iEMSs 2008: International Congress on Environmental Modelling and Software
Integrating Sciences and Information Technology for Environmental Assessment and Decision Making
4th Biennial Meeting of iEMSs, http://www.iemss.org/iemss2008/index.php?n=Main.Proceedings
M. Sànchez-Marrè, J. Béjar, J. Comas, A. Rizzoli and G. Guariso (Eds.)
International Environmental Modelling and Software Society (iEMSs), 2008

Identifying Biological Echoes in Radar Scans Using Machine Learning

Reginald Mead[a], John Paxton[a] and Richard S. Sojda[b]

[a] Department of Computer Science, Montana State University, Bozeman, MT 59717 USA (mead/paxton@cs.montana.edu)

[b] Northern Rocky Mountain Science Center, U.S.D.I.-Geological Survey, Bozeman, MT 59717-3492 USA (sojda@usgs.gov)

Abstract: Radar ornithology using data from the NEXRAD weather radar system has given scientists new tools for studying bird migration in the United States. Unfortunately, the process of identifying echoes from birds in radar data still largely requires that trained technicians spend hours manually scouring radar scans. This paper provides some background for understanding biological echoes in NEXRAD data and then describes our initial investigations of the use of machine learning techniques to help automate the process of echo classification. Doppler data allows researchers to look at bird migration by examining large clusters of birds that could be observed approaching and descending at stopover points where they would rest until they began the next leg of their journey. One of the NEXRAD system's greatest strengths has also been a significant obstacle for researchers studying bird migration: with 154 radar stations across the United States, each often producing hundreds of volume scans per day, the amount of data to sort through is staggering. The real problem is that classifying birds in radar scans currently requires a skilled technician who has been trained in visually identifying the tell-tale signs that distinguish biological echoes from non-biological echoes. Consequently, the task of plotting a specific migration over any significant amount of space and time quickly becomes a difficult and resource intensive problem. We have begun by using a K-nearest neighbour classifier, a naïve Bayes classifier, and a neural network to classify the echoes. Early validation results using tenfold cross-validation procedures are hopeful and indicate that machine learning techniques could be well suited for this task. Accuracy rates have exceeded 98 percent. Although these early results are encouraging, it is important to keep in mind that each of the training sweeps used in these experiments was selected by an expert because it could be considered a prototypical example of one particular echo type dominating a sweep. Intuitively, these sweeps are the easiest to classify, which may explain the results. The real test will be to apply these methods to more complex data, including ambiguous data as well as mixed sweeps containing both types of echoes. Our next efforts will concentrate on acquiring and experimenting with such data. Our eventual goal is to use machine learning methods to map bird migration pathways.

Keywords: Machine learning, Artificial intelligence, Radar ornithology, NEXRAD, Bird migration

1. Introduction

In the 1940s, researchers discovered that radar could detect the position and movements of birds and other flying creatures such as insects [Gauthreaux and Belser, 2003]. With this discovery, the field of radar ornithology was born. Researchers used this technique to the best of their ability with whatever data was available to them. The real breakthrough came in the 1990s when the United States began replacing its WSR57 weather surveillance radars with WSR-88D Weather Surveillance Radar 1988 Doppler [Diehl and Larkin, 2002]. The WSR-88D or NEXRAD NExt Generation RADar system provided freely available radar data to researchers that covered a broad geospatial range.

By far, one of the most important aspects of the new system was the addition of a Doppler component to supplement the reflectivity data. Doppler data allowed researchers to examine large clusters of migrating birds approaching and descending at stopover points where they would rest until they began the next leg of their journey.

One of the NEXRAD system's greatest strengths has also been a significant obstacle for researchers studying bird migration; with 154 radar stations, each producing hundreds of volume scans per day, the amount of data to process is staggering. The real problem is that classifying birds in radar scans currently requires a skilled technician who has been trained in identifying the tell-tale signs that distinguish biological echoes from non-biological echoes. Consequently, the task of plotting a specific migration over any significant amount of space and time quickly becomes a difficult and resource intensive problem.

This paper describes an approach to this problem that leverages the power of machine learning techniques to automate the process of echo classification and provide a means by which researchers can automatically detect scans of interest. We provide a brief technical background for the problem, describe several of the classifiers investigated, and conclude with some early empirical results and directions for the future.

