Predictive Analysis Using Data Mining Techniques and SQL

Remedios de Dios Bulos, Georgio F. Go, Giselle O. Ling, Timothy C. Uy and Lawrence J. Yap
De La Salle University
remedios.bulos@dlsu.edu.ph or remedisdedios@yahoo.com

Abstract: Most applications systems store, access and manipulate their data using relational data bases. This research aims to build a Classifier system that analyzes and mines data using Standard Query Language (SQL). Specifically, this paper discusses a Music Genre Classifier system that uses a relational database to accept tuples of audio features as input data and then uses a model that was constructed using a data mining tool but was parsed and converted to SQL statements to predict the class labels of musical compositions. In building the Music Genre Classifier system, jAudio a Digital Signal Processing tool was used to preprocess the input data through the extraction of audio features of musical compositions (songs); WEKA was used to explore several data mining algorithms and to build the prediction model; MS Access was used to accept inputs in relational format and to execute the prediction model in SQL. Classification, clustering, and association rule mining algorithms in WEKA were studied, explored, compared and then the most appropriate technique was selected to develop the system. Particularly, only algorithms that generated decision trees and rules as models were considered since these types of output can be easily parsed and then converted to SQL statements. This paper also discusses how decision trees and rules generated from WEKA are parsed and converted to SQL statements. For the comparative analysis of the several algorithms that were considered, experiments to test and measure their predictive accuracy were conducted. For the classifiers, J48 obtained the best predictive accuracy; for the Clusterers, Simple K-Means with J48 produced the highest predictive accuracy; and for Association, Predictive Apriori has the highest accuracy rate. Overall, J48 stood out to be the best algorithm for prediction of musical genre.

Key Words: Data Mining; Classification; Clustering; Association; SQL; Music Genre Classification, jAudio, WEKA

1. INTRODUCTION

Data mining which is a confluence of many disciplines can be defined in several ways. According to [Rajaraman et al., 2011], the most commonly accepted definition of "data mining" is the discovery of "models" for data. As a general technology, data mining can be applied to any type of data (e.g., data streams, ordered/sequence data, graph or networked data, spatial data, text data, multimedia data, and the WWW) as long as the data are meaningful for a target application [Han, et al., 2012].

Most applications systems use traditional databases to store, access and manipulate data. To widen the appeal of data mining to the developer and user communities, data mining application systems should be convenient to use and be easily deployable in real-world environment. Central to achieving this objective is the integration of data mining with traditional database systems [Chaudhuri, et al., 2001].

This study focuses on the mining of multimedia data, audio in particular. We have built a Music Genre Classifier system that predicts or classifies the genre of an unclassified musical piece on top of a relational database. Basically, the Music Genre Classifier system employs jAudio to extract the relevant features of an unclassified musical piece. The system accepts these features and stores them in MS Access, a relational database system. A SQL query statement, which originally was a prediction model generated by WEKA, but was converted and coded in SQL, is then executed to predict the genre of the musical piece.

In building the prediction models, three types of data mining techniques were examined, namely, classification, clustering and association rule mining. WEKA, a data mining workbench, was utilized for this purpose. It has numerous built-in machine learning algorithms that can generate prediction models from a training data set. However, investigation of the algorithms was restricted to only those that produce rules and decision trees as prediction models. Rules and decision trees can be easily parsed and translated to SQL query statements.

Our study is organized and presented as follows. Section 2 discusses data preprocessing, where the audio files are prepared for data mining. In section 3, the various techniques that were explored to build the data models for prediction are discussed. Also, the results of the experiments conducted are presented. Lastly, in section 4, conclusions of the study are drawn and recommendations for future research are provided.

2. DATA PREPROCESSING

Data preprocessing involves the transformation of raw data into an understandable format. It prepares raw data for data mining [Technopedia, n. d.]. For music, information such as attack, duration, volume, velocity and instrument type of every single note are available. Statistical measures such as tempo and mean key for each music item can easily be extracted [Kotsiantis et al., 2004] and in this study jAudio was used.

The jAudio [Sourceforge, n. d.] is a Digital Signal Processing system that allows users to extract audio features or properties such as beat points, statistical summaries, etc. It has a GUI, an API for embedding jAudio in applications and a command line interface for facilitating scripting. It has functionalities that allow users to set general parameters such as window size, window overlap, down sampling and amplitude normalization. It can also perform audio synthesis, record audio and transfer MIDI files to audio. It has the capability to display audio signals in both frequency and time domains. It can parse MP3, WAV, AIFF, AIFC, AU and SND files. It allows feature values to be created in either ACE XML or WEKA ARFF files [McKay, C., 2010].

