Data Mining with Oracle 10g using Clustering and Classification Algorithms

Nhamo Mdzingwa

September 25, 2005

Abstract

Deciding on which algorithm to use, in terms of which is the most effective and accurate algorithm in data mining, has always been a challenge for most data miners. The objective of this research is fundamentally focused on investigating the effectiveness of two algorithms available in Oracle10g for data mining. These are the K-Means and the O-Cluster algorithms. The second objective is to gather information from the dataset used in the evaluation. Information gathering involves finding predictors of HIV AIDS prevention behaviour attributes.

The results obtained are as follows; the first set is concerned with the evaluation of the K-Means and O-Cluster algorithms. Here it was observed that the O-Cluster algorithm builds more accurate models than the K-Means algorithm and also that the models by the O-Cluster algorithm find more accurate clusters when applied to new data. The second set of results involves gathering information from the dataset. Here the attributes HIV Test and Know AIDS were identified as predictors of prevention behaviour of condom use and abstinence. These were found by distinguishing the clusters found in the dataset.

1 Introduction

The field of data mining is concerned with learning from large quantities of data or even turning dataset into information. The standard approach of acquiring this information is to sift an algorithm through the large dataset in order to build a model, and then apply the model to new data. However, most data miners find it difficult or challenging to select an algorithm to use, since they do not know which algorithm produces the most accurate and effective results.

This document analyzes and discusses in detail the investigation of clustering algorithms provided by oracle10g data mining (ODM). The investigations pay particular attention to accuracy and effectiveness of K-Means and O-cluster algorithm model building as well as pattern discovery. A data miner would be in a better position to select the most accurate algorithm, on gaining understanding of algorithm performance described in this paper.

The structure of the paper is as follows: Section 2 discusses data mining background in brief and highlights related work within the field concerned with algorithm accuracy and effectiveness. Section 3 describes the principles behind the tools used, while Section 4 details the implementation as well as highlighting the methodology adopted for the investigation that is building the models and applying them. Section 5 is an examination and interpretation of the results.

2 Background and Related Work

2.1 Data mining in brief

As already mentioned in section 1, data mining is concerned with the learning and getting information from a dataset by the use of an algorithm to build a model that will be applied to new data to discover patterns. For this purpose, there is a variety of algorithms available in this field of data mining with each belonging to one of the following categories: classification, estimation, prediction, clustering, decision trees or association rules. They all have different functionality and purpose depending on the type of data that they are applied to. Due to this, the question of accuracy and effectiveness of algorithms is taken into consideration by investigating the algorithms.

2.2 Work related to this research paper

Evaluating unsupervised data mining algorithms is a generally difficult task since the goals of an unsupervised data mining session are frequently not as clear as goals as supervised learning. [Roiger et al, 2003, pg58 and pg232] describes techniques of evaluating unsupervised models. The authors explain four main methods namely,

1. *Employ supervised learning to evaluate unsupervised learning.*
2. *Apply alternative technique's measure of cluster quality.*
3. *Create own measure of cluster quality.*
4. *Perform a between-cluster attribute value comparison*.

All the above methods were adopted in order to have an accurate evaluation of the algorithm models built as shall be discussed in this paper. [Roiger et al, 2003] does not explicitly show or prove which algorithm performs better. My research paper however points out which algorithm performs better.

The paper by [Davis E, 2004] is closely related to this research paper. The authors' objective was to determine the algorithm that gives the best performance by the evaluation of algorithm models and results using Oracle data miner 9i. However, the difference between the paper by [Davis E, 2004] and this one is that, this paper is highly focused on the clustering algorithms provided by oracle data miner 10g while [Davis E, 2004] evaluates classification algorithms.

3. Data mining tool used

Oracle Data Mining (ODM) is the data mining software used in the evaluation of algorithms in this research paper. The Oracle Data Mining suite is embedded in the oracle10g Database Enterprise Edition (EE) and is made up of two components, the data mining Java API and the Data Mining Server (DMS). The DMS is a server-side, in-database component that performs data mining that is easily available and scalable. For the purpose of this paper Oracle10g database version 10.1.0.2.0 was installed and configured. The data mining tools and software (Oracle data miner 10g version 10.1.0.2.0) was also installed and configured for use with the database.