2. Biological Pattern Recognition

2.1 Data Format

At its most abstract level, NEXRAD data is hierarchically organized into four basic structures: volumes, sweeps, rays, and pulse volumes [Diehl and Larkin, 2002; Klazura and Ima, 1993]. The volume is the highest level structure, representing a snapshot in time of the entire three dimensional space around the radar station. Volumes are composed of sweeps. Sweeps are essentially two dimensional structures containing echoes at the same elevation angle. Sweeps are further subdivided into rays. Like the spokes on a tire, rays extend outward from the radar station and contain all of the pulse volumes that share the same azimuth. This leads to the lowest level of the NEXRAD hierarchy, the pulse volume. A pulse volume is essentially a rectangular volume of space (technically, it is conical in shape) for which the radar produces reflectivity, velocity, and spectrum width values. Figure 1 illustrates the three dimensional organization of these components.

2.2 Data Preprocessing

Before training and classification can take place, six preprocessing steps are performed on the data. Some of these steps are obvious; others not so.

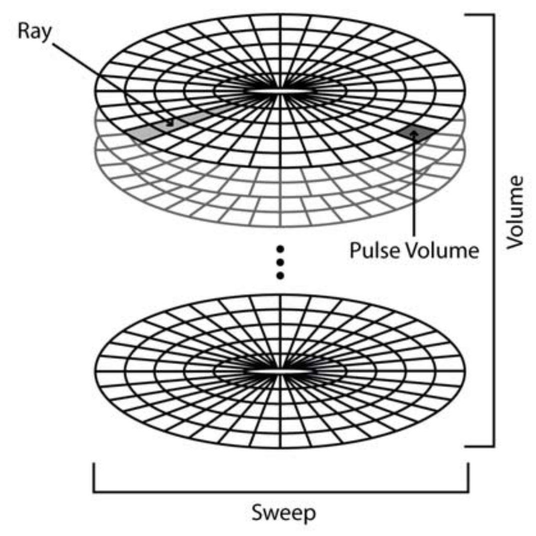

Figure 1. WSR-88D Hierarchical Data Structure

Our first step is to select the sweep or sweeps that we want to use. When studying bird migration, this means using the lowest (.5 degree) elevation sweep. This is the elevation at which most birds will appear.

Our second step is to remove untrustworthy data. This includes both pulse volumes that are very near and pulse volumes that are very far from the radar station. Due to the curvature of the earth and the elevation of the instrument, the radar beam nearest to the ground in the vicinity closest to the radar station. As a result, the echoes nearest the radar station are also the echoes most likely to be the result of ground clutter. To avoid this problem, we discard the first 20 km of data. At the other extreme is data that is very far from the radar. While reflectivity data can be trusted at distant ranges, velocity data past a certain range cannot be. This is due to what has been coined the *Doppler Dilemma* or the *Range-Velocity Ambiguity* [Doviak and Zrnic, 2006]. The *Doppler Dilemma* states that there is a connection between maximum range and maximum velocity: extending one decreases the other. The balance between range and velocity is determined by the pulse repetition frequency of the radar. To avoid using ambiguous velocity data, we discard pulse volumes farther than 145 km (the unambiguous velocity range) from the radar.

Third, after removing untrustworthy data in the second step, pulse volumes with bad or range-folded reflectivity data are removed. Bad values typically indicate empty space resulting from returned echo strength being less than the signal to noise ratio for the radar.

Fourth, bad or range-folded velocity or spectrum width values are set to zero. Unlike bad valued reflectivity data, these values do not typically represent empty space. Therefore, these values are simply zeroed rather than removed so that the algorithm can exploit the uncorrupted reflectivity information.

Fifth, a set of second order features are calculated for each of the three base values. These second order features include variance, kurtosis (a measure of peakedness), and skewness (a measure of asymmetry) [Joanes and Gill, 1998].

Sixth, insignificant or deleterious features are removed from the data. Features, in this context, include the three base values, azimuth, range, and any second order features calculated in the previous step. A feature is classified as insignificant or deleterious depending on the classifier. In the same way, the rationale for removing a feature varies with the classifier. For classifiers that give more weight to features that have a higher correlation with the classification, removing unimportant features can improve the system's execution time. For a classifier that weights all features equally, removing excess features can have a significant effect on classification accuracy.

