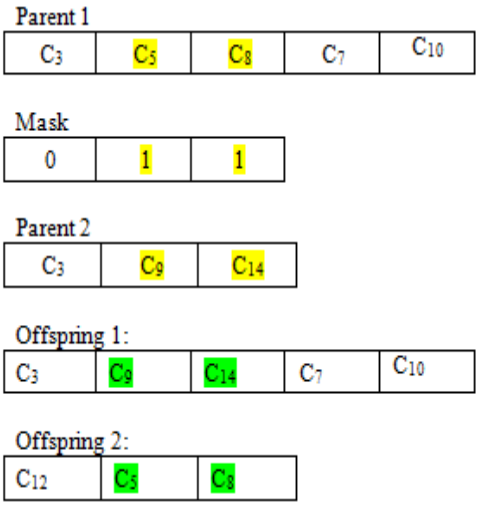

Figure 3. The crossover process using a mask

3.4.1. Crossover

It is a reproduction process that the children chromosomes are generated according to the fitness function values of their parents. We define a crossover operator in such a way that it could accept parent chromosomes with different number of genes. It occurs with probability *crossp*. For each two selected parents, crossover is applied, producing two offsprings. To this end, a binary mask is created. Its length is equal to that of the shorter parent. In this mask, digit "1" occurs with a certain probability *p*.

$$C^{(1)}_j = C^{(1)}_j + \alpha\, C^{(2)}_j \quad (2)$$
$$C^{(2)}_j = C^{(2)}_j + (1-\alpha)\, C^{(1)}_j \quad (3)$$

Figure 4. Depicts the crossover formulas with α as a uniformly distributed random number such that

$\alpha \in [-1, 1]$. Offspring O_1 is created by applying the formula (1) and offspring O_2 is created by applying the formula (2). These formulas in Figure 4 are applied on genes in the parents corresponding to the positions in the mask with a digit "1". Also, we should check the validity of the resulting chromosomes to avoid redundant centers in one chromosome. In the case of redundancy, we replace the redundant gene with a non redundant one, extracted from a parent.

Algorithm2: Clustering with GA
Input: data points, kmax
Output: clusters
Procedure:
(1) Use algorithm 1 to generate 20% of the initial population;
(2) Randomly generate the remaining 80% of initial populations;
(3) Calculate the fitness values of the chromosomes in the initial population;
(4) Select the two fittest chromosomes (P_1, P_2) and randomly choose a target one P_3 from the population;
(5) Apply the crossover and mutation operators on (P_1, P_2) to generate two offsprings (O_1, O_2);
(6) Calculate the fitness values of the offsprings
(7) Replace P_1 and P_2 with O_1 or O_2 , if they are not better than such offsprings;
(8) Repeat from step (3) if there is a significant change between generations, or the number of iteration has not been reached;
(9) Final clusters are derived from the fittest chromosome using the one step K-means algorithm [21].

Figure 5. The main steps of the proposed genetic algorithm to clustering.

3.4.2. Mutation

Mutation is intended to prevent falling of all solutions in the population into a local optimum of the solved problem or enhancing the obtained chromosome. Mutation takes place with a lower probability than that of the crossover. We define two types of mutation to be taken place after the crossover. Topological mutation is intended to add or delete genes from an offspring. Gene mutation is intended to modify a selected gene from an offspring. Gene mutation is applied by randomly selecting a gene from an offspring and replacing it with a data point randomly extract from a dataset. Afterward, a validity check is performed to avoid redundant centers and to keep the cluster number in [2, K_{max}].
Our overall genetic process organizes operations as depicted by Figure 5. To find the closest cluster to every point, we apply a k-means operator (KMO), which is a one step of the classical k-means algorithm. In other words, we assign data points to their clusters for each new chromosome using the KMO operator, in order to compute its fitness.

4. Related works

In the k-means algorithm, the most important challenge is related to determining of the number of clusters and their initial centers. Randomly determining initial centers does not guarantee good quality of clustering, due to the fact that at each run of K-Means algorithm on the same dataset may result in a different clusyers. Besides, the process may converge to suboptimal partitions.