4. Implementation

4.1 Data

The dataset for this research was obtained from the Centre for AIDS Development, Research and Evaluation Institute for Social and Economic Research, Rhodes University. It consists of 131 attributes and 899 rows (cases) which is large enough for data mining. Each attribute is a response from an individual, based on a questionnaire relating to HIV AIDS awareness as well as a South African television drama, Tsha Tsha, which is a HIV AIDS awareness program. A particular row represents a collection of an individual's responses. The secondary goal in the research is to find HIV AIDS predictors of prevention behaviour such as use of condom, abstinence and using data mining rather than statistical methods. The dataset was partitioned into two sets. The 1st set was loaded into a database table TSHA_TSHA_BUILD1 which I used to build the models and the 2nd set was loaded into table TSHA_TSHA_APPLY1 to apply the models. Dataset in TSHA_TSHA_APPLY1 will be referred to as new data

4.2 Algorithms

ODM supports the following clustering algorithms as stated by Oracle10g Data Mining Concepts Release 1: [Oracle, 2005] and these were selected for investigation in this research paper.

- *Enhanced version of K-Means*
- *Proprietary O-Cluster algorithm*

Enhanced k-Means (EKM) and O-Cluster algorithms support identifying naturally occurring groupings within the data population. EKM algorithm supports hierarchical clusters, handles numeric attributes and will cut the population into the user specified number of clusters. The algorithm divides the data set into k number of clusters according to the location of all members of a particular cluster in the data. Clustering makes use of the Euclidean distance formula to determine the location of data instances and their position in clusters and so requires numerical values that have been properly scaled [Han et al, 2001].

When choosing the number of clusters to create it is possible to choose a number that doesn't match the natural structure of the data which leads to poor results. For this reason [Berry et al, 2000] says it is often necessary to experiment with the number of clusters to be used. O-cluster algorithm handles both numeric and categorical attributes and will automatically select the best cluster definitions [Berger, 2004].

4.3 Methodology

4.3.1 Building the Models

Each of the two Clustering algorithms in ODM has a setting that is used to tune the algorithm when building models. Both algorithms also have a parameter; the maximum number of clusters (k), this is available so that the user can pre-define the number of clusters that he wishes to find from the dataset.

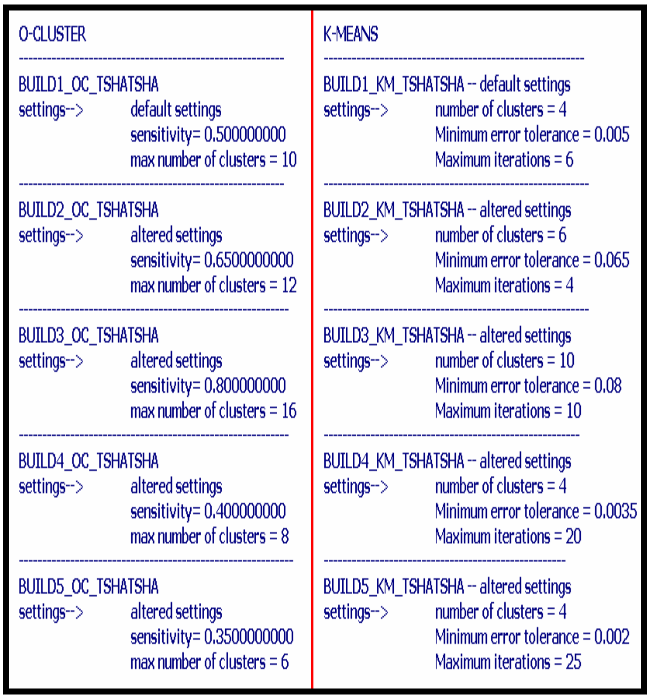
4.3.1.1 Building model by using the K-Means algorithm

There are two settings for this algorithm, Minimum Error Tolerance and Maximum Iterations which determine how the parent-child hierarchy of clusters is formed and can be modified experimentally to observe the changes in cluster definitions. Increasing the tolerance or lowering the iteration maximum will cause the model to be built faster, but possibly with more poorly-defined clusters.

4.3.1.2 Building model by using the O-Cluster algorithm

O-Cluster finds natural clusters up to the maximum number entered as a parameter. That is, the algorithm is not forced into defining a user-specified number of clusters, so the cluster membership is more clearly defined. O-cluster has only one setting; it determines how sensitive the algorithm is to differences in the characteristics of the population. Thus, a higher sensitivity value usually leads to a higher number of clusters.