2.3 General Methodology

Our classification system starts with supervised training data that is comprised of a number of sweeps that have been selected by an expert and each sweep has been classified as either dominated by biological or non-biological echoes. These sweeps are broken into individual pulse volumes and each pulse volume is assigned the classification of its containing sweep.

For validation, the training data is separated into folds, where a fold consists of data from several sweeps. Due to practical computer processing (hardware) limitations, the sweeps are not used in their entirety. Instead, roughly 5% of each sweep is randomly sampled. Our system performs ten-fold cross validation using this training data [Kohavi, 1995].

3. Machine Learning Approaches

3.1 K Nearest Neighbour Classifier

The K nearest neighbour classifier [Wu et al, 2008] is an instance based learning approach. This type of classifier is known as a "lazy" learner because it delays most of the computational workload until the classification phase. The K nearest neighbour classifier treats each training value as a point in n-dimensional space, where n is equal to the number of features being used. In simple systems, training values are stored and no other processing is performed during the training phase. Later, during the classification phase, a distance function is used to find the K nearest neighbours to an unknown instance and the classes of those neighbours are used to assign a class to the new instance. Figure 2 provides an example of a K nearest neighbour classifier trained on two dimensional data. In this example, the unknown instance represented by the gray circle would be classified as non-biological because two of the three closest training instances are non-biological.

Instance based learning classifiers have been successfully applied to a number of problem domains, but there are consequences that must be considered when using this type of classifier [Aha et al., 1991]. When dealing with radar data, the three characteristics of KNN classifiers with the biggest impact are intolerance to noise, intolerance to irrelevant features, and computational complexity. Fortunately, a number of advances have been made in mitigating these problems.

Intolerance to noisy data can often be assuaged by increasing *k*. The idea is to increase the number of neighbours and thereby decrease the effect of noisy training data. The obvious caveat is that increasing k too much can cause the neighbourhood to cross the concept boundary resulting in reduced performance [Wu et al., 2008].

Simple KNN classifiers suffer when irrelevant, or less important, features are included because they can have a large impact on the distance given by the distance function. A related problem arises when features with large ranges of values begin to dominate the distance function. Both of these issues can be addressed by weighting the features [Wu et al., 2008]. Despite the benefits of weighting, lower priority features can still be a problem because weighting them correctly requires specific knowledge regarding the relative importance of features.

High computational cost is the result of calculating distances to every training instance for each new instance during the classification phase. One improvement that is effective for certain applications is to only save training instances that are misclassified [Aha et al., 1991]. These instances generally define the concept boundary.

For our experiments, we used the IBk classifier provided by the Weka machine learning library [Witten and Frank, 2005]. This KNN implementation was used with the classic Euclidean distance function.

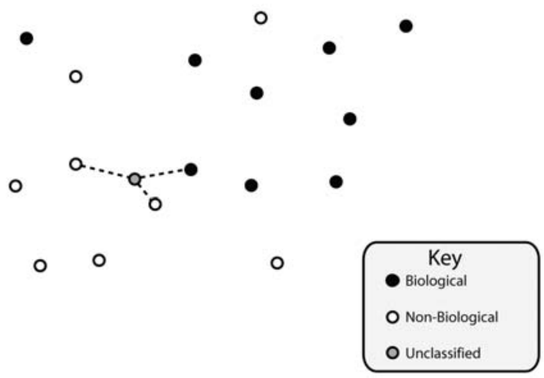

Figure 3. K Nearest Neighbor Classifier Example (K = 3)

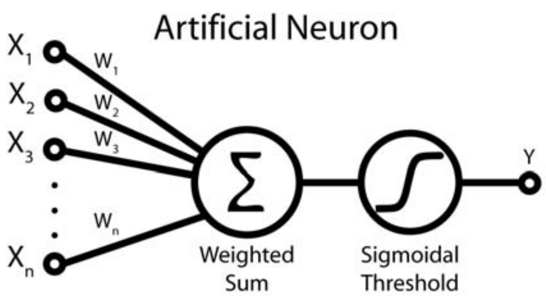

Figure 3. Artificial Neuron

3.2 Naïve Bayes Classifier

The naïve Bayes classifier [Mitchell, 1997] has become popular due to its simplicity, efficiency, and effectiveness. It is often used as a benchmark when experimenting with other, more complex classifiers. This classifier is built based on Bayes' theorem, which states that the probability of seeing event A, given that event B has occurred, can be written as follows:

$$P(A \mid B) = \frac{P(B \mid A) * P(A)}{P(B)}$$

In our classification problem, Bayes' theorem is used to determine the probability of a pulse volume belonging to the biological class given the pulse volume's features. An unknown pulse volume is classified by maximizing this probability across all the possible classes.