In this study, only those features deemed relevant to music genre classification were extracted from jAudio. These features include Spectral Centroid, which is a measure of the "centre of mass" of the power spectrum; Spectral rolloff point, which is a measure of the amount of the right-skewedness of the power spectrum; Spectral flux, which is a good measure of the amount of spectral change of a signal; Compactness, which is a good measure of how important a role regular beats play in a piece of music; Spectral variability, which is a measure of how varied the magnitude spectrum of a signal is; Root mean square (RMS), which is a good measure of the power of a signal; Fraction of low energy windows, which is a good measure of how much of a signal is quiet relative to the rest of a signal; Zero crossings, which is a good measure of the pitch as well as the noisiness of a signal; Strongest beat, which is strongest beat in a signal and it is found by finding the highest bin in the beat histogram; Beat sum, which is a good measure of how important a role regular beats play in a piece of music; Strength of strongest beat, which is a measure of how strong the strongest beat is compared to other possible beats; MFCC, which is a measure of the coefficients that make up the short term power spectrum of sound; LPC, which calculates linear predictive coefficients of a signal; and Method of moments, which is a similar to Area Method of Moments feature, but does not have the large offset [Sourceforge, n. d.]. Two additional features or attributes for each song were added: Class, which is the genre of the song and the ID, which uniquely identifies a song (for reference purposes).

In this study, only five musical genres were considered, namely, classical, country, jazz, reggae and rock. These were selected because they are adjudged to have the most distinct beats and features. A total of 622 songs were collected and preprocessed (feature extraction); 512 of which were used as training data and the rest (110) were used as test data. The training data set contains 100 classical, 100 country, 100 jazz, 100 reggae, 100 rock, 6 classical-rock, 6 country-rock. The test data set contains 20 classical, 20 country, 20 jazz, 20 reggae, 20 rock, 5 classical-rock, 5 country-rock. The extracted features for both training and test data sets were created as ARFF files.

3. BUILDING DATA MODELS

Essentially, building data models for prediction involves the application of data mining techniques that generate meaningful patterns. In this research, the data mining techniques that were considered and studied include classification, clustering and association rule mining. To aid us in the investigation of these many different techniques, we used WEKA, which is a data mining workbench that has evolved immensely in its data mining capabilities. Incorporated in WEKA is an unparalleled range of machine learning algorithms and related techniques. It now includes many new filters, machine learning algorithms, and attributes selection algorithms, and many new components such as converters for different file formats and parameter optimization algorithms. [Witten, et al, 2011]

As described in [Abernethy, 2010], WEKA is the product of the University of Waikato (New Zealand) and was first implemented in 1997. It uses the GNU General Public License (GPL). The software is written in the Java™ language. It contains a GUI for interacting with data files and producing visual results (think tables and curves). It also has a general API, which allows developers to embed WEKA in applications. In terms of functional components, WEKA has three graphical user interfaces, namely, the explorer, experimenter, and knowledge flow and a command line interface. The explorer GUI has six panels which represent a data mining task – preprocess, classify, cluster, associate, select attributes and visualize [The University of Waikato, 2008; Witten, et al., 2011]. In this study, most of the work done was undertaken in the explorer interface, and some via WEKA's command line interface.

As mentioned in section 1, we limited the scope of our investigation of data mining algorithms. We only considered those algorithms that generate rules and decisions trees as models for prediction. Primarily, we did so because rules and decision trees can be easily parsed and translated to SQL query statements.

For classification and association rule mining, the generation of models (in the form of rules and decision trees) and their conversion to SQL query statements is pretty straight forward. However, for clustering, additional steps were undertaken. Since clustering algorithms in WEKA do not generate rules or decision trees, we instead used cluster analysis as a preprocessing technique whereby each sample data in the training set is grouped into a cluster. We then applied J48 to the clustered training data to produce the decision tree.

The succeeding subsections discuss the algorithms that were investigated and present the results of the experiments conducted.

3.1 Classification

According to [Han, et al., 2012], classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts. The model is generated based on the analysis of a set of training data (i.e., data objects for which the class labels are known) and is used to predict the class label of unclassified objects.

The classification algorithms selected are J48, BFTree and RandomTree. For each algorithm, one model per genre is built. In total, there were 45 models generated.