In [3], the author proposed to sort the dateset according to their distances from each other, and then divide the sorted data into k equal segments. He takes the first data item of each segment as a corresponding cluster center. In [12], the authors proposed a solution based on a statistical test, with the assymption that a subset of data follows a Gaussian distribution. The proposed process begins with a small number k of clusters, and then increments k at each step based on appropriate condition.

In [17-21] genetic algorithm-based solutions are proposed. They maintain a population of some coded solutions and apply a one step k-means operator to calculate the fitness of a chromosome. In [13], the authors proposed to use genetic algorithm to determine the initial values of the clusters. They use k-means algorithm to evaluate fitness of each candidate center. At the final iteration, the fittest chromosome will be the k-means initial cluster center. As k-means algorithm must be executed for many times, the approach is usually computationally expensive.

These above mentioned methods share the common drawback of searching initial centers while fixing a given number of clusters. Their aims was to find results in fewer number of steps, but because of intensively calculating neighbourhood distance between data points and rearranging them, it would be very slow in partitioning large datasets. When searching for the

initial centroids, they evaluate the concerned criteria separately for different values of k, leading to exeute K-means algorithm for different numbers of clusters. Such a process is usually very time-consuming.

5. Experimental results

To evaluate our approach and test its efficiency, we compare it with k-means, and the closest work to ours of [9], using two selected datasets from UCI repository of machine learning database, which are Iris and Lymphoma datasets. It is worth noting that the first difference of our approach and the two above mentioned methods is that we automatically determine the k number of clusters and their initial centroids. In the following, we show that our approach outperforms at least such two methods when considering two important parameters, that is, the average error and the average execution time. Experiments have been done usin parameters of the Table 1.

Table 1.

Parameter	Value
Maximum number of clusters, Kmax	15
Population size, P	80
Crossover probability	0.6
Mutation probability	0.3
Maximum no. of iterations,	100

In the Iris dataset, each data point has four feature values, which represents the length and the width of sepal and petal, in centimeters. It has three classes with 50 samples per class. The value of k is therefore chosen to be three (clusters).

In the Lymphoma dataset, we find 62 samples consisting of 4026 genes spanning three classes, which include 42 Diffuse Large B-Cell Lymphoma samples, nine Follicular Lymphoma samples, and 11 B-cell Chronic Lymphocytic Leukemia samples

This dataset is to be partitioned into three clusters. Each dataset was use for each method for 10 times and then we determined the average time and error as mentioned in Table 2, which shows the efficiently experiments of the proposed method over K-means and the work of [9], which closest to ours

Table 2. Evaluation results using UCI data sets

Dataset	parameters	k-means	[11]	Proposed approach
Iris	Avg Error	36.53	29.59	19.59
	Avg Time	18.34	10.56	9.34
Lymphoma	Avg Error	44.12	38.49	22.15
	Avg Time	16.15	10.25	8.28

The experiments have been conducted to measure the average error and average execution time parameters for the three methods. For each dataset, average error and average time are listed to show the trade-off between them. From the table 2 it is observed that, our method outperforms k-means and its variants, especially the work in [11], by enhancing the average error and the average execution time.

Because of our proposed population model consists of chromosomes with varied structures, and using improved genetic operators for reproduction, the offsprings will inherit appropriate information from their parent, leading to find an enhanced solution for the concerned problem. Moreover, the deterministic determination of important portion of the initial population will brings high diversity the genetic reproduction process, leading to enhanced quality of the result

5. Conclusion

In this paper we have introduced a variable structure of genetic algorithms to be applied on the data clustering problem without requiring the number and initial centers of clusters. The cluster centroids are calculated by a genetic algorithm with heterogeneous population leading to better results than using random numbers. Moreover, our approach allowed acceleration of the search process by reducing the average execution time while obtaining better quality of data partitions. As a future works, we plan to investigate improved fitness function and other reproduction operators to achieve better performance