In practice, the theoretical Bayes approach needs an enormous amount of training data and takes an impractical amount of computation time due to the P(B|A) factor in the previous theorem. For our problem, P(B|A) is the probability of seeing the set of features given a specific classification. Calculating the joint probability for a set of features given a classification can be a daunting task due to the number of possible combinations that arise when a nontrivial number of features and classes are being considered. We considered up to 14 features and two classes. Each feature had a wide range of possible values. The combinatorial explosion for this feature set makes calculating the joint probability an intractable problem.

The naïve Bayes classifier avoids this problem by making the profound assumption that attributes are independent of each other. Under this assumption, the following conclusion is reached when calculating the joint probability for features A, B, and C given class D:

$$P(A,B,C \mid D) = P(A \mid D) * P(B \mid D) * P(C \mid D) \tag{2}$$

This product is computed by simply counting the frequencies of the various attributes for each of the candidate classes. Bayes' theorem can now be written as follows:

$$P(D \mid A,B,C) = \frac{P(A \mid D) * P(B \mid D) * P(C \mid D) * P(D)}{P(A,B,C)} \tag{3}$$

Bayes' theorem can be further reduced by eliminating the denominator. This is valid because we are only interested in maximizing the probability across the possible classes. The denominator is a normalization factor and remains constant for each of the various possible values (classes) of *D*. For a set of features A, B, and C, a classification Y can be made by maximizing the probability of the class D given the features.

$$Y = MAX\,(P(D \mid A,B,C)) \tag{4}$$

After applying the independence assumption, (4) can be rewritten as:

$$Y = MAX\,(P(A \mid D) * P(B \mid D) * P(C \mid D) * P(D) \tag{5}$$

The independence assumption is controversial within the scientific community. Although this assumption is invalid in most real world applications, studies have shown that making this assumption is largely inconsequential for many problems. As a result, the naïve Bayes classifier has proven to be remarkably efficient and effective [Mitchell, 1997].

The classifier implementation used in this investigation was the naïve Bayes classifier provided by the Weka 3 machine learning library [Witten and Frank, 2005].

3.3 Neural Network Classifier

Neural networks are a computational framework loosely based on the biological neurons contained in the brain. Basheer and Hajmeer [2000] provides an excellent overview of the fundamentals of neural networks. These networks are composed of a number of simple processing units called neurons. Neurons typically map a number of inputs to a single output. A typical neuron associates a set of weights with the inputs such that when a specific input is presented to the neuron, it multiplies each input by its respective weighting. These weighted inputs are then summed and passed through a threshold function to achieve a final output. Typical output bounds are [-1, 1] and [0, 1]. A common threshold function is the sigmoid function [Mitchell, 1997]:

$$S(x) = \frac{1}{1+e^{-x}} \tag{6}$$

Figure 3 illustrates a typical artificial neuron. In this case a weight vector **W** is used to map an input vector **X** to an output Y.

The specific mapping, or function, that a neuron executes is determined by its set of weights. Learning algorithms have been developed to learn a mapping by adjusting the weights until a set of inputs properly produces a set of outputs. A single neuron can learn simple functions. More advanced, often non-linear, functions require a network of neurons working together.

Neural networks typically group neurons into a number of layers, including an input layer, output layer, and any number of hidden layers. An example of a feed-forward neural network can be seen in Figure 4. Feed-forward refers to the unidirectional nature of communication between nodes in this network. A more complex form of neural network is the backpropagation network. This form of network uses the backpropagation learning algorithm to propagate weight changes back through the network [Du, 2006]. Backpropagation networks are one of the most common forms of neural networks used in practical applications. The backpropagation network implemented by the multilayer perceptron classifier in Weka [Witten and Frank, 2005] was used for this investigation.