Ten models were built in total, with five from each algorithm. The model settings or parameters were based on trial and error followed by a critical analysis. A large number of models provide a wide range of models to select the best from, as well as helping monitor if the algorithm settings affect the algorithm's performance. Initially, the first ten models all had distinct values for the maximum number of clusters (k). For these first ten, each model built from the K-Means algorithm was named as follows: BUILD1_KM_TSHATSHA. Models built by the O-cluster algorithm were named as follows: BUILD1_OC_TSHATSHA. The 1 denotes the first model, 2 second, and so on. Figure.1 shows the models and their settings in detail,

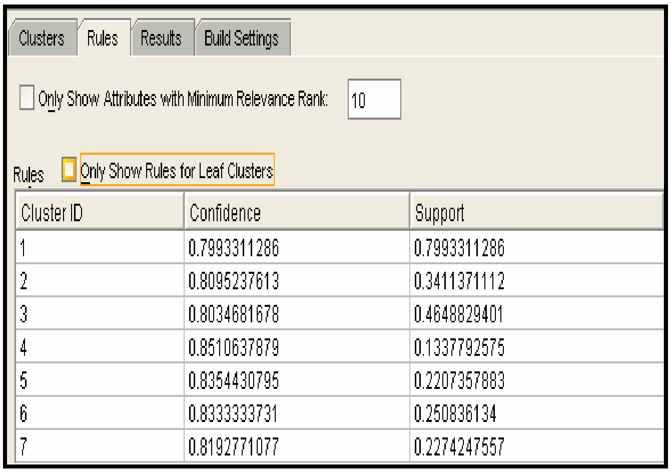


```
O-CLUSTER                                          K-MEANS
-------------------------------------------        -------------------------------------------
BUILD1_OC_TSHATSHA                                 BUILD1_KM_TSHATSHA -- default settings
settings-->     default settings                   settings-->     number of clusters = 4
                sensitivity= 0.500000000                           Minimum error tolerance = 0.005
                max number of clusters = 10                        Maximum iterations = 6
-------------------------------------------        -------------------------------------------
BUILD2_OC_TSHATSHA                                 BUILD2_KM_TSHATSHA -- altered settings
settings-->     altered settings                   settings-->     number of clusters = 6
                sensitivity= 0.6500000000                          Minimum error tolerance = 0.065
                max number of clusters = 12                        Maximum iterations = 4
-------------------------------------------        -------------------------------------------
BUILD3_OC_TSHATSHA                                 BUILD3_KM_TSHATSHA -- altered settings
settings-->     altered settings                   settings-->     number of clusters = 10
                sensitivity= 0.800000000                           Minimum error tolerance = 0.08
                max number of clusters = 16                        Maximum iterations = 10
-------------------------------------------        -------------------------------------------
BUILD4_OC_TSHATSHA                                 BUILD4_KM_TSHATSHA -- altered settings
settings-->     altered settings                   settings-->     number of clusters = 4
                sensitivity= 0.400000000                           Minimum error tolerance = 0.0035
                max number of clusters = 8                         Maximum iterations = 20
-------------------------------------------        -------------------------------------------
BUILD5_OC_TSHATSHA                                 BUILD5_KM_TSHATSHA -- altered settings
settings-->     altered settings                   settings-->     number of clusters = 4
                sensitivity= 0.3500000000                          Minimum error tolerance = 0.002
                max number of clusters = 6                         Maximum iterations = 25
```

Figure.1: Algorithms, model names and their settings with distinct number of clusters.

4.3.2 Interpretation of initial results

After building the ten models from Figure.1 settings, it was observed that each model discovered some clusters. The ODM tool provides an output display for the built models giving a confidence and support value for each cluster found. Part of the output display for each model is shown in Figure.2.



□ Only Show Attributes with Minimum Relevance Rank: 10

Rules □ Only Show Rules for Leaf Clusters

Cluster ID	Confidence	Support
1	0.7993311286	0.7993311286
2	0.8095237613	0.3411371112
3	0.8034681678	0.4648829401
4	0.8510637879	0.1337792575
5	0.8354430795	0.2207357883
6	0.8333333731	0.250836134
7	0.8192771077	0.2274247557

Figure.2: Output produced after BUILD1_OC_TSHATSHA model was built.