The J48 Decision tree classifier is based on C4.5, an algorithm that was developed by J. Ross Quinlan. In J48 algorithm as described in [Padhye, n. d.], in order to classify a new item, a decision tree based on the attribute values of the available training data is created first. Whenever a set of items (training set) is encountered, an attribute that discriminates the various instances most clearly is identified. This feature is called information gain. It is used to determine the best way to classify the data. Among the possible values of this feature, if there is any value for which there is no ambiguity, the branch is terminated and then target value that was obtained is assigned to it. For the other cases, another attribute that gives the highest information gain is searched. The iteration continues until a clear decision of what combination of attributes gives a particular target value is obtained, or when all attributes has been exhausted. In the event that attributes have been exhausted, or unambiguous result from the available information cannot be obtained, a target value that the majority of the items under this branch possess is assigned to this branch.

On the other hand, the BFTree algorithm builds a best-first decision tree classifier. It uses binary split for both nominal and numeric attributes. For missing values, the method of 'fractional' instances is used [Theofilis, 2013]. Meanwhile, the Random Tree algorithm constructs a tree that considers K randomly chosen attributes at each node. It does not perform pruning [Theofilis, 2013].

Table 1. Comparison of Classifiers: Percentage Prediction Accuracy

	J48	BF Tree	Random Tree
Classical	90.91	90	90.91
Country	85.45	88.18	80.91
Jazz	90	85.45	91.82
Reggae	82.87	84.55	84.55
Rock	88.18	90	87.27

Table 2. Comparison of Classifiers: Kappa statistics, TP rate and FP rate

	Kappa	TP Rate	FP Rate
BFTree	0.61	0.88	0.30
Random Tree	0.60	0.87	0.29
J48	0.63	0.88	0.30

Table 1 shows a comparison of the percentage of predictive accuracy of the three classification algorithms while Table 2 shows the comparison of their Kappa statistics, TP rates and FP rates.

The Kappa statistic or Kappa coefficient is used to measure the agreement between predicted and observed categorizations of a dataset, while correcting for an agreement that occurs by chance. A kappa value of 1 suggests perfect agreement while a kappa value of 0 shows agreement that is equivalent to chance [Witten et al., 2011; the University of Waikato, n. d.]. Based on the results shown, the kappa rate of J48 is highest. It also has the highest TP Rate, the lowest FP rate and highest percentage of predictive accuracy. Thus, among the three classification algorithms that were investigated, J48 stood out to be the best.

3.2 Association Rules

In [Rouse, M. 2011] association rules are described as if/then statements that help reveal relationships between seemingly unrelated data in a data set. An association rule consists of two parts, an antecedent (if) and a consequent (then). An antecedent is an item found in the data while a consequent is an item that is found in combination with the antecedent. Association rules are generated by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships. Support is a measure of how frequently the items appear in the database. On the other hand, confidence indicates the number of times the if/then statements have been found to be true. In data mining, association rules can at times be used for prediction [Deogun et al., 2005].

In this study the association rule mining algorithms that were chosen are Apriori, Filtered Associator, and Predictive Apriori. As described in [Han, et al., 2012], the Apriori is a seminal algorithm that “employs an iterative approach known as a level-wise search, where k-itemsets are used to explore (k + 1)-itemsets. First, the set of frequent 1-itemsets is found by scanning the database to accumulate the count for each item, and collecting those items that satisfy minimum support. The resulting set is denoted by L1. Next, L1 is used to find L2, the set of frequent 2-itemsets, which is used to find L3, and so on, until no more frequent k-itemsets can be found. The finding of each Lk requires one full scan of the database. To improve the efficiency of the level-wise generation of frequent itemsets, an important property called the Apriori property is used to reduce the search space.” The Filtered Associator is an algorithm for “running an arbitrary associator on data that has been passed through an arbitrary filter. Like the associator, the structure of the filter is based exclusively on the training dataset and test instances will be processed by the filter without changing their structure” [Knime n. d.]. On the other hand, PredictiveApriori algorithm “searches with an increasing support threshold for the best 'n' rules concerning a support-based corrected confidence value “[Knime (2), n. d.]. In WEKA, the association rules are not used for prediction. To be able to use them for prediction we converted the association rules produced by the three algorithms to SQL query statements. These statements were then executed to predict the genre of each musical piece in the test data set.

Table 3 shows a comparison of the percentage predictive accuracy of the three association rule algorithms using SQL query statements. Based on the results shown in Table 3, Predictive Apriori has the highest percentage of predictive accuracy.