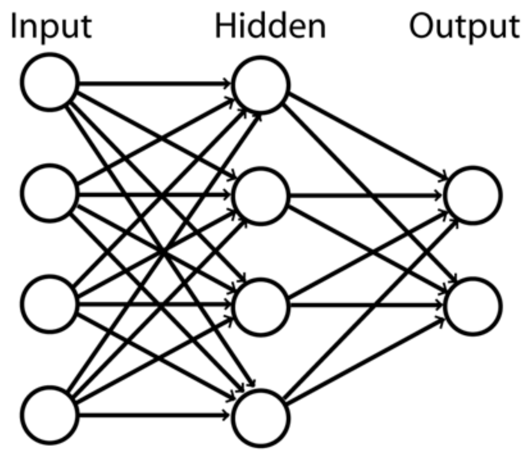

Figure 4. Typical Feed-Forward Neural Network

4. Empirical Validation

Our experiments used ten-fold cross validation [Kohavi, 1995] to evaluate each of the three classifiers. We used 40 sweeps categorized beforehand by radar ornithologists, and each fold consisted of data from four sweeps. Due to computational limitations, only 5% of each sweep was used during training. This resulted in each fold containing approximately 29,000 pulse volumes. The decision to use 5% of the data was based solely on practical considerations. A formal sensitivity analysis examining this decision will be conducted in the future.

After training a classifier on 36 sweeps from nine folds, each of the remaining four sweeps in the tenth fold was categorized by that classifier. Such classification was done at the pulse volume level at this stage. Every pulse volume in the four unclassified sweeps was presented to the classifier as a set of features. Each of the classifiers in the Weka framework produce a classification probability distribution. Although this is natural for naïve Bayes, constructing distributions for the neural network and KNN are not as obvious. The neural network contains output nodes for each of possible classes and each output node produces a numeric score. These scores are then normalized to produce a distribution. In a similar fashion, the class frequencies present in the *k* nearest neighbours are normalized to provide a distribution for KNN. For our problem domain, the classifiers produce a probability distribution over the two possible classes: non-biological and biological echoes. The distribution [.20, .80], for example, would signify that the classifier gives a 20% chance of the pulse volume being non-biological and an 80% chance of being biological. After this distribution was produced, we assigned each pulse volume a class by majority rule.

Next, an entire sweep was assigned a class. Unlike pulse volume classification, sweep classification was not by majority rule. One goal of our work is to reduce the number of sweeps that researchers must consider in their search for biological echoes. As such, eliminating false positives is more important than eliminating false negatives. We found through a series of experiments that using a 70% threshold did not significantly effect classification accuracy, but did substantially bias the algorithm towards producing false negatives rather than the alternative. Therefore, a sweep was only classified as biological if at least 70% of its pulse volumes had been so classified.

Subsequently, each of the remaining nine folds was iteratively processed, resulting in all 40 sweeps being classified once. It is at this stage that we compared the sweep's classification with that provided by the expert. This entire process was repeated for each of the three classification algorithms using the same 40 sweeps.

The following table summarizes the experimental results. The K Nearest Neighbour approach and the Naïve Bayes approach classified sweeps with the same level of accuracy. When time is taken into account, however, the K Nearest Neighbour classifier required nearly 40 times as long to complete.

Table 1. Cross Validation Classification Results

Classifier	Correctly Classified Sweeps	Time to Train and Classify
K Nearest Neighbour	39 / 40	191 min
Naïve Bayes	39 / 40	5 min
Neural Network	40 / 40	17 min

The neural network took three times longer than Naïve Bayes, but made up for the extra time by classifying all 40 sweeps correctly. We used a midrange PC running Ubuntu Linux as our test machine. The test machine contained 1GB of RAM and had a 3.0 GHz Pentium 4 processor.

5. Conclusions and Future Directions

All of the classifiers investigated in this paper performed well on the training data, and the naïve Bayes and neural network classifiers required a fraction of the time used by the K Nearest Neighbour classifier.

These early results are encouraging, but more research is needed. It is important to keep in mind that each of the training sweeps used in these experiments was selected by an expert because it could be considered a prototypical example of one particular echo type dominating a sweep. Intuitively, these sweeps are the easiest to classify, which may explain the results. The real test will be to apply these methods to more complex data. This includes ambiguous data, i.e., data that is not clearly of one type of echo, as well as mixed sweeps containing definitive examples of both types of echoes.