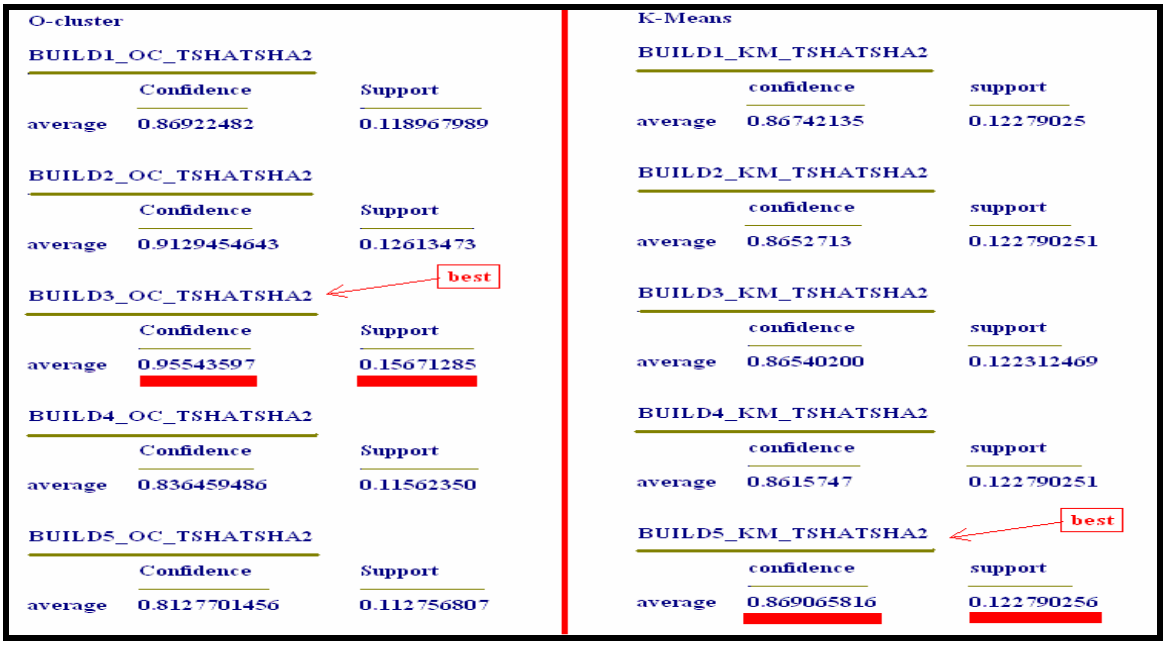
The Confidence is a measure of the homogeneity of the cluster; that is, how close together are the cluster members [ODM Tutorial, 2004]. Due to this reason, I made the Confidence to be a measure of accuracy such that a cluster with the highest confidence value is more accurate and effective than that with a lower value. Thus, a computed average confidence for all the clusters in a model would determine a models' accuracy in discovering clusters.

The support is a measure of the relative size of a cluster (the total need not be 1.00), such that the higher the value the larger the cluster [ODM Tutorial, 2004]. In this paper it is used as an alterative measure to the confidence.

The ClusterID is a value that differentiates the clusters found. The order of numbering used for the ClusterID is as follows; the mining tool generally looks for all clusters in the dataset depending on the algorithm settings. The maximum number of clusters (k) that one sets during model building determines the number of clusters that the mining tool displays as the leaf clusters.

On analysing these average confidence values computed from the model clusters, I observed a high degree of bias in the results. Here the bias is mainly due to the variation in the value of the maximum number of clusters (k) that was set for each algorithm during model building as shown in Figure 1 (algorithm settings). This makes it difficult to determine the best model built from the two algorithms. To overcome the problem, k = 7 for both models was chosen because, the default number of clusters for the k-means algorithm in ODM is 4 and that for O-cluster is 10, therefore setting the value of k for the two algorithms to an average of the two default values was reasonable.

I then re-built 10 more models with the same settings as in Figure.1 but with the maximum number of clusters (k) fixed at 7 for all models. The new model names were in the form BUILD1_OC_TSHATSHA2 for O-cluster and BUILD1_KM_TSHATSHA2 for K-means, with the 2 at the end indicating the second set of built models. This time a more consistent output was achieved giving computed average confidence and average support figures as shown in Figure.3.