A comparison of Table 1 and Table 3 shows that classification registered better results than association. It should however be noted that the lower predictive accuracy of association rules is mainly caused by the incomplete generation of the rules due to the lack of computing resources.

Table 3. Comparison of Association Algorithms: Percentage Prediction Accuracy Using SQL

	Apriori	Filtered Associator	Predictive Apriori
Classical	60	60	80
Country	44	44	44
Jazz	45	45	45
Reggae	60	60	60
Rock	70	80	83.33

3.3 Clustering

Clustering diverges from classification and regression. While both classification and regression analyze class labeled (training) data sets, clustering on the other hand analyzes data objects without consulting class labels. In many cases where class labeled data may simply not exist at the beginning, clustering can be utilized to produce class labels for a group of data [Han, et al., 2012].

In clustering, "the objects are grouped based on the principle of maximizing the intraclass similarity and minimizing the interclass similarity. That is, clusters of objects are formed so that objects within a cluster have high similarity in comparison to one another, but are rather dissimilar to objects in other clusters. Each cluster so formed can be viewed as a class of objects, from which rules can be derived" [Han, et al., 2012].

Since clustering algorithms in WEKA, do not generate rules or decision trees, in our study, we used clustering as a technique to preprocess the training data. Based on the outcome of the cluster analysis that was undertaken, we relabeled the class attribute of every instance in the training data set. The attached genre to each musical piece was dropped in favor of the cluster group.

The clustering algorithms used are EM (expectation-maximization), Make Density Based Clusterer and Simple K-Means. As described in [Wikipedia, n. d.], "the EM algorithm is used to find the maximum likelihood parameters of a statistical model in cases where the equations cannot be solved directly." [Wikepedia-1, n. d.]. On the other hand, according to [Wikepedia-2, n. d.], "in density-based clustering, clusters are defined as areas of higher density than the remainder of the data set. Objects in these sparse areas - that are required to separate clusters - are usually considered to be noise and border points." Meanwhile, "K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells" [Wikepedia-3, n. d.].

After performing cluster analysis on the training data, only 2 clusters (that is, a song is either categorized as classical or non-classical) were formed. The clustering algorithms could not clearly distinguish the other 4 musical genres from each other. Consequently, we relabeled the class labels (or genre) of the training data as either belonging to classical or non-classical. The J48 which has the highest percentage of predictive accuracy among the classification algorithms was then applied to generate the decision trees.

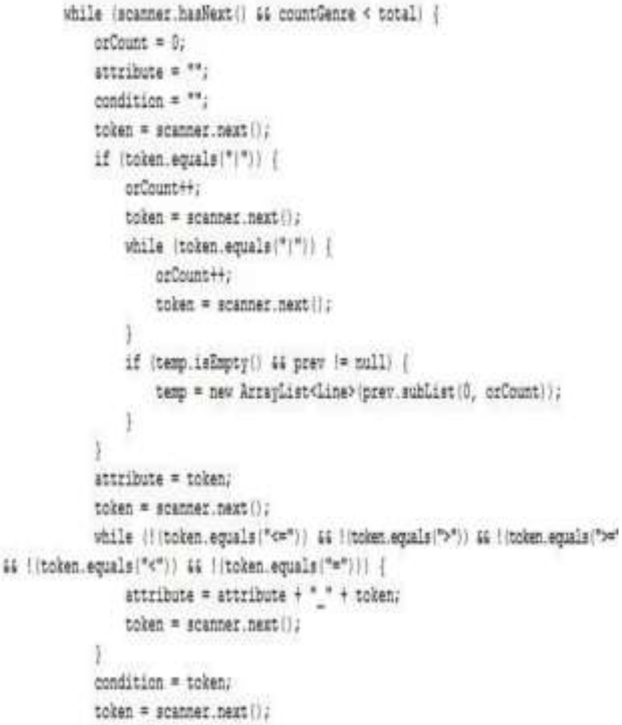
Table 4. Comparison of Cluster Analysis: Percentage Prediction Accuracy using J48

	EM	Simple K-Means	Make Density Based Clusterer
Classical	92.73	92.73	95.45
NonClassical	92.50	93.64	92.28

Table 4 shows a comparison of the percentage predictive accuracy of J48 using the three preprocessed (thru clustering) data sets. Based on the results shown, Simple K-Means with J48 produced the highest percentage of predictive accuracy. However, it should be noted that Country, Jazz, Reggae and Rock were all clustered as non-classical.