References

[1] Mecca GS, Raunich A, Pappalardo. New Algorithm For Clustering Search Results. Data And Knowledge Engineering. 2007; 62: 504-522.
[2] H He, Y Tan. A Two-Stage Genetic Algorithm for Automatic Clustering. Neuro-Computing. 2012; 81: 49-59.
[3] C Raposo, C Antunes, J Barreto. Automatic Clustering Using A Genetic Algorithm With New Solution Encoding And Operators. ICCSA (2). Lecture Notes In Computer Science, Springer. 2004; 8580: 92-103.
[4] K Jain, MN Murty, PJ Flynn. Data Clustering: A Review, ACM Comput. Surv. 1999; 31(3): 264–323.
[5] S García, J Luengo, F Herrera. Data Pre-Processing In Data Mining, Cham. Springer. 2004: 327.
[6] SN Sivanandam, SN Deepa. Introduction To Genetic Algorithms. Berlin Heidelberg New York: Springer. 2008.
[7] K Fukunaga. Introduction To Statistical Pattern Recognition. New York Academic. 1990.
[8] KB Sawant. Efficient Determination of Clusters in K-Mean Algorithm Using Neighborhood Distance. The International Journal of Emerging Engineering Research and Technology. 2015; 3(1): 22-27.
[9] V Chittu, N Sumathi. A Modified Genetic Algorithm Initializing K-Means Clustering. Global Journal of Computer Science And Technology. 2011; 11(2).
[10] K Fukunaga. Introduction To Statistical Pattern Recognition. Newyork: Academic. 1990.
[11] DW Scott. Multivariate Density Estimation. Wiley, New York. 1992.
[12] G Hamerly, C Elkan. Learning the k in k-means. Advances in Neural Information Processing Systems. 2004; 17: 281–288.
[13] G Karegowda, VT Shama, MA Jayaram, AS Manjunath. Improving Performance of K-Means Clustering by Initializing Cluster Centers Using Genetic Algorithm and Entropy Based Fuzzy Clustering for Categorization of Diabetic Patients. In Proceedings of International Conference on Advances in Computing. MSRIT, Bangalore: Springer India. 2013; 899-904.
[14] M Ettaouil, E Abdelatif, F Harchli. Improving The Performance of K-Means Algorithm Using An Automatic Choice of Suitable Code Vectors And Optimal Number of Clusters. Journal of Theoretical and Applied Information Technology. 2013; 56.
[15] Yugal Kumar, G Sahoo. A New Initialization Method To Originate Initial Cluster Centers For K-Means Algorithm. International Journal of Advanced Science And Technology. 2014; 62: 43-54.
[16] SK Nandha Kumar, P Renuga. Reactive Power Planning Using Real GA Comparison With Evolutionary Programming. Intl Journal of Recent Trends In Engineering. 2009; 1(3).
[17] K Krishna, MN Murty. Genetic K-means algorithm. IEEE Transactions on Systems, Man, and Cybernetics. 1999; 29(3): 433-439.
[18] G Delavar, GH Mohebpour. A New Genetic Center Based Data Clustering Algorithm Based on K-Means. International Journal of Mechatronics, Electrical And Computer Technology. 2014; 4(13): 1820-1839.

[19] K Valarmathi, D Devaraj, TK Radhakrishnan. Real-Coded Genetic Algorithm For System Identification and Controller Tuning. Applied Mathematical Modelling. 2009; 33: 3392–3401.
[20] DE Goldberg. Genetic Algorithms In Search Optimization And Machine Learning, Addison Wesley. 1989.
[21] Najmah M, Ghasemzadeh, Ch Meinel. A Variant of Genetic Algorithms For Non-Homogeneous Population. Intl Conf. on Applied Mathematics. Computational Science and Systems Engineering. Taly-Rome. 2016.
[22] Asuncion J. Newman, UCI Mach. Learning Repository, Irvine, CA: Dpt Of Inf. And Univ. of California, Http://Www.Ics.Uci.Edu/~Mlearn/Mlrepository.Html. Computer Science. 2016.