O-cluster

BUILD1_OC_TSHATSHA2

	Confidence	Support
average	0.86922482	0.118967989

BUILD2_OC_TSHATSHA2

	Confidence	Support
average	0.9129454643	0.12613473

BUILD3_OC_TSHATSHA2

	Confidence	Support
average	0.95543597	0.15671285

BUILD4_OC_TSHATSHA2

	Confidence	Support
average	0.836459486	0.11562350

BUILD5_OC_TSHATSHA2

	Confidence	Support
average	0.8127701456	0.112756807

K-Means

BUILD1_KM_TSHATSHA2

	confidence	support
average	0.86742135	0.12279025

BUILD2_KM_TSHATSHA2

	confidence	support
average	0.8652713	0.122790251

BUILD3_KM_TSHATSHA2

	confidence	support
average	0.86540200	0.122312469

BUILD4_KM_TSHATSHA2

	confidence	support
average	0.8615747	0.122790251

BUILD5_KM_TSHATSHA2

	confidence	support
average	0.869065816	0.122790256

Figure.3: Computed confidence and support averages for the models built.

From Figure.3, it is evident by analysis that BUILD3_OC_TSHATSHA2 (for O-Cluster) and BUILD5_KM_TSHATSHA2 (for K-Means) have the highest values for both the average confidence and support values. It is also evident that the change in the settings for the K-Means algorithm had little effect on the clusters found hence resulting in very small differences in the computed average values. On the other hand, the O-cluster algorithm models were affected by the change in the settings.

4.3.3 Applying the Models

The two models BUILD3_OC_TSHATSHA2 and BUILD5_KM_TSHATSHA2 which were the best models built from the dataset TSHA_TSHA_BUILD1 as described above, were applied to new data TSHA_TSHA_APPLY1. The models found clusters and the results were loaded in the tables APPLY_OC3_TSHATSHA for the O-cluster and APPLY_KM5_TSHATSHA for the K-Means to facilitate further analysis of the clustering algorithms.

4.3.4 Tests on model results

This section primarily deals with determining cluster quality from the cluster results obtained after applying the cluster algorithm models. Ideally, this involves finding out which algorithm model finds more accurate clusters. Although, indicates that the evaluation of clustering algorithms is difficult, Here I intend to use an evaluation technique by [Roiger et al, 2003]

This technique uses supervised learning evaluation to evaluate unsupervised clustering. Here I make use of a classification algorithm (supervised learning), the Adaptive Bayes Networks (ABN) algorithm with the technique. Making use of the ABN was motivated by the results obtained by [Davis, 2004] which concluded that the algorithm is more accurate in predicting attributes for the classification algorithms in Oracle Data Miner.

Basically, the evaluation technique involves taking the resultant table obtained after applying a cluster model (APPLY RESULTS), pick a random sample of instances (roughly two thirds) from each cluster found, place them in a new database table that will be used to build a classification model, in this case using ABN. The attribute being predicted is identified; in this case it will be the ClusterID. The resultant model is then applied to the remaining instances from the APPLY RESULTS (the one third of instances) with the ClusterIDs removed (stored in excel for comparison later). The results after applying the ABN model which predict the ClusterIDs are then compared to the initial APPLY RESULTS ClusterIDs (the cluster ids in the excel file).

4.3.4.1 Building classification models

Two database tables, build1_abn_FROM_OC and build1_abn_FROM_KM were created and these are used for building the Classification models with the ABN algorithm. The table build1_abn_FROM_OC was loaded with two thirds of instances from the O-cluster model results table (APPLY_OC3_TSHATSHA) created. Two thirds of instances from the table APPLY_KM5_TSHATSHA were also loaded into the table build1_abn_FROM_KM to cater for the K-Means model results evaluation.

The steps taken in building ABN models are clearly explained in the research by [Davis, 2004] and I did is simply adopt these steps. Since [Davis, 2004] concluded that the ABN algorithm provides more accurate results than the Naïve Bayes algorithm in Oracle for classification algorithms, I then decided to use the default ABN algorithm settings in building these models. These settings included a SingleFeatureBuild model type, a maximum number of predictors of 25, a maximum network feature depth of 10 and no time limit for the running of the algorithm.