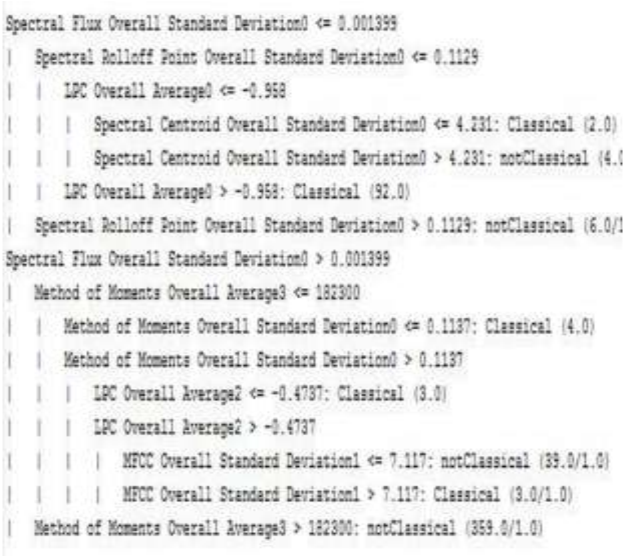
4. GENRE PREDICTION USING SQL

This section shows a snippet of the algorithm that was used to parse and convert the decision tree to SQL Query statements (see Fig 1). Fig 2 shows a sample of a decision tree produced by J48. Fig 3 shows a snippet of the SQL Query statement that was generated by the algorithm shown in Fig. 1.



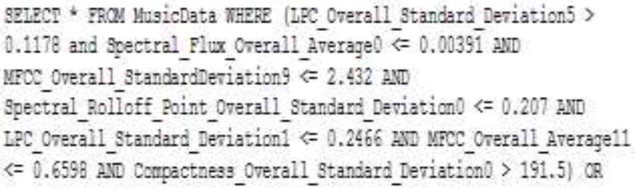
```
while (scanner.hasNext() && countGenre < total) {
    orCount = 0;
    attribute = "";
    condition = "";
    token = scanner.next();
    if (token.equals("|")) {
        orCount++;
        token = scanner.next();
        while (token.equals("|")) {
            orCount++;
            token = scanner.next();
        }
        if (temp.isEmpty() && prev != null) {
            temp = new ArrayList<Line>(prev.subList(0, orCount));
        }
    }
    attribute = token;
    token = scanner.next();
    while (!(token.equals("<=")) && !(token.equals(">")) && !(token.equals(">="
&& !(token.equals("<")) && !(token.equals("="))) {
        attribute = attribute + "_" + token;
        token = scanner.next();
    }
    condition = token;
    token = scanner.next();
```

Fig. 1. Code Snippet for Converting Decision Trees to SQL Select Statements



```
Spectral Flux Overall Standard Deviation0 <= 0.001399
|   Spectral Rolloff Point Overall Standard Deviation0 <= 0.1129
|   |   LPC Overall Average0 <= -0.958
|   |   |   Spectral Centroid Overall Standard Deviation0 <= 4.231: Classical (2.0)
|   |   |   Spectral Centroid Overall Standard Deviation0 > 4.231: notClassical (4.
|   |   LPC Overall Average0 > -0.958: Classical (92.0)
|   Spectral Rolloff Point Overall Standard Deviation0 > 0.1129: notClassical (6.0/1
Spectral Flux Overall Standard Deviation0 > 0.001399
|   Method of Moments Overall Average3 <= 182300
|   |   Method of Moments Overall Standard Deviation0 <= 0.1137: Classical (4.0)
|   |   Method of Moments Overall Standard Deviation0 > 0.1137
|   |   |   LPC Overall Average2 <= -0.4737: Classical (3.0)
|   |   |   LPC Overall Average2 > -0.4737
|   |   |   |   MFCC Overall Standard Deviation1 <= 7.117: notClassical (39.0/1.0)
|   |   |   |   MFCC Overall Standard Deviation1 > 7.117: Classical (3.0/1.0)
|   Method of Moments Overall Average3 > 182300: notClassical (359.0/1.0)
```

Fig. 2. J48 Classical Decision Tree



```
SELECT * FROM MusicData WHERE (LPC_Overall_Standard_Deviation5 >
0.1178 and Spectral_Flux_Overall_Average0 <= 0.00391 AND
MFCC_Overall_StandardDeviation9 <= 2.432 AND
Spectral_Rolloff_Point_Overall_Standard_Deviation0 <= 0.207 AND
LPC_Overall_Standard_Deviation1 <= 0.2466 AND MFCC_Overall_Average11
<= 0.6598 AND Compactness_Overall_Standard_Deviation0 > 191.5) OR
```

Fig. 3. Snippet SQL Statements for J48 Classical Decision Tree in Fig. 2

5. CONCLUSIONS AND RECOMMENDATIONS

In this study, we were able to show that SQL Query statements can be used for genre prediction by converting decision tree and rule-based models generated by WEKA to SQL statements. We were able to integrate of data mining with traditional database systems.