We are currently working with domain experts to acquire training data that covers a broader range of possibilities. Testing the system against more complicated data should bring to light areas in the system that can be improved. If harder data does reduce classification, we will also examine how modifying the basic configuration of some of our classifiers affects accuracy. For example, we will study how varying k, or using a different distance function affects the accuracy of the *k* nearest neighbour classifier.

Our future goal is to develop a system capable of spatially and temporally tracking groups of migrating birds. Such a system could identify migration corridors that are likely to contain migrating birds at certain times of the year as well as day. Applications of this system would include consultation to lessen the environmental impact by development of such things as wind power generation facilities and power lines. In the scientific community, this system would be a tool for scientists researching migration behaviour, allowing them to sift through enormous data sets to study both local bird movements as well as regional migration. Such information can be used by wildlife managers to target their efforts in those areas where birds are either prevalent or are not, depending on their objectives. Likewise, such information can inform development of natural resources to minimize environmental impacts.

To meet this goal, the current system will need to be expanded in a number of ways. Segmentation and boundary detection for classifying groups of echoes at the sub-sweep level will allow us to extract valuable information from sweeps that are not dominated by biological echoes. Another possible improvement would be the inclusion of additional classes or subclasses (i.e., rain, snow, or dust, rather than simply non-biological). The system might also benefit from clustering algorithms that could group similar echo types in an unsupervised way. Our system will also need the ability to consider inter-sweep relationships between pulse volumes. Knowledge of relationships between sweeps (i.e., elevation and acquisition time) would allow the system to track targets as they move through time and space (even traveling into the range of another radar station). Finally, the weather radar system in the United States is being upgraded to a NEXRADII system which will provide an additional four moments of data as well as increase the precision of the data, itself.

Acknowledgements

We would like to thank the United States Department of Interior's Geological Survey and Fish and Wildlife Service (FWS) for facilitating and funding this research. Specifically, we wish to thank J.B. Bortner and M. Green of the FWS for believing in the early stages of this project and recognizing the possibilities for mapping bird migration. Similarly, R.H. Diehl of the University of Southern Mississippi provided early conceptual ideas and critical insights into the problem domain. Subsequently, he and his students provided the training data that we used.

References

Aha, D., D. Kibler, and M. Albert, Instance-based learning algorithms, *Machine Learning*, 6(1), 37-66, 1991.

Basheer, I., and M. Hajmeer, Artificial neural networks: fundamentals, computing, design, and application, *Journal of Microbiological Method*, 43(1), 3-31, 2000.

Diehl, R.H., and R.P. Larkin, Introduction to the WSR-88D (NEXRAD) for ornithological research, *Proceedings of the Third International Partners in Flight Conference*, 876-888, Albany, California, 2002.

Doviak, R.J., and D.S. Zrnic, *Doppler radar and weather observations: second edition*, Dover Publications, 562 pp., Mineola, New York, 2006.

Du, K. –L, and M. N. S. Swamy, *Neural Networks in a Softcomputing Framework*, Springer, 566pp., London, 2006.

Gauthreaux, S., Jr., and C. Belser, Radar ornithology and biological conservation, *The Auk*, 120(2), 266-277, 2003.

Joanes, D., and C. Gill, Comparing measures of sample skewness and kurtosis, *Journal of the Royal Statistical Society: Series D (The Statistician)*, 47(1), 183-189, 1998.

Klazura, G.E., and D.A. Ima, A description of the initial set of analysis products available from the NEXRAD WSR-88D system, *Bulletin of the American Meteorological Society*, 74(7), 1293–1311, 1993.

Kohavi, R., A study of cross-validation and bootstrap for accuracy estimation and model selection, *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 2, 1137-1143, San Mateo, California, 1995.

Mitchell, T.M. *Machine learning*, McGraw-Hill, 414pp., Jurong, Singapore, 1997.

Witten, I. H., and E. Frank, *Data mining: practical machine learning tools and techniques, second edition*, Morgan Kaufmann, San Francisco, California, http://www.cs.waikato.ac.nz/ml/weka/, 2005.

Wu, X. D., V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z. H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, Top 10 algorithms in data mining, *Knowledge and Information Systems*, 14(1), 1-37, 2008.