The two models created were named *OC_abn_Build* from the dataset *build1_abn_FROM_OC* and *KM_ abn_Build* from the dataset *build1_abn_FROM_KM*. Investigating the accuracy of the models built here is unnecessary for this evaluation. This is because any errors or abnormalities found in the algorithm would exert the same effect on both models built since one algorithm with the same conditions (i.e. algorithm settings) is used.

4.3.4.2 Applying the Adaptive Bayes Network (ABN) models

The resultant ABN models were applied to the remaining one third of instances from the respective cluster models. The target attribute in both instances when applying the ABN models is the ClusterID. The results were exported to spreadsheets to allow for inspection and comparisons.

4.3.4.3 Comparison of ClusterIDs

The comparison of ClusterIDs is between the classification model results and the cluster model results. Here I wish to find ClusterIDs that appear in both the two distinct model results. In this case I made a comparison of the K-Means model results with the ABN model results and a comparison of the O-Cluster model results with the ABN model results. Table 1 gives a summary of the tables that were used in the comparison (the tables contain the ClusterIDs) while Table 2 depicts the outcome of the comparison.

Classification table		*Cluster table*
OC_APPLY_ABN	Vs	**APPLY_OC3_ TSHATSHA**
KM_APPLY_ABN	**Vs**	**APPLY_KM5_ TSHATSHA**

Table 1: Database tables compared

Table.2. shows that the classification tables from the results of the ABN model are compared with the clustering table results obtained from the cluster algorithm models. The clustering algorithms basically find clusters in the data; ODM then assigns each data instance to a particular cluster by assigning it a ClusterID. These ClusterIDs are removed from the clustering results and are predicted by the classification algorithm, after which a comparison is made.

DATA SOURCE	CLUSTERIDS **IN BOTH TABLES**	**PERCENTAGE OF IDS IN BOTH MODELS**
From O-Cluster results	42 out of 107	39%
From K-Means results	18 out of 107	17%

Table 2: Results from the comparison of cluster and classification ClusterIDs

Table.3 shows the percentage of the ClusterIDs that appear in both model results. 39% of ClusterIDs appeared in both the cluster model results and classification model results for the O-Cluster algorithm while only 17% for the K-means algorithm. According to this evaluation technique by [Roiger et al, 2003], the percentage outcome is treated as a measure of accuracy for the algorithms, were the greater percentage indicates that that algorithm has more accuracy in finding clusters of high quality. According to Table 3 the O-cluster algorithm has a larger percentage hence is more accurate than the K-Means algorithm.

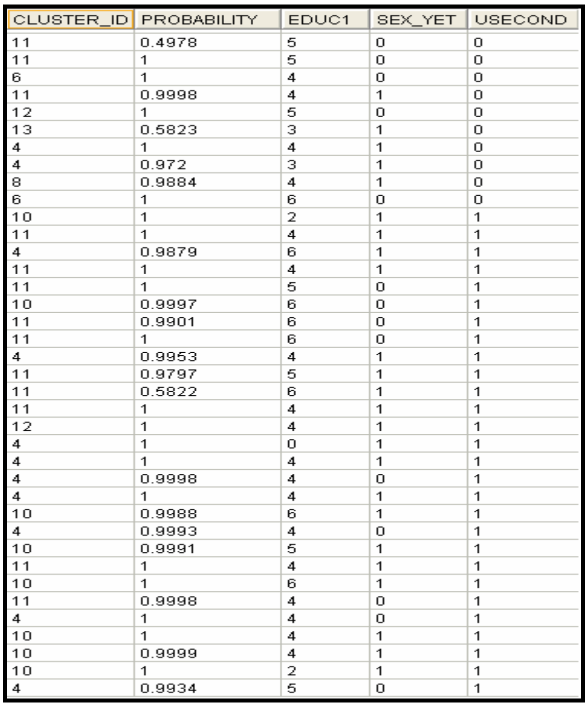
5 Gathering Information from dataset

This section is mainly concerned with gathering information from the dataset. Here I need to find predictors of HIV AIDS prevention behaviour. In order to this it is necessary to define what we mean by predictors of prevention behaviour. This makes it easier and feasible to know what we are actually looking for exactly. Therefore my definition is as follows:

HIV AIDS predictors of prevention behaviour are attributes within our dataset that influence an individual to: (A) use a condom when he/she decides to be sexually active, (B) lead to abstain from having sexual intercourse for at least a year or more and (C) attributes that lead one to having fewer sexual partners. These are the attributes that I want to find from the dataset.