Among the data mining techniques of classification and association rule mining that were investigated, this study found that based on the results of the experiments conducted, J48 classification algorithm has the highest percentage of predictive accuracy.

It was also found that most probably due to the small training data set used in this study, clustering algorithms were only able to identify 2 clusters of data (that is, a musical piece is either categorized as classical and non-classical). As a consequence, clustering as a preprocessing technique was not proven to be useful in this particular case.

The Association rules technique has registered the lowest percentage of predictive accuracy and this was due to some limitations encountered during the study. WEKA required a lot of memory capacity when processing association rule algorithms. Due to the lack of more powerful computing resources, the generation of association rules was not completed; only selected rules were used for prediction.

In the future, to produce more meaningful results, we recommend the use of a larger training data set. In the study conducted by [McKay, 2004], more than 35,000 songs were available for training data.

6. REFERENCES

Abernethy, M. (2010).   Data mining with WEKA, Part 1: Introduction and regression. Developer Works. IBM. Retrieved

January 29, 2012 from http://www.ibm.com/developerworks/library/os-weka1/

Chaudhuri, S. ; Fayyad, U. ; Bernhardt, J. (2001). Integrating data mining with SQL databases: OLE DB for data mining. Proceedings of the 17th International Conference on Data Engineering, 2001. pp. 379 – 387.

Deogun, J. and Jiang, L. (2005) Prediction Mining – An Approach to Mining Association Rules for Prediction. Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing Lecture Notes in Computer Science Volume 3642, 2005, pp 98-108

Han, J., Kamber, M. and Pei, J. (2012). Data Mining: Concepts and Techniques 3rd Edition, Morgan Kaufmann

Kotsiantis, S., Kanellopoulos, D. and Pintelas, P. (2004). Multimedia Mining. WSEAS Transactions on Systems 3 (10), 3263-3268

Knime. (n. d.). FilteredAssociator (3.6). Retrieved on January 29, 2014 from http://www.knime.org/files/nodedetails/weka_associations_FilteredAssociator.html

Knime(2). (n. d.). PredictiveApriori. Retrieved on January 29, 2014 from http://www.knime.org/files/nodedetails/weka_associations_PredictiveApriori.html

McKay, C. (2010). Automatic Music Classification with jMIR. PhD Dissertation. Department of Music Research Schulich School of Music McGill University, Montreal.

Padhye, A. (n. d.). Chapter 5. Classification Methods. Retrieved on January 29, 2014 from http://www.d.umn.edu/~padhy005/Chapter5.html

Rajaraman, A. & Ullman, J. (2010). Mining of Massive Datasets.(available for free on the web)

Rouse, M. (2011) Association Rules (in data mining). Search business analytics. Retrieved on January 29, 2014 from http://searchbusinessanalytics.techtarget.com/definition/association-rules-in-data-mining

Sourceforge (n. d.), jAudio. Retrieved January 29, 2014 from http://jaudio.sourceforge.net/

Technopedia (n. d.). Data Preprocessing. Retrieved on January 31, 2014 from http://www.techopedia.com/definition/14650/data-preprocessing

The University of Waikato (2008). WEKA 3 – Data Mining with Open Source Machine Learning Software in Java. Source URL: http://www.cs.waikato.ac.nz/~ml/weka/

The University of Waikato (n. d.). Primer. Retrieved on January 29, 2014 from http://weka.wikispaces.com/Primer

Theofilis, G. (2013). Weka Classifiers Summary. Retrieved on January 29, 2014 from http://www.academia.edu/5167325/Weka_Classifiers_Summary

Wikipedia-1. (n. d.) Expectation–maximization algorithm. Retrieved on January 31, 2014 from http://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm

Wikipedia-2. (n. d.). Cluster analysis. Retrieved on January 31, 2004 from http://en.wikipedia.org/wiki/Cluster_analysis

Wikipedia-3 (n. d.). k-means clustering. Retrieved on January 31, 2004 from http://en.wikipedia.org/wiki/K-means_clustering

Witten, I. and Frank, E., (2011). Data Mining Practical Machine Learning Tools and Techniques, 3rd Edition, Elseveir