Determining these predictors involved distinguishing the clusters found by the O-Cluster algorithm which I identified in the evaluation process as the most as accurate in building models and finding accurate clusters when applied to new data. After I applied the model to new data the results were exported to spreadsheets.

However from this resultant table APPLY_OC3_TSHATSHA it proves to be difficult to distinguish the clusters due to the large number of attributes in the table (number of attribute is 21). To overcome this I simply re-apply the same O-Cluster model to the same dataset, but in the output table I removed any attribute that I felt had little contribution to solving this problem. This step was repeated nine times with each predictor that was mentioned in my definition of prevention predictors being in each table. Critical analysis of the results was done although the technique used here is a trial and error method. An example of one of the tables is shown below.



CLUSTER_ID	PROBABILITY	EDUC1	SEX_YET	USECOND
11	0.4978	5	0	0
11	1	5	0	0
6	1	4	0	0
11	0.9998	4	1	0
12	1	5	0	0
13	0.5823	3	1	0
4	1	4	1	0
4	0.972	3	1	0
8	0.9884	4	1	0
6	1	6	0	0
10	1	2	1	1
11	1	4	1	1
4	0.9879	6	1	1
11	1	4	1	1
11	1	5	0	1
10	0.9997	6	0	1
11	0.9901	6	0	1
11	1	6	0	1
4	0.9953	4	1	1
11	0.9797	5	1	1
11	0.5822	6	1	1
11	1	4	1	1
12	1	4	1	1
4	1	0	1	1
4	1	4	1	1
4	0.9998	4	0	1
4	1	4	1	1
10	0.9988	6	1	1
4	0.9993	4	0	1
10	0.9991	5	1	1
11	1	4	1	1
10	1	6	1	1
11	0.9998	4	0	1
4	1	4	0	1
10	1	4	1	1
10	0.9999	4	1	1
10	1	2	1	1
4	0.9934	5	0	1

Figure.4: Sample of output table

5.1 Conclusions drawn from the Analysis of Cluster tables

Observations of the tables as explained in previous section make it evident that the attribute HIVTEST influences condom use. Analysis provides clear conclusions that knowing about AIDS leads to abstinence. Therefore from these table observations I have concluded that the attributes HIVTEST and KNOWAIDS have been clearly been identified as predictors of prevention behaviour.

6 Conclusions

Following on from the conclusions and recommendations of the theory research, I have managed to conclude that the O-Cluster algorithm has been identified as the most accurate clustering algorithm in Oracle data mining 10g. The process of finding HIV predictors of prevention behaviour involved critical analysis of the results as well as good reasoning. In conclusion the attributes HIV test and Know Aids were been clearly identified as predictors of prevention behaviour of condom use and abstinence.

References:

[Berger, 2004]. Berger, C., 09/2004, *Oracle Data Mining, Know More, Do More, Spend Less - An Oracle White Paper*, URL: http://www.oracle.com/technology/products/bi/odm/pdf/bwp_db_odm_10gr1_0904.pdf
Accessed: 14 September 2005.

[Berry et al, 2000] *Mastering Data Mining: The Art and Science of Customer Relationship Management*, Michael J.A. Berry and Gordon S. Linoff, USA, Wiley Computer Publishing, 2000

[Davis E, 2004] Emily Davis: *An Evaluation of Commercial Data Mining*. Oracle Data Mining Honours Research Project 2004. URL: http://www.cs.ru.ac.za/research/previous/g01d1801/
Accessed: 14 September 2005

[Roiger et al, 2003] *Data mining: a tutorial- based primer* by Richard J. Roiger and Michael W. Geatz, Boston, Massachusetts, Addison Wesley, 2003, pg 58 and pg 232

[Oracle, 2005], *The Oracle Home Page*. Revised February 2005. Oracle 10g Data mining FAQ http://www.oracle.com/technology/products/bi/odm/odm_10g_faq.html#api Accessed: 24 September 2005.

[ODM Tutorial, 2004] The Oracle Home Page http://www.oracle.com/technology/products/bi/odm/odminer.html Accessed: 24 September 2005